T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YENİ BİR KAOTİK SİSTEM İLE FPGA TABANLI BİR KAOTİK HABERLEŞME SİSTEMİ TASARIMI VE GERÇEKLEŞTİRİLMESİ

DOKTORA TEZİ

Serdar ÇİÇEK

Enstitü Anabilim Dalı	**:**	**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ**
Enstitü Bilim Dalı	**:**	**ELEKTRONİK**
Tez Danışmanı	**:**	**Prof. Dr. Abdullah FERİKOĞLU**

Ekim 2016

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YENİ BİR KAOTİK SİSTEM İLE FPGA TABANLI BİR KAOTİK HABERLEŞME SİSTEMİ TASARIMI VE GERÇEKLEŞTİRİLMESİ

DOKTORA TEZİ

Serdar ÇİÇEK

Enstitü Anabilim Dalı : ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ

Enstitü Bilim Dalı : ELEKTRONİK

Bu tez 14 / 10 / 2016 tarihinde aşağıdaki jüri tarafından oybirliği ile kabul edilmiştir.

Prof. Dr. **Abdullah FERİKOĞLU** **Jüri Başkanı**	**Doç. Dr.** **Kerem KÜÇÜK** **Üye**	**Doç. Dr.** **Ali ÖZTÜRK** **Üye**
Doç. Dr. **Yılmaz UYAROĞLU** **Üye**	**Yrd. Doç. Dr.** **Murat KARABACAK** **Üye**	

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Serdar ÇİÇEK

14.10.2016

TEŞEKKÜR

Doktora tez çalışmam boyunca değerli yardım ve katkılarıyla beni bilgilendiren ve yönlendiren, her türlü emeği ve ilgiyi gösteren başta danışmanım Prof. Dr. Abdullah FERİKOĞLU olmak üzere Doç.Dr. İhsan PEHLİVAN ve Doç.Dr. Yılmaz UYAROĞLU'na teşekkürlerimi sunarım.

Tez çalışmalarımda ilgi ve yardımlarını esirgemeyen tüm arkadaşlarıma teşekkür ederim.

Anlayış, destek ve yardımlarından ötürü eşime teşekkür ederim.

Ayrıca bu çalışmanın maddi açıdan desteklenmesine olanak sağlayan Sakarya Üniversitesi Bilimsel Araştırma Projeleri (BAP) Komisyon Başkanlığına (Proje No: 2015-50-02-003) teşekkür ederim.

İÇİNDEKİLER

SİMGELER VE KISALTMALAR LİSTESİ

A	: Amper
ASCII	: Bilgi değişimi için Amerikan standart kodlama sistemi
ASIC	: Uygulamaya özel tümleşik devre (Application specific integrated circuit)
AWGN	: Toplanır beyaz gauss gürültüsü - Additive White Gaussian Noise
BER	: Bit hata oranı – Bit error rate
bps	: Saniyedeki bit sayısı – Bir per second
C	: Kondansatör
CCII+	: İkinci nesil pozitif akım taşıyıcı
CDSK	: Correlation delay shift keying
CFOA	: Akım geri beslemeli işlemsel yükselteç
CLB	: Programlanabilir mantık bloğu (Configurable logic block)
CM	: Chaotic masking
COOK	: Chaotic on-off keying
Cov	: Kovaryans
CPLD	: Karmaşık programlanabilir mantık devreleri (Complex PLD)
CSK	: Chaos shift keying
d	: Fraktal boyut
dB	: Desibel
DCSK	: Differential chaos shift keying
D_L	: Lyapunov boyutu
DSP	: Sayısal sinyal işleyici (Digital signal processing)
E	: Denge noktası
E_b/N_0	: Bit enerjisinin gürültü enerjisine oranı
f_{bit}	: Bit frekansı

FF	: Flip-Flop
FIPS	: Federal bilgi işleme standardı - Federal information processing standarts
FKKA	: Farksal kaos kaydırmalı anahtarlama
FM-DCSK	: Frequency modulated differential chaos shift keying
FM-FKKA	: Frekans modülasyonlu farksal koas kaydırmalı anahtarlama
FPAA	: Alanda programlanabilir analog dizi
FPGA	: Alanda programlanabilir kapı dizileri
GRSÜ	: Gerçek rastgele sayı üreteci
h	: Euler adım büyüklüğü
HDL	: Donanım tanımlama dili (Hardware description language)
HRSÜ	: Hibrit rastgele sayı üreteci
ICB	: Ara bağlantı bloğu (In connection block)
IEEE	: Elektrik ve Elektronik Mühendisleri Enstitüsü (The institute of electrical and electronics engineers)
IO	: Giriş çıkış (Input-Output)
IOB	: Giriş çıkış bloğu (Input output block)
J	: Jakobiyen matrisi
KAKA	: Kaotik açma-kapama anahtarlama
Kbps	: Kilo bps
KGKKA	: Korelasyon gecikmeli kaos kaydırmalı anahtarlama
KKA	: Kaos kaydırmalı anahtarlama
KM	: Kaotik maskeleme
LSB	: En düşük değerlikli bit (The least significant bit)
LUT	: Başvuru tablosu (Look up table)
Mbps	: Mega bps
MHz	: Mega hertz
MSB	: En yüksek değerlikli bit (The most significant bit)
NIST	: Ulusal standartlar ve teknoloji enstitüsü - National institute of standarts and technology
ns	: Nano saniye
OPAMP	: İşlemsel yükselteç

PAL : Proglamlanabilir dizi mantık (Programmable array logic)

P-DCSK : Permutation based differential chaos shift keying

PLA : Programlanabilir mantık dizisi (Programmable logic array)

PLD : Programlanabilir mantık elemanı (Programmable logic device)

QCSK : Quadrature chaos shift keying

R : Direnç

s : Saniye

SCSK : Symmetric chaos shift keying

SKKA : Simetrik kaos kaydırmalı anahtarlama

SRSÜ : Sözde rastgele sayı üreteci

t : Zaman

T_b, T_{bit} : Bit periyodu süresi

TDK : Türk Dil Kurumu

V : Volt

Var : Varyans

Vdc : DC gerilim

VHDL : Çok yüksek hızlı entegre devre donanımı tanımlama dili (Very high speed integrated circuit hardware description language)

VN : Negatif kaynak gerilimi

VP : Pozitif kaynak gerilimi

μs : Mikro saniye

ρ : Korelasyon katsayısı

Ω : Ohm

λ : Öz değer

λ_L : Lyapunov üsteli

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

ÖZET

Anahtar Kelimeler: Kaos, kaotik sistemler, kaotik modülasyon, kaotik haberleşme, FPGA, gerçek rastgele sayı üreteci.

Bu tez çalışmasında yeni bir üç boyutlu kaotik sistem elde edilerek, bu yeni kaotik sistem ile FPGA tabanlı şifreli kaotik haberleşme sisteminin tasarımı yapılmış ve gerçekleştirilmiştir. Bu amaçla ilk olarak elde edilen yeni üç boyutlu kaotik sistemin, dinamik analizleri incelenerek nümerik benzetimi, analog elektronik devre ve FPGA üzerinde tasarımı gerçekleştirilmiştir. Böylece yeni kaotik sistemin analog ve sayısal tasarımlar için gerçek ortam uygulamalarında kullanılabileceği gösterilmiştir.

Tez çalışmasında daha sonra elde edilen yeni kaotik sistemin kaotik haberleşme uygulamalarındaki başarımının testi için, Matlab-Simulink programında çeşitli kaos tabanlı sayısal haberleşme uygulamalarının benzetim çalışmaları yapılmıştır. Haberleşme sisteminde şifreleme amacıyla yeni kaotik sistem kullanılarak kaotik bir gerçek rastgele sayı üretecinin benzetimi ve FPGA üzerinde tasarımı gerçekleştirilmiştir. Gerçek rastgele sayı üreteci çıkışları, FIBS 140-1 ve NIST 800-22 rastgelelik testlerine tabi tutulmuş ve testlerden başarılı olmuştur.

Tüm bu aşamalardan sonra şifreli kaotik haberleşme sistemi verici ve alıcı birimi olmak üzere ilk önce benzetim ortamında tasarlanmış ve ardından 32 bit kayan noktalı sayı formatı kullanılarak VHDL programlama dilinde FPGA üzerinde gerçekleştirilmiştir. Gerçekleştirilen FPGA tabanlı şifreli kaotik haberleşme sistemi metin, görüntü ve ses bilgisi kullanılarak test edilmiştir. Haberleşme sisteminin 100 MHz kristal bağlı FPGA ile test çalışmalarında; veri iletim hızı 1,5 Mbps ve veri iletim gecikmesi 6,4 μs olarak verici birim tarafından gönderilen orijinal bilgi alıcı birim tarafından başarılı bir şekilde tekrar elde edilmiştir. Gönderilen orijinal bilgi ile verici birim tarafından iletim ortamına aktarılan bilgi arasındaki korelasyon katsayı değerleri metin, görüntü ve ses bilgisi için sırayla 0,0202, -0,0050, -0,0022 olarak tespit edilmiştir. Korelasyon katsayısı değerleri sıfıra çok yakın olduğundan gönderilen orijinal bilgi ile verici birim tarafından iletim ortamına gönderilen bilgi arasındaki ilişki çok çok düşüktür. Sonuç olarak gerçek ortam uygulamalarında metin, görüntü ve ses gibi sayısal verilerin güvenli şekilde iletilmesinde kullanılabilecek bir FPGA tabanlı şifreli kaotik haberleşme sistemi tasarlanmış ve gerçekleştirilmiştir. Tez çalışmasının sonucu literatürde rastlanılan diğer çalışmalar ile karşılaştırılarak çalışmanın üstünlükleri belirtilmiştir.

DESIGN AND IMPLEMENTATION OF AN FPGA BASED CHAOTIC COMMUNICATION SYSTEM WITH A NEW CHAOTIC SYSTEM

SUMMARY

Keywords: Chaos, chaotic systems, chaotic modulation, chaotic communication, FPGA, true random number generator.

In this thesis work, a new chaotic system is obtained and with which an FPGA based cryptic chaotic communication system is designed and implemented. With this aim, the new 3D chaotic system is first dynamically analyzed, numerically simulated using analog circuit components, and designed on FPGA. Thus, the new chaotic system is shown to be capable of usage in real environment applications for analog and digital designs.

Then, in order to test the performance of the new chaotic system in chaotic communication applications, different chaos based digital communication applications are simulated on MATLAB. In the communication system, a chaotic true random number generator for encryption is simulated and designed on FPGA, using also the new chaotic system. The outputs of the true random number generator are subjected to FIBS 140-1 and NIST 800-22 randomness tests and proved successful.

Afterwards, the new cryptic communication system with its transmitter and receiver units, is first designed on simulation environment and then implemented on FPGA, in VHDL language using 32-bit floating point number format. The implemented FPGA based cryptic chaotic system is tested using text, image and voice information.

In the test work of the 100 MHz crystal connected FPGA based communication system; the information sent at a rate of 1,5 MHz is successfully recovered at the receiver unit with a propagation delay of 6,4 microseconds. Correlation factor between the original information and that transferred by the transmitter unit to the transmission medium are found as 0,0202, -0,0050, -0,0022 for text, image and voice information, respectively. This near-zero correlation factor reveals that the relation between the original signal and the prepared for transmission signal is much low. In conclusion an FPGA based cryptic chaotic communication system, which is apt to secure real environment applications such as text, image and voice data communication is designed and implemented. The results are compared with the literature and its superiorities are indicated.

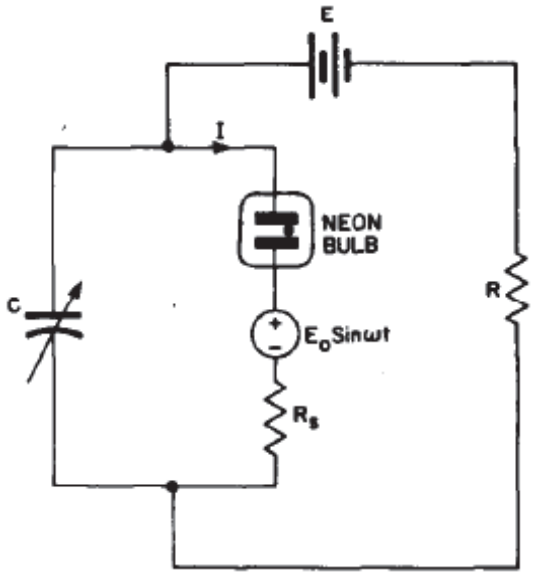
BÖLÜM 1. GİRİŞ

Her an öngörülemez, düzensiz olaylar başta fizikçiler olmak üzere doğa bilimciler için bir sır olmuştur. Bu özelliğinden dolayı bu tip olaylara "bilinmezlik, karışıklık" anlamında "kaos" adı verilmektedir [1]. Kaos tanımı, özetle düzensizliğin düzeni şeklinde ifade edilebilir. Kaos karmaşık davranışlar göstermesinin yanında kendine özel bir iç düzene sahiptir. Karmaşıklık özelliğinin yanında bir düzene sahip olması kaos olayının bir rastgele durum olmadığını belirtir. Dinamik sistemlerin bilinen en karmaşık hali "kaos" dur. Kaos, "determisnistik kaos" ve "rastlantısal (stokastik) kaos" olarak iki durumda incelenebilir. Genelde incelenen kısım deterministik kaos durumudur. Gerçek hayatta var olan olayların çoğu, belirli bölgelerde doğrusal davranış gösterirken, bu bölgelerin dışında doğrusal olmayan (non-lineer) davranış sergilerler [2]. Sigaradan havaya yükselen duman, bir musluktan akan su damlaları, rüzgârın etkisiyle savrulan yaprak, köpüren nehir, kasırgalar vb. bu tür olaylara örnektir [1, 2].

Kaos bilimi doğrusal (lineer) olmayan olayları açıklamak için kullanılır. 1892 yılında Fransız matematikçi olan Jules Henri Poincaré, basit dinamik kuralların çok karmaşık kararlı hal davranışlarına yol açabildiğini keşfetmiştir. Poincaré fizik dünyasındaki hareket kanunlarını araştırmak için geometriden faydalanmıştır. Araştırmalarında kaos ihtimalini anlayan ilk bilgindir [2, 3, 7].

Bilim tarihi boyunca karmaşık ve doğrusal olmayan sistem davranışı birçok alanda gözlemlenmiştir. İlk olarak 1920'li yıllarda Hollandalı bir elektrik mühendisi olan Balthazar van der Pol, neon lamba osilatörü (Şekil 1.1.) üzerinde yaptığı çalışmalar esnasında o zamanlar bilmese de kaosu gözlemlemiştir. Van der Pol, osilatör sinyalinin değişmesini o zamanlar osilaskop olmadığı için telefon ahizesi ile dinleyerek incelemekteydi. Osilatörün kapasitansı değiştikçe osilatörün frekansı

değişmekte ve telefondan dinlediği sesin tonu bir frekanstan diğerine atlıyor ve atladığı frekansa kenetleniyordu. Fakat Van der Pol bazen osilatör frekansının bir frekanstan diğerine atlarken açıklayamadığı düzensiz bir gürültü duymuştur. Fark edilen bu düzensiz gürültü ile ilgili incelemeleri Van der Pol ve J. Van der Mark 1927 yılında Nature dergisinde "Frequency demultiplication" adı ile yayınlamışlardır. 1986 yılında M. Peter Kennedy, periyot çoğullama kaosa götürür tanımından faydalanarak Van der Pol'un düzensiz gürültü olarak tanımlamadığı bu durumun kaos olduğunu göstermiştir [2, 4, 5, 6, 7, 8].



Şekil 1.1. Van der Pol'un kaos sinyallerini gördüğü neon lamba osilatörü [5]

Kaosun matematiksel modeli 1963 yılında ilk olarak Edward Norton Lorenz tarafından ortaya atılmıştır. Meteorolog olan Lorenz hava durumunu tahmin etme üzerine çalışmalar yapmaktaydı. Lorenz hava durumunu tahmin etmek için bir bilgisayar programı yazmıştı ve bu programa veriler girerek hava durumu tahmin denemeleri yapmaktaydı. Denemelerini sürdürürken eski denemesini tekrar incelemek için bilgisayara verileri tekrar girdi. Fakat bu sefer zamandan tasarruf etmek için bilgisayara girdiği verilerin ondalıklı kısımlarını üç hane yuvarlayarak girmişti. Çıkan sonuçları inceleyen Lorenz, sonuçların eski sonuçlar ile hiç alakasının olmadığını gördü. Lorenz farkında olmadan sistemin başlangıç koşullarında çok küçük değişiklikler yapmış ve sonuçta sistemin çok farklı noktalara gittiğini keşfetmişti. Lorenz bu durumu Aralık 1972'de Washington'daki Amerikan

Bilimi Geliştirme Derneği'ndeki konuşmasında "Brezilya'da kanatlarını çırpan bir kelebek, Teksas'da bir tornadoya neden olabilir mi?" sözü ile "dünyanın bir yerinde meydana gelen küçük bir hava akımı dünyanın başka bir yerinde çok büyük olaylara neden olabilir" demiştir. Bu durum "Kelebek Etkisi" olarak tanımlanmaktadır [7, 9]. Lorenz'in kaotik sisteminin matematiksel ifadesi Denklem 1.1'de verilmiştir. Bu denklemde x, y ve z durum değişkenleri, σ, r ve b ise sistem parametreleridir [9]. Lorenz'in bu çalışması bir meteoroloji dergisinde yayımlanmıştı ve yayımlanmasından çok yıl sonralara kadar bu çalışmanın önemi anlaşılamadı.

$$\begin{aligned} \dot{x} &= \sigma.(y - x) \\ \dot{y} &= -xz + rx - y \\ \dot{z} &= xy - bz \end{aligned} \tag{1.1}$$

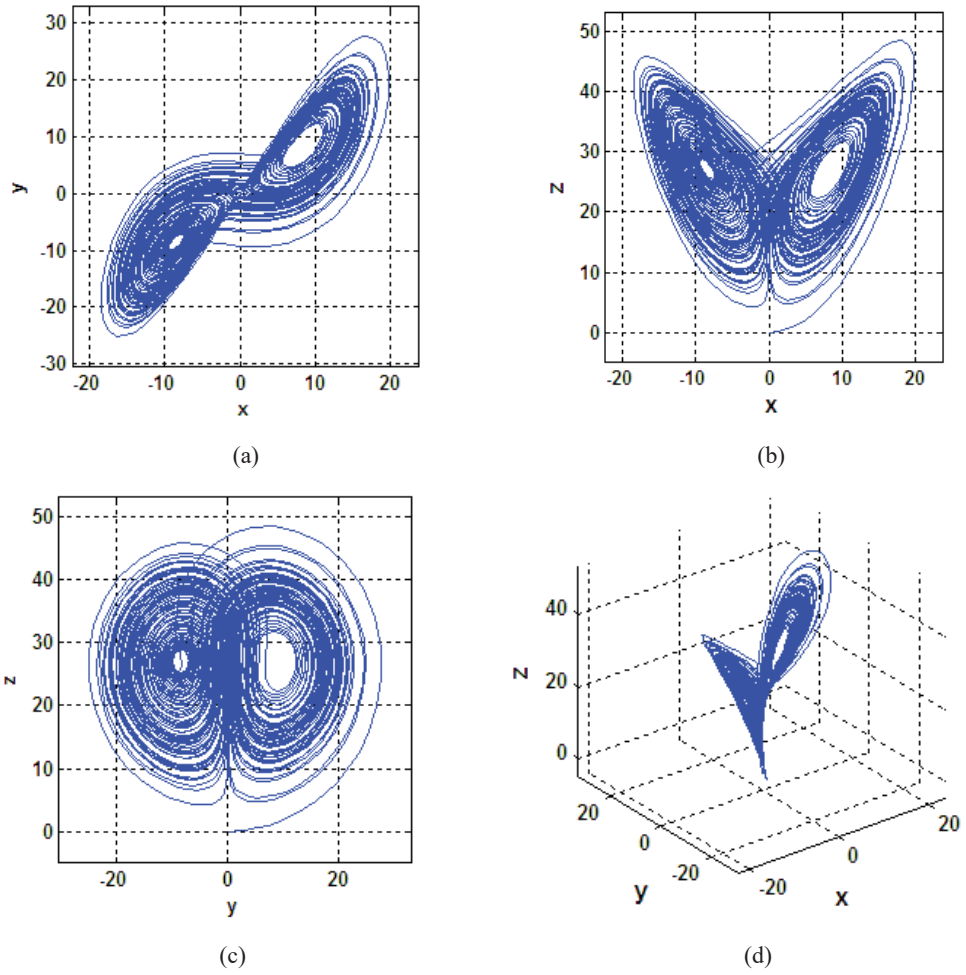
1971 yılında Ruelle ve Takens, Lorenz'in çalışmasından halen haberdar değillerdi ve enerji tüketen dinamik sistemler için "*strange attractor* – garip çekici veya acayip çekici" terimini duyurmuşlardı [8]. 1975 yılında M. J. Feigenbaum, periyot çoğullama kaos belirtisi gösterir tezi ile ilgili makalesi [5] ile kaos olayı anlaşılmaya başlanmıştır. T.Y. LIE ve J.A. YORKE 1975 yılında yayınladıkları "Period three implies chaos - Kaos üçüncü periyotta saklanır" adlı makalelerinde [10], bu karmaşık durumu "kaos" olarak adlandırmışlardır. Bu isim sonraları çok tutmuştur ve bugün kullanılan "kaos" ismi bu makalenin sonucudur [7, 11].

Kaotik sistemler "otonom olan" ve "otonom olmayan" olmak üzere iki sınıfta incelenebilir. Otonom olmayan kaotik sistemlerde sistemin bir durum değişkeni mutlaka zamana (t) bağlıdır. Otonom kaotik sistemlerde ise sistem zamandan bağımsızdır [12].

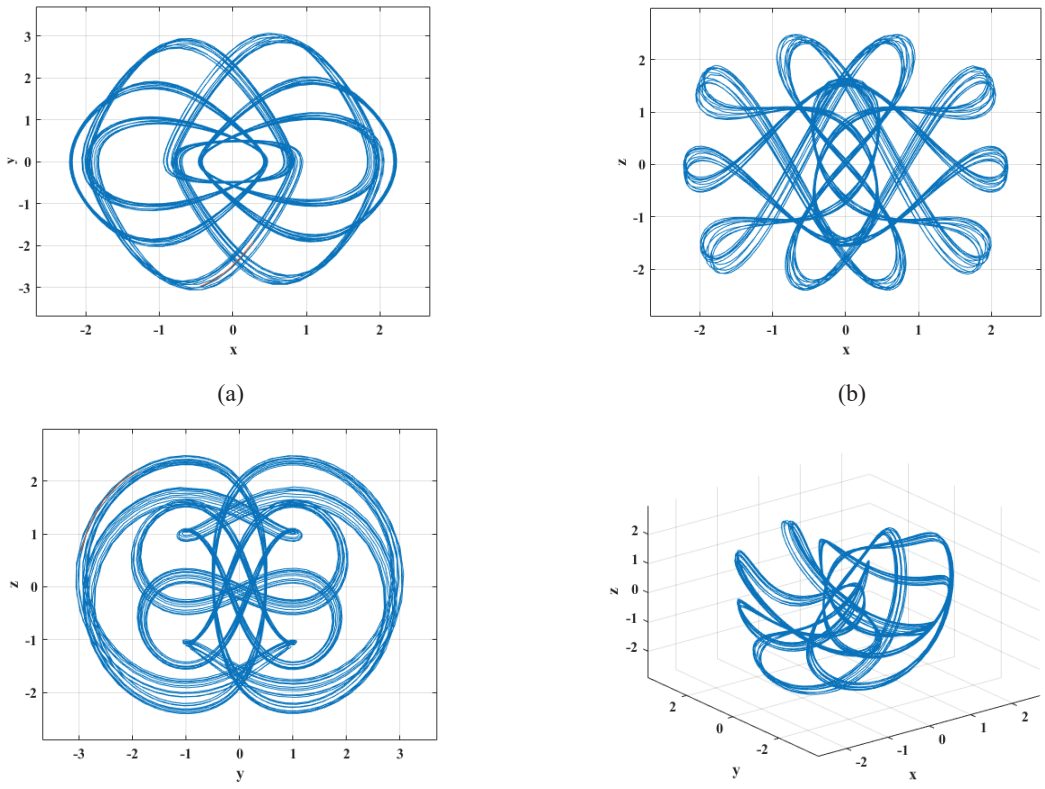
Elektronik devrelerde kaosun ilk olarak gözlemlenmesi otonom olmayan ve harici bir kaynakla sürülen nonlineer osilatör devrelerinde olmuştur. Bunlar, Van der Pol & Van der Mark [6] ile Kennedy & Chua [5] tarafından incelenen sinüzoidal bir kaynak ile çalışan neon lamba osilatörü, Ueda & Akamatsu [13] tarafından geliştirilen

zorlamalı negatif dirençli osilatör ve yine harici bir kaynakla çalışan direnç, indüktör ve diyot elemanlarından oluşan osilatör devreleridir [2, 14, 15].

Otonom olmayan kaotik osilatör devrelerinin yanında çok daha fazla sayıda da otonom kaotik osilatör devreleri yani kaotik sistemler elde edilmiştir. Bu otonom osilatör devrelerinde en çok üzerlerinde çalışma yapılanlar olarak Chua osilatörü [16], Rössler osilatörü [17], Lorenz kaotik sistemi (Şekil 1.2.) [9], Chen kaotik sistemi [18], Sprott kaotik sistemleri [19] (Şekil 1.3.) verilebilir.



Şekil 1.2. Lorenz kaotik sisteminin faz portreleri (a) x-y (b) x-z (c) y-z (d) x-y-z



Şekil 1.3. Sprott-A kaotik sisteminin faz portreleri (a) x-y (b) x-z (c) y-z (d) x-y-z

Kılıç ve arkadaşları tarafından, anahtarlama mekanizma ile hem otonom hem de otonom olmayan kaotik sinyaller üretebilen kaotik devre modeli de tanıtılmıştır [20]. Bu sistemlerden başka kaotik davranış özellikleri çeşitli elektronik devrelerde de gözlemlenmiştir [21-33].

Son yıllardaki çalışmalar incelendiğinde çok farklı dinamik özelliklere sahip çeşitli üç boyutlu kaotik ve dört boyutlu hiper-kaotik sistemlerin literatürde tanıtıldığı görülmüştür [34-44].

Kaosun matematiksel modeli kurulduktan sonra mühendislik, bilgisayar bilimleri, haberleşme, tıp, biyoloji, finans, tüketici elektroniği, enerji vb. alanlarda kullanılmaya başlanmıştır [45]. Kaos ile ilgili bu çalışmalar genelde iki başlık altında toplanabilir. Bunlardan birincisi kaos davranışının sistemde istenmediği durumlar ve bunu önlemek için yapılan kaotik kontrol çalışmalarıdır [46-48]. Kaos ile ilgili çalışmaların toplandığı ikinci başlık ise kaotik sistemlerden olumlu yönde

yararlanma fikri ile yapılan çalışmalardır. Bu çalışmaların içinde çok popüler pratik uygulamalarından biri de güvenli haberleşmedir [49-52].

Kaotik sistemler başlangıç şartlarına çok duyarlıdır ve bazı kaotik sistemler gürültü sinyalleri gibi çok geniş frekans spektrumuna sahiptir. Bundan dolayı kaotik sistemler çeşitli güvenli haberleşme uygulamalarında tercih edilmektedir. Güvenli haberleşme uygulamaları için özdeş veya farklı iki kaotik sistemin senkronize olmaları çok önemlidir. Bu fikrin ortaya çıktığı ilk zamanlarda kaotik sistemlerin başlangıç şartları ve sistem parametrelerine hassas bağımlılığı nedeni ile senkronize olamayacağı düşünülüyordu. İlk olarak 1983 yılında Fujisaka ve Yamada kaotik sistemlerin senkronizasyonu problemini ortaya atmıştır [53]. Daha sonra Pecora ve Carroll, farklı başlangıç şartlarındaki iki özdeş kaotik sistemin senkronizasyonunu gerçekleştirmiştir [54-56]. Son yirmi yılda birçok araştırmacı tarafından kaotik sistemlerin senkronizasyonu konusunda çalışmalar yapılmıştır. Kaotik senkronizasyon için, Pecora-Carroll complete replacement, Pecora-Carroll kaskat, one-way, OGY, feedback, time-delay feedback, adaptif yöntemleri gibi birçok yöntem geliştirilmiş ve bu yöntemlerle çalışmalar yapılmıştır [54-74].

İlk kaotik haberleşme sistemi, 1992 yılında Oppenheim ve arkadaşlarının bilgi işaretine kaotik işaret ekleyerek bilgi işaretinin kaotik sinyal ile maskelenmesi çalışmasıdır [75-77]. Oppenheim ve arkadaşları bu çalışmasında [75] kaotik sistem olarak Lorenz sistemini kullanmışlardır. Aynı kaotik maskeleme mantığını Kocarev ve arkadaşları Chua devresi ile denemiştir [78].

Cuomo ve Oppenheim'in analog ve sayısal bilgi işaretleri için kullanılabilen kaotik maskeleme metodu ile kaotik haberleşme çalışmasından sonra sadece sayısal veriler için kullanılan sayısal (dijital) kaotik modülasyon-demodülasyon metotları ile sayısal kaotik haberleşme çalışmaları da yapılmıştır. Bu sayısal kaotik haberleşme çalışmaları aşağıdaki gibi özetlenebilir [75-86, 165]. Aşağıda verilen yöntemlerden türetilmiş yöntemler üzerinde de çalışmalar yapılmıştır [87-93].

- CM (Chaotic Masking – Kaotik Maskeleme) [75-78]

- COOK (Chaotic On-Off Keying - Kaotik Açma-Kapama Anahtarlama) [79]
- CSK (Chaos Shift Keying - Kaos Kaydırmalı Anahtarlama) [80, 81]
- DCSK (Differential Chaos Shift Keying - Farksal Kaos Kaydırmalı Anahtarlama) [165]
- FM-DCSK (Frequency Modulated Differential Chaos Shift Keying - FM Modülasyonlu Farksal Kaos Kaydırmalı Anahtarlama) [82]
- CDSK (Correlation Delay Shift Keying - Korelasyon Gecikmeli Kaydırmalı Anahtarlama) [83]
- SCSK (Symmetric Chaos Delay Shift Keying - Simetrik Kaos Kaydırmalı Anahtarlama) [83]
- Dörtlü Kaos Kaydırmalı Anahtarlama (Quadrature Chaos Shift Keying – QCSK) [83-86]
- Permütasyon Tabanlı Farksal Kaos Kaydırmalı Anahtarlama (Permutation Based Differantial Chaos Shift Keying – P-DCSK) [83-86]

Kaos çalışmalarının tarihsel olarak gelişimi Tablo 1.1.'de kısaca özetlenmiştir.

Tablo 1.1. Yıllara göre kaos alanı ile ilgili temel gelişmeler

Yıl	Gelişim
1892	Fransız matematikçi olan Henri Poincaré, basit dinamik kuralların çok karmaşık kararlı hal davranışlarına yol açabildiğini keşfetmiştir. Araştırmalarında kaos varlığını ilk anlayan bilim adamıdır.
1920	İlk olarak Hollandalı bir elektrik mühendisi olan Balthazar van der Pol, neon lamba osilatörü üzerinde yaptığı çalışmalar esnasında osilatör frekansının bir frekanstan diğerine atlarken açıklayamadığı düzensiz bir gürültü duymuştur. O zamanlar bilmese de açıklayamadığı bu gürültü ile kaosu gözlemlemiştir.
1927	Vander der Pol ve J. Van der Mark yaptıkları çalışmayı Nature dergisinde "Frequency Demultiplication" adı ile yayınlamışlardır.
1963	Edward Norton Lorenz, hava durumu tahmini denklemi ile yaptığı çalışmalar neticesinde denklemdeki çok çok düşük değişikliğin sonuçta çok büyük farka neden olduğunu görmüştür. O zamanlar kaos ismi kullanılmasa da, Lorenz kaosun ilk matematiksel modelini ortaya atmıştır.

Tablo 1.1. (Devamı)

Yıl	Gelişim
1971	Rulle ve Takens Lorenz'in çalışmasından haberdar değillerdi ve enerji tüketen dinamik sistemler için "strange attractor – garip/acayip çekici" terimini kullanmışlardır.
1972	Lorenz, Washinton'daki Amerikan Bilimi Geliştirme Derneği'ndeki konuşmasında "Brezilya'da kanat çırpan bir kelebek, Texas'ta bir tornadoya neden olabilir mi?" sözü ile dünyanın bir yerinde meydana gelen küçük bir hava akımı dünyanın başka bir yerinde çok büyük olaylara neden olabilir şeklinde yorumlamıştır. Bu durum "Kelebek etkisi" olarak tanımlanmıştır.
1975	M. J. Feigenbaum, periyot çoğullamanın kaos belirtisi gösterir tezi ile ilgili makalesi ile kaos olayı anlaşılmaya başlanmıştır.
1975	T. Y. Lie ve J. A. Yorke, yayınladıkları "Period three implies chaos – Kaos üçüncü periyotta saklanır" adlı makalelerinde bu karmaşık durumu "kaos" olarak adlandırmışlardır. Bu isim sonraları tutmuş ve bugünkü "kaos" ismi bu makalenin sonucudur.
1983	İlk olarak Fujisaka ve Yamada kaotik sistemlerin senkronizasyonu problemini ortaya atmıştır.
1990	Pecora ve Carroll, farklı başlangıç şartlarındaki iki özdeş kaotik sistemin senkronizasyonu gerçekleştirmiştir.
1992	Koatik senkronizasyonun gerçekleştirilmesinden sonra ilk kaotik haberleşme sistemi, Oppenheim ve arkadaşlarının bilgi işaretine kaotik işaret ekleyerek kaotik maskeleme çalışması olmuştur.
1992 – 2000	Sayısal (dijital) kaotik haberleşme için COOK, CSK, DCSK, FM-DCSK, CDSK, SCSK gibi farklı kaotik haberleşme metotları geliştirilmiştir.

Nümerik olarak benzetim (simülasyon) çalışmaları yapılan kaotik sistemlerin, senkronizasyon ve kaotik haberleşme uygulamalarının ardından, OPAMP, CFOA, CCII+ gibi aktif elemanlar ve direnç, kondansatör gibi pasif elemanlar ile analog olarak gerçek zamanlı uygulamalarda kullanılmak üzere çeşitli elektronik devre tasarımları da gerçekleştirilmiştir [94-109].

Kaotik sistemlerin ve uygulamalarının analog elemanlar ile tasarımının yanında dijital olarak FPGA (Field Programmable Gate Array – Alanda Programlanabilir

Kapı Dizileri) tümleşik elemanı da kaotik sistem uygulamalarında kullanılmıştır [110-120].

Tümleşik devre olarak FPGA ile yapılan tasarımların yanında FPGA tümleşik devresinin analog eşdeğeri olarak tanımlanan FPAA (Field Programmable Analog Array – Alanda Programlanabilir Analog Dizi) tümleşik devresi ile de kaotik sistem tasarımları gerçekleştirilmiştir [121-124].

Bilişim teknolojisi çok hızlı bir şekilde gelişim göstermiş ve günümüzde de artan bir hızda gelişim göstermeye devam etmektedir. Aynı paralellikle artık matematik, fizik, kimya, biyoloji, mühendislik, sosyal bilimler, güzel sanatlar, tıp vb. her alanda bilişim teknolojisi kullanılmakta ve bilişim teknolojisine bağımlılık artmaktadır. Bilişim teknolojisinin temelini ise bilgi (veri) oluşturmaktadır. Bu bağlamda bilgi güvenliği öne çıkan bir konu olmaktadır. Bilgi güvenliği konusunda ise bilginin şifrelenmesi ve güvenli haberleşme sistemleri ile iletimine ihtiyaç duyulmuş ve bunun sonucunda da genel olarak bilginin güvenli bir şekilde iletimi sorunu ortaya çıkmıştır. Bu sorunun çözümü için bilinen klasik yöntemlerin yanında yeni yöntemler geliştirilmeye çalışılmış ve çalışılmaya devam edilmektedir. Kaotik sistemlerin özelliğinden dolayı kaotik haberleşme de bilginin güvenli bir şekilde iletilmesi için tercih edilen bir yöntem olmaktadır.

Analog elemanlar ile gerçekleştirilen kaotik haberleşme sistemlerinde verici taraftan gönderilen bilgi (mesaj) sinyalinin alıcı tarafta doğru bir şekilde tekrar elde edilebilmesi için verici ve alıcı birimlerinin birbirleriyle tam senkronize olması gereklidir. Bunun sağlanması için verici ve alıcı sistemdeki tüm elektronik elemanların aynı özelliği göstermesi gereklidir. Fakat direnç, kondansatör gibi analog elemanların değerleri ortamın sıcaklığına, nem miktarına, elemanın kullanılma süresine, toleransına bağlı olarak değişmektedir. Bu durum da alıcı birimde verici birim ile eş olarak kurulan kaotik devrenin ortam şartlarına göre farklı sonuç üretmesine neden olabilmektedir.

Daha iyi bir çözüm olarak sayısal (dijital) donanım ile kaotik sistemler ve buna paralel olarak daha iyi bir kaotik haberleşme sistemi tasarlanabilir. FPGA tümleşik elemanları sayısal olarak işlem gördüklerinden ve ortama bağlı parametre değişiklikleri toleransı çok çok az olduğundan analog elemanlara oranla çok daha hassas sonuçlar üretebilmektedirler. Ayrıca FPGA tümleşik elemanları paralel işlem yetenekleri, yüksek çalışma hızları, tekrar programlanabilme gibi özellikleriyle analog elemanlara göre çok daha esnek bir tasarım ve geliştirme ortamı sunabilmektedir.

Bu bağlamda yapılacak tez çalışmasında önce güvenli haberleşme için kullanılabilecek yeni bir kaotik sistemin elde edilmesi ve dinamik analizlerinin incelenmesi hedeflenmiştir. Ardından elde edilen yeni kaotik sistem ile sayısal kaotik haberleşme metotlarının benzetim (simülasyon) çalışmaları yapılması ve ardından bir kaotik haberleşme yöntemi ile FPGA üzerinde kaotik bir haberleşme sisteminin tasarımı amaçlanmıştır.

Tezin amacı doğrultusunda, ikinci bölümde kaotik sistemler ile ilgili temel kavramlar hakkında bilgi verilmiştir.

Üçüncü bölümde kaotik haberleşme yöntemleri hakkında bilgi verilmiştir.

Dördüncü bölümde, tez çalışması kapsamında elde edilen yeni üç boyutlu kaotik sistem tanıtılmış, dinamik analizleri, nümerik benzetimi ve elektronik devre tasarım çalışmaları verilmiştir.

Beşinci bölümde, nümerik benzetimi yapılan yeni kaotik sistemin FPGA üzerinde tasarım çalışması verilmiştir.

Altıncı bölümde, yeni kaotik sistem ile ilgili çeşitli kaotik haberleşme yöntemleriyle benzetim çalışmaları yapılmıştır.

Yedinci bölümde, tasarlanacak haberleşme sisteminde şifreleme biriminde kullanmak üzere yeni kaotik sistemden gerçek rastgele sayı üreteci (GRSÜ) tasarımının hem nümerik benzetim hem de FPGA üzerinde gerçekleştirilmesi çalışmaları ile tasarlanan GRSÜ tarafından üretilen bilginin FIPS 140-1 ve NIST 800-22 test sonuçlarına yer verilmiştir.

Sekizinci bölümde, yeni kaotik sistem ile şifreli bir kaotik haberleşme sisteminin nümerik benzetim ile tasarım çalışmaları ve performans sonuçları verilmiştir.

Dokuzuncu bölümde, nümerik benzetimi yapılan şifreli kaotik haberleşme sisteminin FPGA tabanlı tasarımı ve gerçekleştirilmesi açıklanmıştır. Ayrıca sistemin metin, ses ve görüntü bilgisi iletim testleri ve performans sonuçlarına yer verilmiştir.

Tezin onuncu bölümü ise tez çalışmasının sonuçları ve ileriki çalışmalar için önerileri içermektedir.

BÖLÜM 2. KAOTİK SİSTEMLER

Bu bölümde kaotik sistemlerinde dâhil olduğu dinamik sistemler ve kaos analizi hakkında genel bilgiler verilmiştir.

2.1. Dinamik Sistem Tanımları

Dinamik sistemler, bir sistemin zaman içinde ileri doğru gelişimini gösteren bir deterministik matematiksel reçete olarak tanımlanabilir. Burada zaman, devamlı (continuous) veya ayrık (discrete) bir değişken olabilir. Denklem 2.1'de sürekli bir sistemin vektör formunda gösterimi, Denklem 2.2'de ise adi diferansiyel eşitlikler ile gösterimi verilmiştir. Denklem 2.1'de x, N boyutlu bir vektörü temsil etmektedir [125].

$$dx(t)/dt = F[x(t)] \tag{2.1}$$

$$\left.\begin{aligned} dx_{(1)}/dt &= F_1(x^{(1)}, x^{(2)}, ..., x^{(N)}), \\ dx_{(2)}/dt &= F_2(x^{(1)}, x^{(2)}, ..., x^{(N)}), \\ &\vdots \\ dx_{(2)}/dt &= F_N(x^{(1)}, x^{(2)}, ..., x^{(N)}), \end{aligned}\right\} \tag{2.2}$$

Her $x(0)$ giriş durumu için, $t > 0$ iken gelecekteki sistem durumu $x(t)$, elde edilebilir olduğundan Denklem 2.1 bir dinamik sistemdir. Dinamik sistemler ve dolayısıyla kaotik sistemler zamana bağlılığı "otonom olan" ve "otonom olmayan" olmak üzere iki sınıfta incelenebilir. Otonom olmayan kaotik sistemlerde sistemin bir durum değişkeni mutlaka zamana (t) bağlıdır. Otonom kaotik sistemlerde ise sistem

zamandan bağımsızdır [12]. Denklem 2.1 otonom bir dinamik sistemi ifade etmektedir. Denklem 2.3 ise otonom olmayan bir dinamik sistemi ifade eder [2].

$$dx(t)/dt = F[x(t), t] \tag{2.3}$$

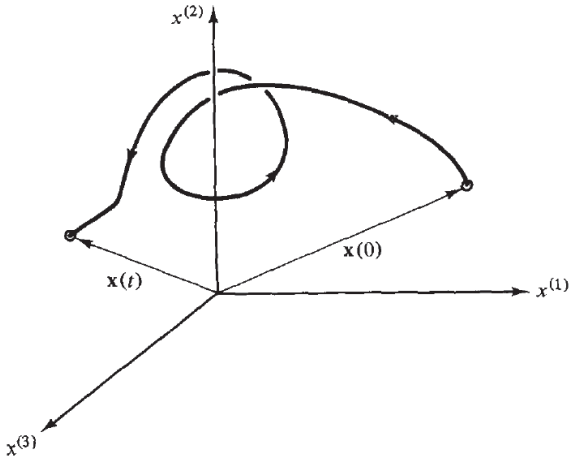
Dinamik sistemler Denklem 2.1 veya Denklem 2.3'te gösterildiği gibi sürekli olabileceği gibi ayrık zamanlı da olabilir. Denklem 2.4'te ayrık zamanlı otonom dinamik sistem ve Denklem 2.5'te ayrık zamanlı otonom olmayan dinamik sistem gösterimi vektör formda verilmiştir [125].

$$x_{n+1} = M(x_n) \tag{2.4}$$

$$x_{n+1} = M(x_n, t_n) \tag{2.5}$$

Denklem 2.4'te x_n, N bileşene sahiptir $x_n = (x_n^{(1)}, x_n^{(2)}, ..., x_n^{(N)})$ ve n, zaman değişkenini temsil eder $(n = 1, 2, ...)$. Sistemin bir zaman değerinden bir sonraki zaman değerine gitme kuralını M fonksiyonu belirler. Başlangıç durumu x_0 verildiğinde $x_1 = M(x_0)$ ile zaman *n=1* iken ki iterasyon durumu elde edilir. x_1 hesap edildiğinde $x_2 = M(x_1)$ ile zaman *n=2* iken ki iterasyon durumu elde edilir. Bu şekilde ayrık zamanlı sistemin yörüngesi elde edilir [125, 126].

Bir dinamik sistemin mümkün olan tüm durumlarının birleşimi o dinamik sistemin faz uzayını (durum uzayını) (phase portrait) oluşturur. Dinamik sistemin bir başlangıç şartı ile çözümü sonucu elde edilen durum değişkenleri değerlerinin faz uzayına iz düşümleri dinamik sistemin yörüngesi (trajectory, orbit) olarak tanımlanır. Sürekli zaman dinamik sistemler için yörünge kavramı akış (flow) olarak da ifade edilir. Şekil 2.1.'de üç boyutlu faz uzayında bir yörünge şekli verilmiştir [125, 127].



Şekil 2.1. Üç boyutlu faz uzayında bir yörünge örneği [121]

Denklem 2.1'de ifade edilen otonom bir dinamik sistemin vektörel alanı bir fonksiyon olduğundan faz uzayında her noktadaki *x(t)* değeri tektir. Bu durum sistemin bir yörüngesinin aynı noktadan iki kez geçmeyeceği anlamını doğurur [128]. Bu özellik kaotik sistemlerin şifreleme, güvenli haberleşme gibi alanlarda kullanılmasını arttırmıştır.

2.2. Dinamik Sistemlerde Denge Noktaları ve Kararlılık Analizi

Doğrusal olmayan (nonlinear) bir dinamik sistemin denge noktaları (equilibrium) o sistemin davranışı hakkında bilgi verir. Sistemin denge noktalarını bulmak için Denklem 2.6'da verildiği gibi sistem sıfıra eşitlenir. $dx(t)/dt = F[x(t)] = 0$ eşitliğini sağlayan denge noktaları, yakınlarındaki çözümlerin davranışını da temsil eder. Böylece doğrusal olmayan bir dinamik sistemin davranışı, elde edilen denge noktaları etrafında doğrusal bir sistem gibi yaklaşık olarak incelenebilir [2].

$$dx(t)/dt = F[x(t)] = 0 \tag{2.6}$$

Kaotik sistemler kararsız bir davranış gösterirler. Bir dinamik sistemin özdeğerlerinden (eigenvalue) en az biri pozitif ise dinamik sistem kararsızdır. Dinamik sistemin özdeğerlerini bulmak için ilk önce sistemin Jakobiyen matrisi Denklem 2.7'de verildiği gibi bulunur.

$$J = \begin{bmatrix} \partial F_1 / \partial X_1 & \partial F_1 / \partial X_2 & \cdots & \partial F_1 / \partial X_n \\ \partial F_2 / \partial X_1 & \partial F_2 / \partial X_2 & \cdots & \partial F_2 / \partial X_n \\ \vdots & \vdots & \ddots & \vdots \\ \partial F_n / \partial X_1 & \partial F_n / \partial X_2 & \cdots & \partial F_n / \partial X_n \end{bmatrix} \tag{2.7}$$

Denklem 2.7'de hesaplanan Jakobiyen matrisi kullanılarak Denklem 2.8'de verilen determinant ile özdeğerler (λ) hesaplanır. Denklem 2.8'de I, birim matrisi temsil etmektedir.

$$\det(J - \lambda.I) = \left|J - \lambda.I\right| = 0 \tag{2.8}$$

2.3. Kaotik Sistem Özellikleri

Kaotik sistemler karmaşık davranış gösteren dinamik sistemlerdir. Kaotik sistemlerde görülen özellikler aşağıdaki gibi özetlenebilir. Bu özelliklerden sadece birinin olması o dinamik sistemin kaotik olduğunu göstermez. Yani bu özelliklerin birkaçının kaotik sistemlerde olması gerek ama yeter şart değildir [129-131].

- Zaman domeninde düzensiz davranış
- Sınırsız sayıda farklı aperiyodik salınım
- Gürültü benzeri geniş güç spektrumu
- Başlangıç koşullarına hassas bağımlılık
- Pozitif Lyapunov üsteli
- Sistem boyutunun fraktal olması

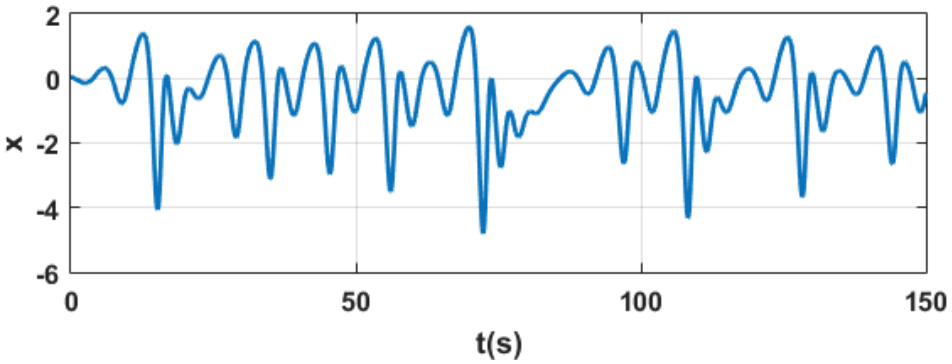
2.4. Kaos Analiz Yöntemleri

Bir dinamik sistemin kaotik özellik gösterip göstermediğini analiz etmek için çeşitli yöntemler vardır. Bu yöntemler ile o sistemdeki kaos varlığı açığa çıkartılabilir. Aşağıda kaos analiz yöntemleri maddeler halinde verilmiştir [132].

- Yörünge planı (Trajectory plot)
- Faz uzayı (Phase portrait)
- Poincaré haritası (Poincaré map)
- Güç spektrumu (Power spectrum)
- Lyapunov üstelleri (Lyapunov exponents, *LEs*) ve Başlangıç şartlarına hassas bağımlılık
- Lyapunov boyutu (Lyapunov dimension, D_L) veya Fraktal boyut (Fractal dimension)
- Çatallaşma diyagramı (Bifurcation diagram)

2.4.1. Yörünge planı

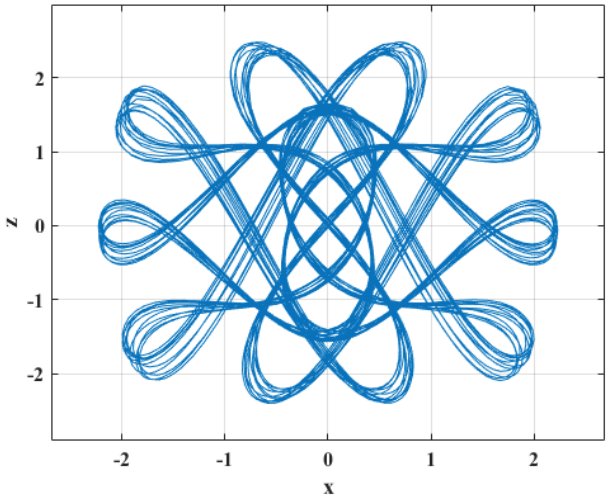
Kaotik sistem durum değişkenlerinin her birinin zamana göre değerleri çizdirilip izlendiğinde aperiyodik (periyodik olmayan) bir davranış sergilemesi gereklidir. Yörünge planı yöntemi ile bu durumun var olup olmadığı gözlemlenebilir [133]. Şekil 2.2.'de kaotik bir sistemin durum değişkeninin zamana göre çizdirilmiş değerleri verilmiştir.



Şekil 2.2. Kaotik bir sistem durum değişkeninin zamana göre değerlerinin çizimi

2.4.2. Faz uzayı

Kaotik sistemlerin durum değişkenleri birbirlerine göre çizdirildiklerinde yani faz uzayı incelendiğinde belli bir düzen sınırı içinde karmaşık bir şekil ortaya çıkar. Bu durum sistemin kaotik davranış sergilediğini gösterir [132, 133]. Sabit bir değerin faz uzayı bir nokta, periyodik bir sinyalin faz uzayı kapalı bir eğri, yarı-periyodik (quasi-periodic) sinyallerin faz uzayı torus şeklinde olur [133]. Şekil 2.3.'te Sprott-A kaotik sisteminin x ve z durum değişkenlerinin faz uzayı verilmiştir. Şekil 2.3.'ten görüldüğü üzere faz uzayı belli bir sınır içinde karmaşık bir davranış göstermiştir. Yani düzen içinde düzensiz bir davranışla kaotik durum sergilemiştir.



Şekil 2.3. Sprott-A kaotik sistemi x-z faz uzayı

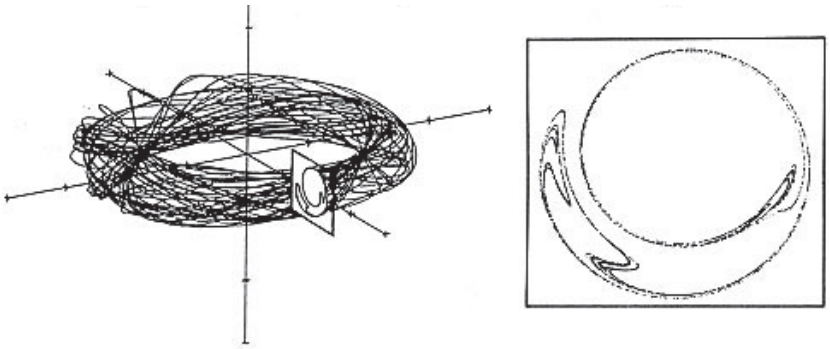
2.4.3. Poincaré haritası

Kaotik sistemlerin oluşturduğu faz uzayının yani garip çekicilerin (strange attractor) oluşturduğu üç boyutlu şekillerin iç yapısını incelemek çok zordur. Üç boyutlu bu şekilleri düz resimler haline dönüştürmek için ilk önceleri projeksiyon tekniği kullanılmıştır. Fakat bu yöntemin kullanılması karmaşık garip çekiciler söz konusu olunca çok zorlaşmaktaydı. Bu soruna çözüm olarak Poincaré dönüşümü veya haritalaması tekniği kullanılmaktadır. Bu teknik ile garip çekicilerin karmaşık durumundan ince bir dilim halinde iki boyutlu kesit çıkarılır. Yani bir çekicinin içinden bir boyut çıkarılmakta ve bu sayede devamlı bir çizgi noktalardan oluşan bir

topluluk haline getirilir [7]. Diğer bir tanımla, n. dereceden sürekli zaman bir sistemin, (n-1). dereceden ayrık zaman sistemle yer değiştirilmesidir.

Poincaré haritalaması için kaotik sistemin faz uzayından Poincaré kesiti olarak adlandırılan bir kesit seçilerek bu kesit üzerinde yörüngenin geçtiği noktalar Poincaré dönüşümü ile elde edilip işaretlenerek bir harita elde edilir. Diğer bir ifade ile Poincaré kesiti, dinamik sistemin hareketinin dondurulmuş halidir. Eğer incelenen sistem periyodik davranış gösteriyorsa Poincaré haritasından sabit bir nokta elde edilir. Eğer sistem yarı-periyodik ise kapalı bir çevrim şeklinde noktalar elde edilir. Fakat sistem kaotik ise rastgele fraktal şekil oluşur [132-134].

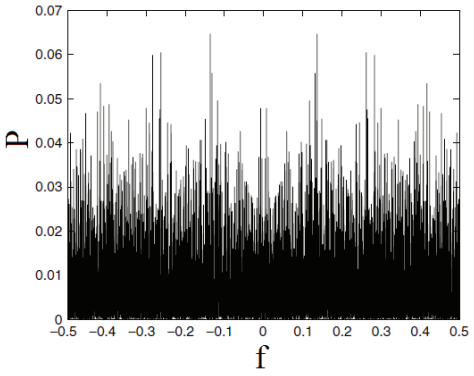
Şekil 2.4.'te bir kaotik sistemin faz uzayından Poincaré dönüşümü ile elde edilen Poincaré haritası verilmiştir [7].



Şekil 2.4. Poincaré dönüşümü ile kaotik bir sistemin faz uzayından elde edilen Poincaré haritası [7]

2.4.4. Güç spektrumu

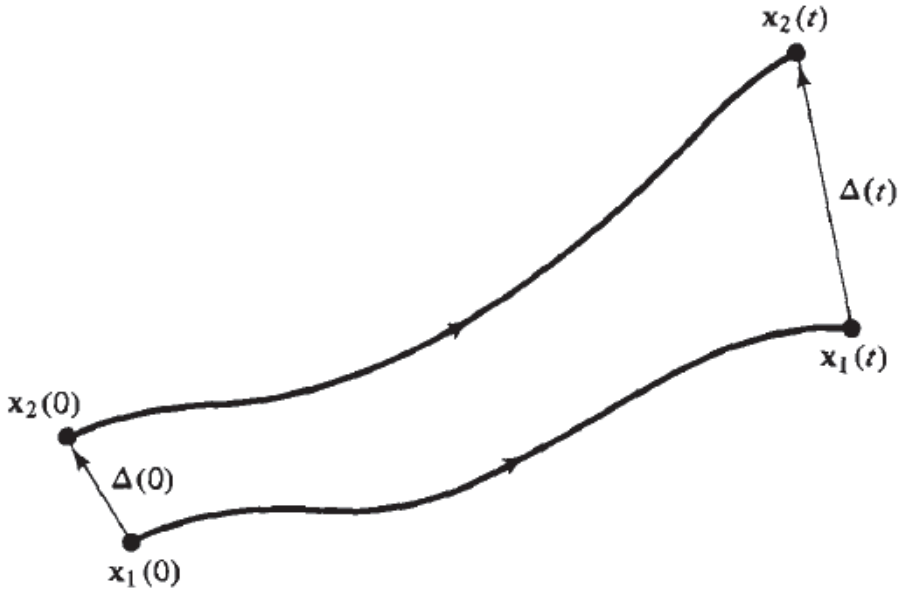
Kaotik sinyallerin bazılarının güç spektrumu veya diğer ifade ile frekans spektrumu geniştir ve gürültü benzeri bir durum sergilerler. Bu sayede bir sinyalin kaotik olup olmadığı güç spektrumundaki genişlik ve dağılıma bakılarak belirlenebilir. Şekil 2.5.'te örnek bir kaotik sinyalin güç spektrumu (veya frekans spektrumu) verilmiştir [135].



Şekil 2.5. Kaotik bir sinyalin güç spektrumu (frekans spektrumu) örneği [131]

2.4.5. Lyapunov üstelleri ve başlangıç şartlarına hassas bağımlılık

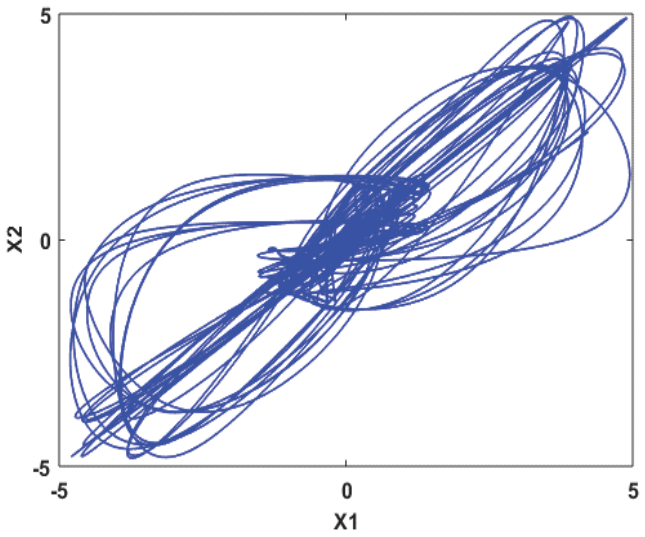
Kaotik sistemler başlangıç şartlarına çok hassas bağımlıdırlar. Başlangıç şartlarında meydana gelen küçük bir fark sistemin ileriki zaman davranışında çok büyük bir farka neden olur. Başlangıç değerindeki küçük değişimin sistem yörüngesinde meydana getirdiği daha büyük değişim Şekil 2.6.'da gösterilmiştir [125].



Şekil 2.6. Başlangıç değerindeki küçük değişimin yörüngede meydana getirdiği daha büyük değişim [125]

Kaotik bir sistemin başlangıç şartına hassas bağımlılığına örnek olarak Şekil 2.7. incelenebilir. Şekil 2.7.'de bir kaotik sistemin *X* durum değişkeninin başlangıç şartı $X_1 = -4$ ve $X_2 = -4{,}0001$ alınmıştır. Aralarında sadece 0,0001 fark bulunan bu iki farklı

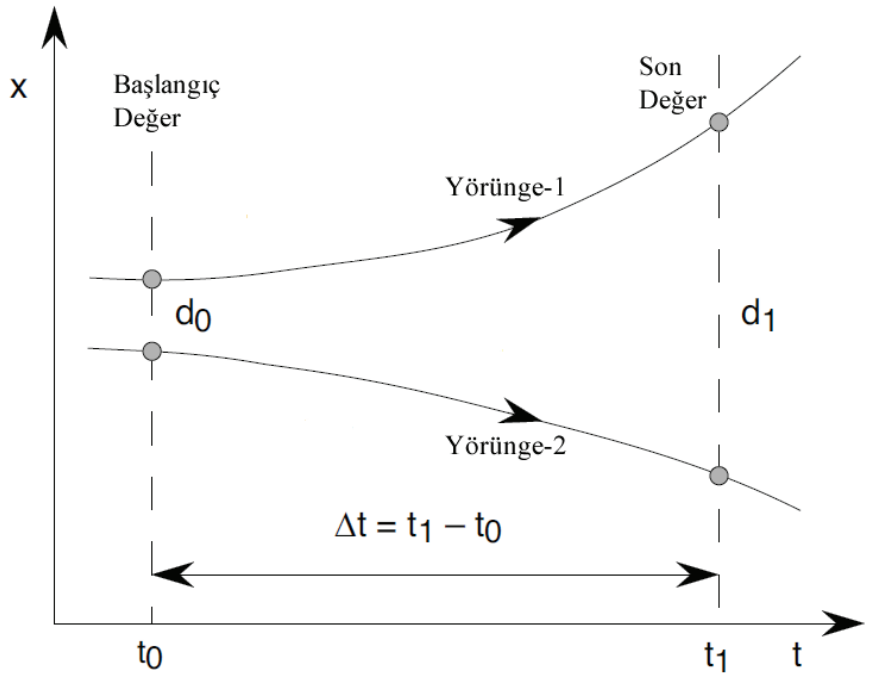
başlangıç şartlarında X durum değişkeninin aldığı değerler birbirlerine göre çizdirildiğinde sistemin çok farklı sonuçlar ürettiği gözlemlenmektedir.



Şekil 2.7. Kaotik bir sistemin başlangıç şartına hassas bağımlılığı

Dinamik bir sistemin başlangıç şartlarına hassas bağımlı olup olmadığı Lyapunov üstelleri ile belirlenir. Kaotik sistemlerde başlangıç şartlarındaki çok küçük değişiklik sistemin takip ettiği yörüngede büyük değişiklik meydana getirir. İşte Lyapunov üstelleri başlangıç şartlarındaki bu küçük değişiklik sonucu yörüngelerin birbirinden uzaklaşma oranını verir [135, 136].

Lyapunov üsteline örnek olarak Şekil 2.8. incelenebilir. Şekil 2.8.'de görüldüğü gibi kaotik bir sistemin x durum değişkeninin iki farklı başlangıç değeri sonucu aldığı değer görülmektedir. Başlangıçta t_0 anında x durum değişkeni değerleri arasındaki fark d_0 iken (Yörünge-1) sonra t_1 anında d_1 olmuştur (Yörünge-2). Bu şekilde sürekli zamanlı sistemlerde Lyapunov üsteli Denklem 2.9'da verildiği gibi hesaplanır. Ayrık zamanlı sistemlerde Lyapunov üsteli ise Denklem 2.10'da verildiği gibi hesaplanır. Denklem 2.10'da n, sürekli zamanlı sistemlerdeki zaman (t) yerine n. değeri ifade etmektedir. Yine Denklem 2.10'da $\Delta n = n_1 - n_0$'dır. [137].



Şekil 2.8. Farklı başlangıç şartında iki komşu yörüngenin birbirinden uzaklaşması [137]

$$\lambda = (1/\Delta t)\ln(d_t/d_0) \tag{2.9}$$

$$\lambda = (1/\Delta n)\ln(d_n/d_0) \tag{2.10}$$

Kaotik sistemin Lyapunov üsteli exponansiyel olarak değiştiğinden dolayı Denklem 2.9 ve 2.10 ile hesaplanan Lyapunov üsteli çok doğru sonuç vermeyecektir. Bu nedenle daha doğru bir Lyapunov üsteli hesabı için kaotik sistem belli bir zaman değerine ulaşana kadar sürekli olarak Lyapunov üstelleri hesaplanır ve bunların ortalaması alınır. Bu şekilde seçilen yakın yörüngelerin N segmentlik değeri için, ortalama Lyapunov üsteli değeri Denklem 2.11'de verildiği gibi hesaplanır. Eğer hesaplanan Lyapunov üsteli $\lambda>0$ ise sistem başlangıç şartlarına duyarlı, $\lambda<0$ ise sistem başlangıç şartlarına duyarsızdır denir [137].

$$\lambda = \lim_{N\to\infty}\frac{1}{N}\sum_{j=0}^{N\pm1}\lambda_j \tag{2.11}$$

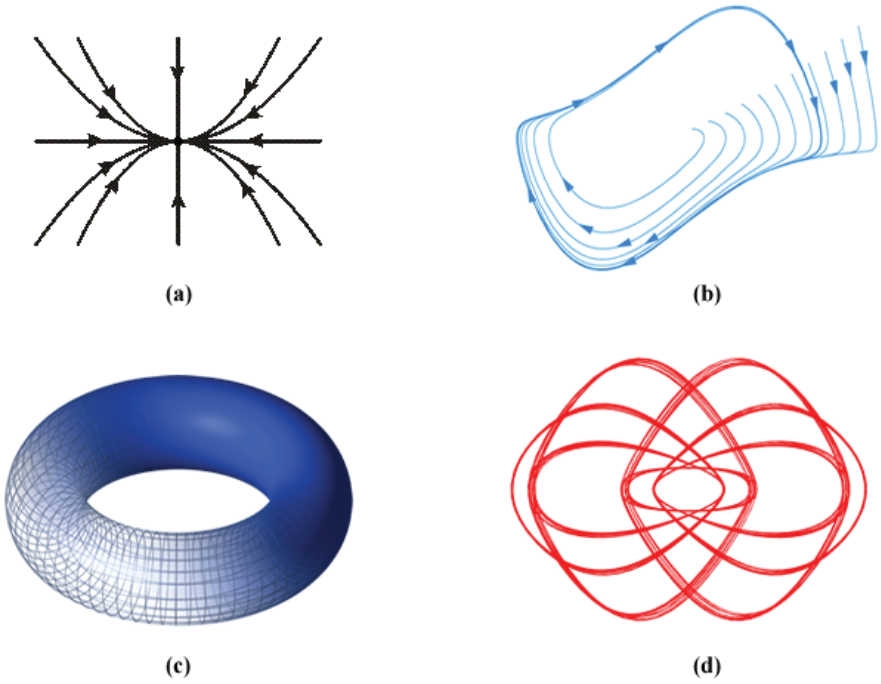
Bir sistemin Lyapunov üstelleri Denklem 2.11'de verilen formülün bir bilgisayar programı yardımıyla belli bir zaman aralığı için hesaplatılıp bulunabilir [138].

Üç boyutlu (3D) dinamik sistemlerin sahip olduğu Lyapunov üstellerine göre sisteme ait çekiciler (attractor) yani faz uzayları şekilleri aşağıda belirtilmiştir. Üç boyutlu sürekli zamanlı sistemlerde kaotik davranış için tek mümkün durum (–, 0, +) durumudur ($\lambda_1<0, \lambda_2=0, \lambda_3>0$) [139, 140].

- Lyapunov üstelleri (–, –, –) ise sistem kararlı nokta (stable node) veya odak (focus) şeklinde (Şekil 2.9a.) [141],
- Lyapunov üstelleri (–, –, 0) ise sistem kararlı limit çevrim (stable limit cycle) şeklinde (Şekil 2.9b.) [142],
- Lyapunov üstelleri (–, 0, 0) ise sistem torus şeklinde (Şekil 2.9c.) [143],
- Lyapunov üstelleri (–, 0, +) ise sistem garip çekici (strange attractor) yani kaos durumundadır (Şekil 2.9d.).

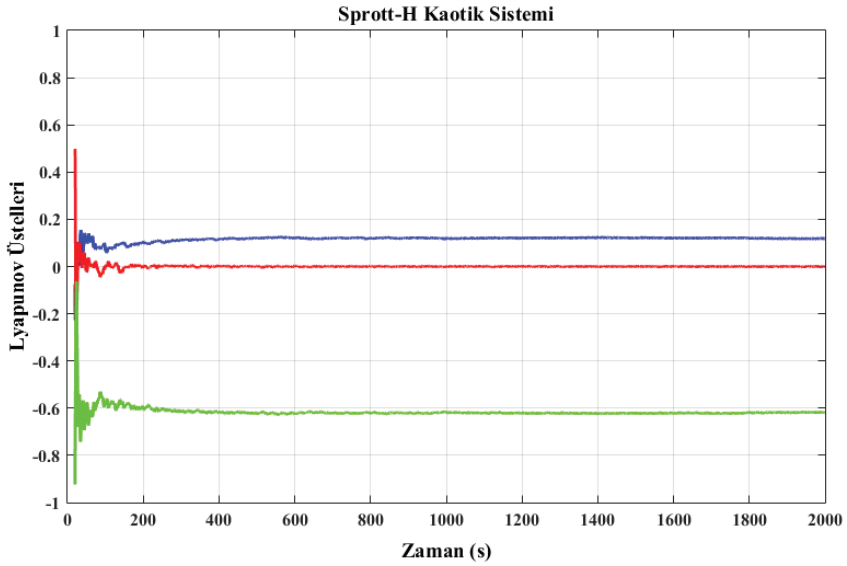
Dört boyutlu (higher-dimension) sistemlerde ise üç farklı kaos durumu mevcuttur. Aşağıda dört boyutlu sistemlerin Lyapunov üstellerine göre aldığı durumlar belirtilmiştir. Eğer sistemin birden fazla Lyapunov üsteli pozitif ise bu durum hiper-kaos (hyperchaos) olarak adlandırılır [139, 140].

- Lyapunov üstelleri (+, +, 0, –) ise sistem hiper-kaos (hyper-chaos),
- Lyapunov üstelleri (+, 0, –, –) ise sistem kaos,
- Lyapunov üstelleri (+, 0, 0, –) ise sistem torus kaos.



Şekil 2.9. Üç boyutlu dinamik sistemlerin Lyapunov üstellerine göre çekici şekilleri (faz uzayları) (a) kararlı nokta [141] (b) limit çevrim [142] (c) torus [143] (d) kaos

Örnek olarak Şekil 2.10.'da Sprott-H kaotik sisteminin hesaplanan Lyapunov üstellerinin grafiği verilmiştir. Sprott-H kaotik sisteminin Lyapunov üstelleri yaklaşık olarak $\lambda_1 = 0{,}11841$, $\lambda_2 = 0$, $\lambda_3 = -0{,}61869$ hesaplanmıştır. Buradan da görüldüğü gibi sistem bir pozitif, bir sıfır ve bir negatif değerli Lyapunov üstellerine sahip olduğundan dolayı kaotik özellik gösterir [144].



Şekil 2.10. Sprott-H kaotik sisteminin Lyapunov üstelleri grafiği [144]

2.4.6. Fraktal kavramı ve Lyapunov boyutu

Kaos analizinde kullanılan diğer bir özellikte sistemin fraktal boyuta sahip olup olmadığıdır. Fraktal boyut kavramı için fraktal tanımının anlaşılması gerekir. Bu nedenle bu kısımda ilk önce fraktal ve fraktal boyut kavramı ardından da kaotik bir sistemin varlığının ispatı için kullanılan Lyapunov boyutu incelenmiştir.

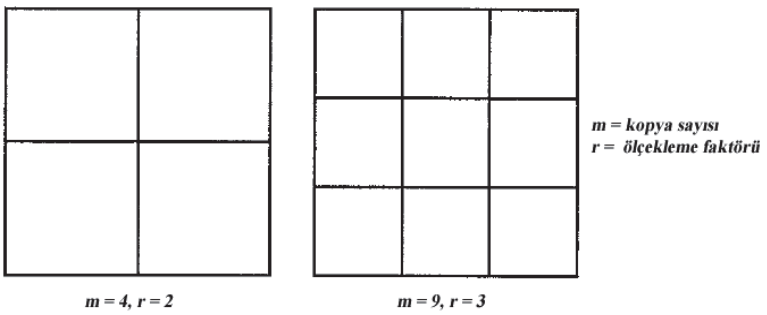
2.4.6.1. Fraktal kavramı

Fraktal terimi ilk olarak 1975'te Polonya asıllı matematikçi Benoit B. Mandelbrot (1924-2010) tarafından ortaya atılmıştır. Fraktal kelimesi, Latincede kırık taş –kırık ve şekilsiz- anlamına gelen "fractus" kelimesinden türetilmiştir. Mandelbrot, "Britanya'nın kıyı uzunluğu ne kadardır?" diye sormuştur. Britanya kıyısı metre ile ölçüldüğünde, kıyıdaki küçük girinti ve çıkıntılar metre ölçü birimiyle tam olarak ölçülmeyeceği için yaklaşık bir değer hesaplanacaktır. Eğer kıyı şeridi 10 cm'lik ölçü ile ölçülürse, hesaplanan değer metre ile ölçülen değerden daha büyük olacaktır. Çünkü 1 metre ile ölçülemeyen girinti ve çıkıntılar 10 cm'lik ölçü ile ölçülebilmektedir. Aynı şekilde kıyının uzunluğu 5 cm'lik bir ölçü birimi ile hesaplandığında hesaplanan sonuç daha da büyük olacaktır. Çünkü bu sefer de 10 cm'lik ölçü birimi ile ölçülemeyen çok küçük girinti ve çıkıntılar da hesaba katılmış olacaktır. Aynı mantıkla daha küçük ölçü birimleriyle kıyının uzunluğu ölçülmeye çalışıldığında kıyı şeridinin uzunluğu ölçülemeyecek kadar büyük hesaplanacaktır. İşte bu örnekten yola çıkarak Mandelbrot, gözlemlenen nesnenin bulunulan yer ve ölçü birimine göre değiştiğini belirtmiştir. Örneğin bir futbol topuna uzaktan bakıldığında iki boyutlu bir daire olarak, yakınlaşıldığında ise üç boyutlu bir küre olarak görülür. Mandelbrot, kıyı şeridi örneğine benzer problemlerin çözümü için tam boyutlu alandan "fraktal boyut" ismini verdiği alana geçilmesi gerektiğini iddia etmiştir. Fraktaller kesin bir tanımlaması olmayan şekiller için kullanılır. Fraktallerin kendine benzeme özelliği vardır. Küçülen ölçeklerde yinelenen fraktaller fraktalın bütününe benzer. Bazı fraktal sistemlerde şekilsel olarak birbirine benzeme görünmese de istatiksel yönden kendine benzerdir [145].

Doğada var olan neredeyse tüm nesnelerin şekilleri fraktal özellik gösterir. Bulutlar küre şeklinde değildir, dağlar koni şeklinde değildir. Kar tanesi, ağaç dalları, insan damarlarının dağılımı gibi doğadaki tüm yapılar fraktaldır. Doğadaki bu şekiller karmaşık bir yapı gibi görünmelerine rağmen içlerinde tekrarlayan bir düzen vardır [145, 146].

Klasik geometri Yunan matematikçi Öklid'in kanunlarına dayanır. Öklid'in şekilleri üçgen, kare, daire gibi basit yapıdadır ve bu şekillerin boyutları tam sayıdır [145]. Öklid uzayında boyut, verilen bir noktanın konumunu tam olarak belirlemek için gereken minimum koordinat sayısıdır. Bu yönden bakıldığında bir doğrunun boyutu bir, bir karenin boyutu iki, bir kürenin boyutu üçtür. Bir dinamik sistemin boyutu ise, sistem dinamiğini tanımlamak için gereken durum değişkenlerinin sayısıdır. Bu örneklerdeki boyutlar tam sayı olup kaotik bir çekiciyi tanımlayamazlar. Mandelbrot, kesirli yani fraktal boyuta sahip sistemlere "fraktal" ismini vermiştir [145, 147].

Örneğin bir karenin boyutu tam sayıdır ve bu nedenle kare fraktal değildir. Bir karenin tüm kenarları 2 ile ölçeklenerek ($r=2$) küçültülürse yani tüm kenarlarının orta noktası alınıp bölünürse toplam 4 kare yani kopya elde edilir ($m=4$) (Şekil 2.11.). Boyut hesabı için Denklem 2.12'de verilen genel formül kullanılabilir. Buna göre karenin boyutu Denklem 2.13'te verildiği gibi hesaplanır ve 2 bulunur. Görüldüğü gibi karenin boyutu 2 yani tam sayı olduğundan kare fraktal değildir. Aynı mantık ile eğer karenin tüm kenarları üç eşit parçaya bölünürse ($r=3$), toplam 9 kopya elde edilir ($m=9$) (Şekil 2.11.). Karenin boyutu aynı şekilde hesaplandığında Denklem 2.14'te görüldüğü gibi boyutu yine 2 çıkar [148].



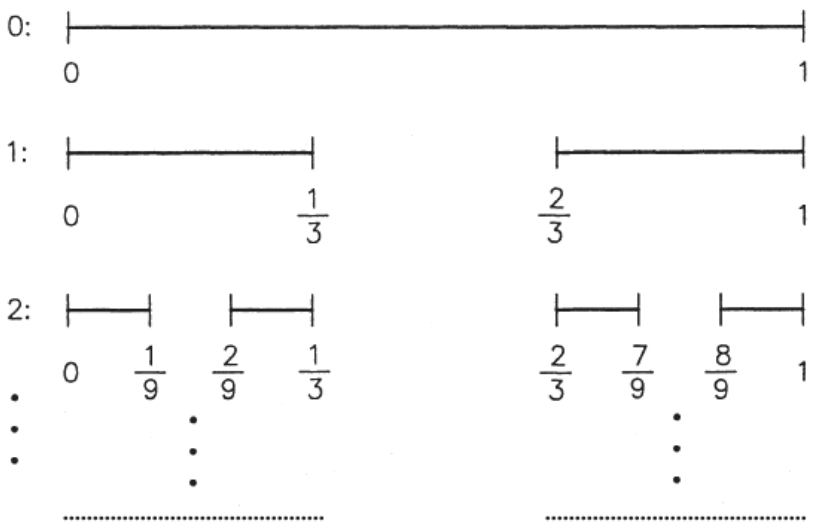
Şekil 2.11. Karenin boyutunun hesaplanması [148]

$$d = \frac{\ln m}{\ln r} = \frac{\ln \text{ kopya sayısı}}{\ln \text{ ölçekleme faktörü}} \tag{2.12}$$

$$d = \frac{\ln m}{\ln r} = \frac{\ln 4}{\ln 2} = 2 \tag{2.13}$$

$$d = \frac{\ln m}{\ln r} = \frac{\ln 9}{\ln 3} = 2 \tag{2.14}$$

Cantor kümesi fraktal yapıya örnektir. Cantor kümesini elde etmek için için şu yol izlenir. [0 1] kapalı aralığını üç eşit parçaya bölüp $(0,\frac{1}{3},\frac{2}{3},1)$, ortadaki parçayı yani $(\frac{1}{3},\frac{2}{3})$ aralığını atalım. Geriye $(0,\frac{1}{3})$ ve $(\frac{2}{3},1)$ aralığı kalır. Aynı şekilde kalan $(0,\frac{1}{3})$ ve $(\frac{2}{3},1)$ parçalarını da üç eşit parçaya bölersek sırayla $(0,\frac{1}{9},\frac{2}{9},\frac{1}{3})$ ve $(\frac{2}{3},\frac{7}{9},\frac{8}{9},1)$ elde edilir. Bunlardan da ortadaki parça atılırsa sırayla $(0,\frac{1}{9})$ ile $(\frac{2}{9},\frac{1}{9})$ parçası ve $(\frac{2}{3},\frac{7}{9})$ ile $(\frac{8}{9},1)$ parçası kalır. Bu işlem bu şekilde sonsuza kadar yapılabilir. Açıklanan metot ile elde edilen Cantor kümesi Şekil 2.12.'de verilmiştir [8, 127,148].



Şekil 2.12. Cantor kümesi [8]

İşlem sonsuza kadar sürdürüldüğünde her işlem sonucu elde edilen eleman sayısı sonsuza, elemanların uzunlukları ise sıfıra yaklaşmaktadır. Bir noktanın boyutu sıfırdır. Bir doğrunun boyutu ise birdir. Cantor kümesi işlem sayısı sonsuza gittikçe bir uzunluğu olmadığından boyutu bir değil ama birden küçük olur. İşlem sayısı sonsuza gittiğinde parçalar nokta haline benzer ama çok çok sayıda nokta oluştuğundan boyutunun nokta gibi sıfır olduğu da söylenemez. İşte Cantor kümesinin boyutu bu nedenle tam sayı değil fraktaldır yani kesirlidir [127].

Cantor kümesinde fraktallarda olan kendi kendine benzerlik özelliği de görülmektedir. (n+1). iterasyondaki bir doğru parçası, n. iterasyondaki doğru parçasının 1/3 oranında küçültülmüş benzeridir. Cantor kümesinde doğru parçası *r=3* eşit kısma bölünüp ortadaki bir parça atıldığında, *m=2* kopya alınmış olur, bu durumda Cantor kümesinin fraktal boyutu Denklem 2.15'teki gibi hesaplanır [147, 148].

$$d = \frac{\ln m}{\ln r} = \frac{\ln 2}{\ln 3} \cong 0.6309 \tag{2.15}$$

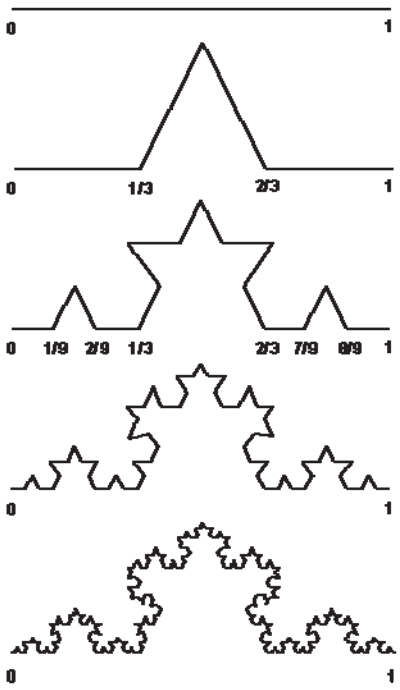
Fraktallara başka bir örnek olarak da Sierpinski üçgenleri verilebilir. Sierpinski üçgenlerini elde etmek için, bir üçgen alınır ve bu üçgenin kenarlarının orta noktaları birleştirilerek dört adet eş üçgen elde edilir. Merkezdeki üçgen atılır ve kalan üçgenlere aynı işlem uygulanarak Şekil 2.13.'te verilen Sierpinski üçgenleri elde edilir. Sierpinski üçgenlerinin de boyutu Denklem 2.16'da verildiği gibi hesaplanır. Sistemde üçgenin her kenarı 2'ye bölünmektedir yani 2 ile ölçeklenmektedir *r=2*. Bu işlem sonucunda da 4 adet üçgen kopya elde edilmekte ama ortadaki üçgen atıldığı için elde edilen kopya sayısı *m=3* olmaktadır. Sierpinski üçgenlerinin boyutu *d=1,58496* olmaktadır. Dolayısıyla Sierpinski üçgenleri de fraktaldır [127].

Şekil 2.13. Sierpinski üçgenleri [127]

$$d = \frac{\ln m}{\ln r} = \frac{\ln 3}{\ln 2} \cong 1,58496 \tag{2.16}$$

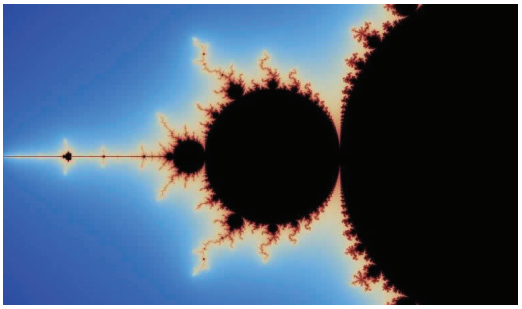
Fraktal yapıya bir başka örnekte Koch kar tanesi veya günümüzde sadece Koch eğrisi ismiyle anılan yapı verilebilir. Helge von Koch (1870-1924), sonsuz matrisler ve onların uzunlukları arasındaki ilişkinin anlaşılmasına katkı sağlayan İsveçli bir matematikçidir. Koch eğrisini elde etmek için [0, 1] aralığındaki bir doğru üç eşit parçaya bölünür ve ortadaki parça çıkartılır. Çıkartılan parça ile eş uzunluktaki bir parça boşta kalan iki kenara yerleştirilip uçları birleştirilerek bir üçgen oluşturulur (Şekil 2.14.). Bu işlem sonsuza kadar devam ettirildiğinde Koch eğrisi elde edilir. Koch eğrisinde doğru 3 eşit parçaya bölündüğünden yani 3 ile ölçeklendiğinden *r=3* olur. Ölçekleme işlemi sonrasında Şekil 2.14.'ten te görüldüğü gibi 4 adet doğru kopyası elde edildiğinden *m=4* olur. Dolayısıyla Denklem 2.17'de verildiği gibi Koch eğrisinin boyutu *d=1,2619* çıkar ve sonuç kesirli olduğundan Koch eğrisi fraktal bir yapıdır [149].

$$d = \frac{\ln m}{\ln r} = \frac{\ln 4}{\ln 3} \cong 1,2619 \tag{2.17}$$



Şekil 2.14. Koch eğrisi [149]

Aşağıda Şekil 2.15.'te fraktal Mandelbrot kümesi [150], Şekil 2.16.'da ise fraktal yapıda olan bir selvi ağacı dalı da [151] fraktal yapılara örnek olarak verilmiştir.



Şekil 2.15. Fraktal Mandelbrot kümesi [150]

Şekil 2.16. Doğada bulunan fraktal yapıdaki selvi ağacı dalı [151]

2.4.6.2. Lyapunov boyutu

Bölüm 2.4.6.1.'de fraktal ve fraktal boyut kavramları açıklanarak kaos yapısının fraktal özelliği vurgulanmıştır. Dinamik bir sistemde de kaosun varlığı araştırılırken sistemin boyutunun fraktal yani kesirli olup olmadığına bakılır. Dinamik sistemlerde sistem boyutunun pratik olarak tespiti için Kaplan ve Yorke tarafından sunulan metot ile Lyapunov üstelleri kullanılarak Lyapunov boyutu hesaplanır. Eğer sistemin Lyapunov boyutu kesirli yani fraktal çıkarsa sistemin kaotik özellik gösterdiği söylenebilir. Denklem 2.18'de Lyapunov boyutu hesaplanması verilmiştir. Denklem 2.18'deki j terimi $\lambda_1 + \lambda_2 + ... + \lambda_j > 0$ koşulunu sağlayan en büyük tamsayıyı ifade eder [140, 152, 153].

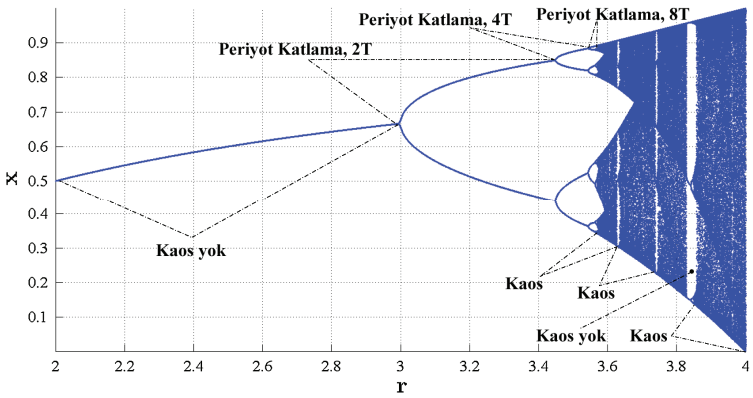
$$D_L = j + \frac{\sum_{i=1}^{j} \lambda_{L_i}}{\left|\lambda_{L_{J+1}}\right|} \tag{2.18}$$

Örnek olarak Sprott-H kaotik sisteminin hesaplanan Lyapunov üstelleri $\lambda_1 = 0{,}11841$, $\lambda_2 = 0$, $\lambda_3 = -0{,}61869$'dur. Buradan $\lambda_1 + \lambda_2 > 0 \Rightarrow 0{,}11841 + 0 > 0 \Rightarrow 0{,}11841 > 0$ olduğundan Denklem 2.18'deki j değeri *j=2* olacaktır. Buna göre Denklem 2.18'in çözümü Denklem 2.19'da verildiği gibi olur. Sistemin Lyapunov boyutu fraktal çıktığından sistemin kaotik davranış gösterdiğinden söz edilebilir.

$$D_L = j + \frac{\sum_{i=1}^{j} \lambda_{L_i}}{\left|\lambda_{L_{J+1}}\right|} = 2 + \frac{\sum_{i=1}^{2} (\lambda_{L_1} + \lambda_{L_2})}{\left|\lambda_{L_{2+1}}\right|} = 2 + \frac{\sum_{i=1}^{2} (\lambda_{L_1} + \lambda_{L_2})}{\left|\lambda_{L_3}\right|} = 2{,}1914 \qquad (2.19)$$

2.4.7. Çatallaşma diyagramı

Dinamik bir sistemin akışı yani çekici davranışı sistemin parametre değerlerine göre değişir. Dinamik sistemdeki bu değişiklik *çatallaşma* olarak isimlendirilir. Dinamik sistemin bir parametresinin alacağı değişik değerlere göre sistem durum değişkeninin aldığı değerlerin birbirlerine göre çizdirilmesi ile elde edilen grafik *çatallaşma diyagramı* olarak tanımlanır. Kaotik sistemlerin çatallaşma diyagramında *periyot katlama* olayı görülür. Çatallaşma diyagramında 2T, 4T, 8T… gibi periyot katlama olayları görülür ve sistem parametrenin bazı değerlerinde kaosa girer. Parametrenin sistemi kaosa götürdüğü bu değer aralığına kritik değer denir. Şekil 2.17.'de lojistik harita kaotik sisteminin çatallaşma diyagramı verilmiştir. Diyagramda *x*, sistem durum değişkenini, *r* ise sistem parametresini ifade etmektedir. Diyagramda sistemin kaosa girmediği ve girdiği alanlar belirtilmiştir [2, 148].



Şekil 2.17. Lojistik haritanın çatallaşma diyagramı

BÖLÜM 3. KAOTİK HABERLEŞME SİSTEMLERİ

Bu bölümde kaotik haberleşme sistemlerin genel olarak gelişimi ile kaos tabanlı sayısal (dijital) haberleşme yöntemleri incelenmiştir.

3.1. Kaotik Haberleşme Sistemlerinin Gelişimi

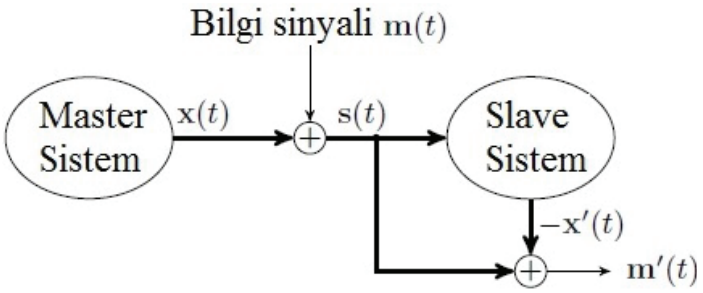
Son yirmi yılda haberleşme alanı için kaos tabanlı bir dizi modülasyon ve demodülasyon yöntemi önerilmiştir. Geleneksel ve kaos tabanlı (kaotik) haberleşme sistemleri taşınan bilginin çeşidine göre *analog* ve *sayısal (dijital)* haberleşme olarak iki grupta incelenir [135]. Kaotik haberleşme sistemleri temel yapı olarak incelendiğinde dört nesil altında gelişim göstermiştir. Bunlar [135, 154, 155];

- I. Nesil kaotik haberleşme sistemleri (Kaotik maskeleme veya Ekleyici kaos maskeleme)
- II. Nesil kaotik haberleşme sistemleri (Kaotik modülasyon)
- III. Nesil kaotik haberleşme sistemleri (Kaotik şifrelemeli haberleşme)
- IV. Nesil kaotik haberleşme sistemleri

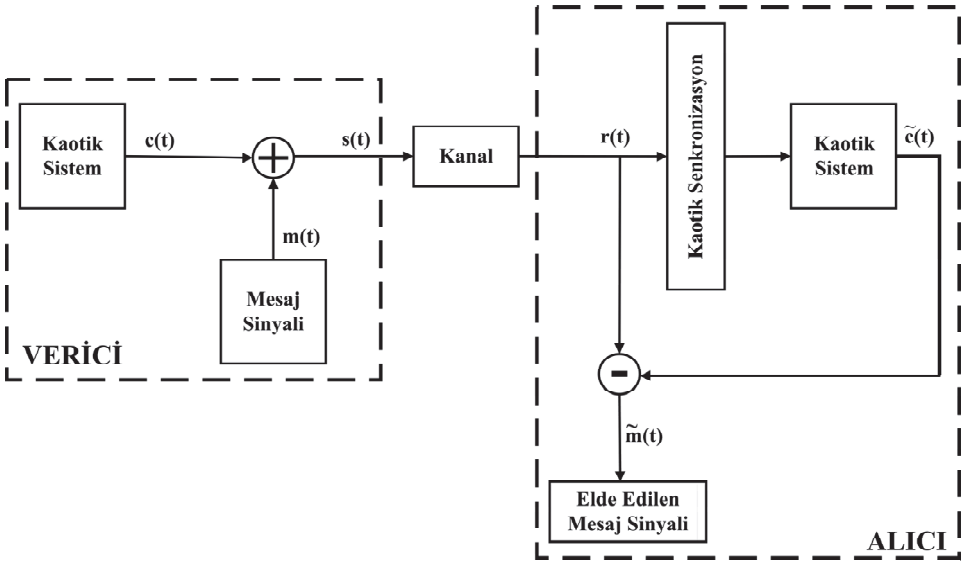
3.1.1. I. nesil kaotik haberleşme sistemleri

Birinci nesil kaotik haberleşme sistemleri, 1993 yılında geliştirilmiş olup iki farklı yöntemi içerir. Bunlardan biri *"ekleyici kaos maskeleme (additive chaotic masking)"* veya *"kaotik maskeleme (chaotic masking)"* diğeri ise *"Kaotik/Kaos anahtarlama (chaotic/chaos switching)"* veya *"Kaotik/Kaos kaydırmalı anahtarlama (chaotic/chaos shift keying)"* olarak tanımlanmaktadır [154]. Kaotik maskeleme yönteminde, gönderici kısımda kaotik sinyale gönderilecek bilgi sinyali *m(t)* eklenir. Alıcı kısımda ise verici kısımdaki kaotik sinyal kaotik senkronizasyon yöntemiyle

yeniden oluşturulur ve gelen sinyalden bu oluşturulan kaotik sinyal çıkartılarak bilgi sinyali $\tilde{m}(t)$ elde edilir. Şekil 3.1.'de kaotik maskeleme yönteminin temel yapısı Şekil 3.2.'de ise ayrıntılı yapısı verilmiştir. Kaotik maskeleme yönteminin başarılı olabilmesi için senkronizasyonun güçlü olması ve gönderilen analog bilgi sinyalinin genliğinin kaotik taşıyıcı sinyal genliğinden 20-30db kadar düşük olması gerekmektedir. Bu yöntem kanal gürültüsüne çok duyarlıdır ve güvenlik derecesi düşüktür [135, 154, 155].



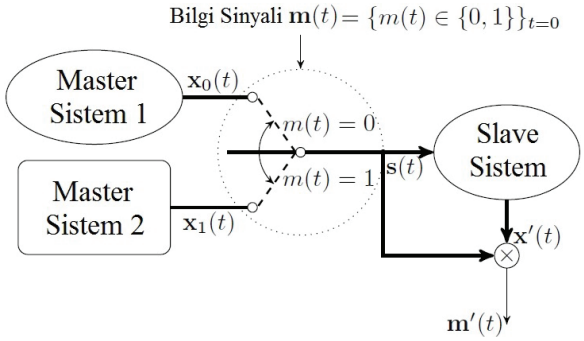
Şekil 3.1. Kaotik maskeleme haberleşme sistemlerinin temel yapısı (I. Nesil) [155]



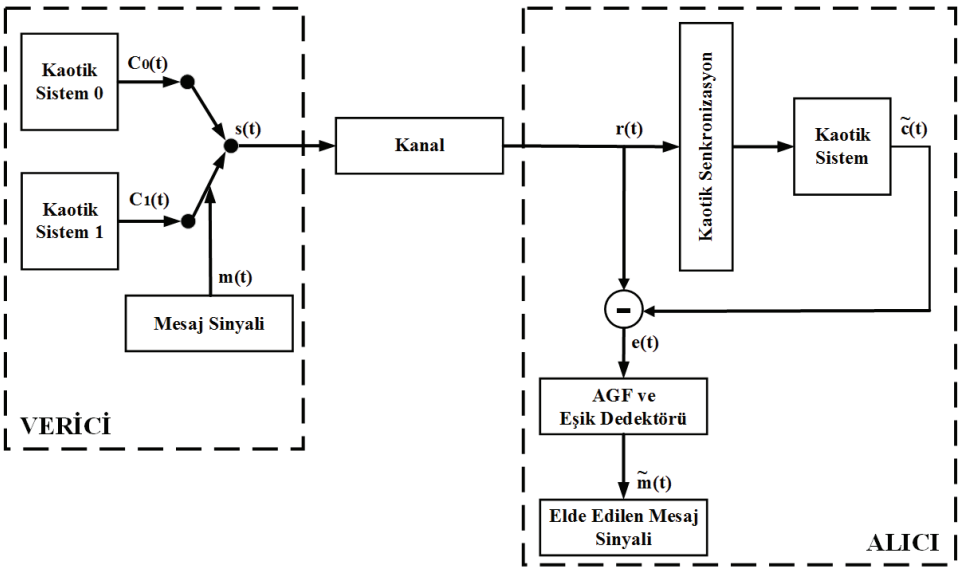
Şekil 3.2. Kaotik maskeleme haberleşme sistemlerinin ayrıntılı yapısı (I. Nesil) [154]

Kaos kaydırmalı anahtarlama yöntemi dijital sinyalleri göndermek için kullanılır. Gönderilecek mesajın lojik-0 ve lojik-1 bilgisi için verici tarafta iki farklı kaotik sistem kullanılır. Bir kaotik sistemin istenen parametre değerleri değiştirilerek de iki farklı kaotik sistem elde edilebilir. Mesaj bilgisi tarafından iki kaotik sistem arasında

lojik-0 ve lojik-1 için anahtarlama yapılır. Alıcı tarafta sadece bir tane kaotik sistem vardır. Alıcı tarafa gelen sinyal ile verici ve alıcı arasında senkronizasyon sağlanarak gönderilen mesaj elde edilir. Kaos kaydırmalı anahtarlama sisteminin temel yapısı Şekil 3.3.'te, ayrıntılı yapısı ise Şekil 3.4.'te verilmiştir. Lojik-0 ve lojik-1 için gönderilen mesaj bilgisinin her iki durumunda da verici-alıcı arasındaki senkronizasyonun sağlanması için iletim süresi yeterli bir süre uzun olmalıdır. Bu nedenle kaotik anahtarlama yöntemi kaotik maskeleme yöntemine göre daha yavaştır. Fakat buna karşın kaotik anahtarlama sisteminin en büyük avantajı sinyal gürültü oranı (S/N) düşük olsa bile gönderilen mesaj bilgisi tam olarak elde edilebilir. Vericiden gelen sinyal ile senkronize hata sinyali *e(t)*, alçak geçiren filtreden ve ardından bir eşik detektöründen geçirilir. Sinyalin gücüne göre gelen sinyalin lojik-0 veya lojik-1 olduğu tespit edilir [154, 155].



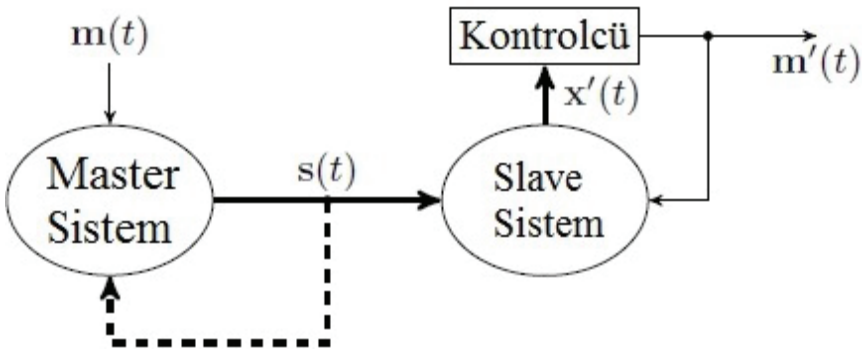
Şekil 3.3. Kaos/kaotik kaydırmalı anahtarlama haberleşme sisteminin temel yapısı (I. Nesil) [155]



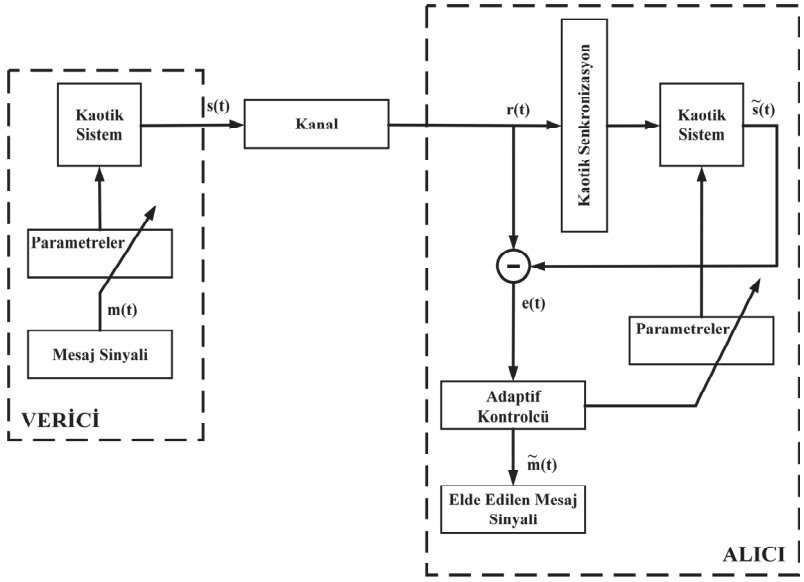
Şekil 3.4. Kaos/kaotik kaydırmalı anahtarlama haberleşme sisteminin ayrıntılı yapısı (I. Nesil) [154]

3.1.2. II. nesil kaotik haberleşme sistemleri

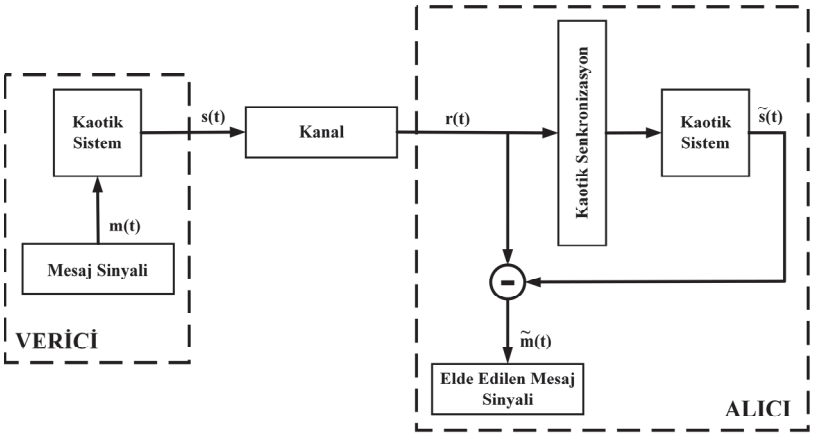
İkinci nesil kaotik haberleşme sistemleri, 1993-1995 yılları arasında geliştirilmiş olup *"kaotik modülasyon"* olarak bilinir. Kaotik modülasyon yönteminin genel yapısı Şekil 3.5.'te verilmiştir. Kaotik modülasyon iki farklı tipte gerçekleştirilir. Bunlar; *"parametre modülasyonu"* (Şekil 3.6.) ve *"direkt modülasyon* veya diğer adıyla *otonom olmayan modülasyon (non-autonomous)* (Şekil 3.7.)". Parametre modülasyonunda, bilgi sinyali *m(t)*, kaotik sistemin bir veya daha fazla kontrol parametresini modüle eder. Direkt modülasyonda (otonom olmayan modülasyonda) ise, bilgi sinyali *m(t)*, master sistemin kontrol parametrelerinin değerini değiştirmeden master sistemin bir veya daha fazla değişkenine enjekte edilir. Genelde bir adaptif kontrolcü belli kurallara göre slave sisteme eklenir. Böylece slave sistem çıkışı $\tilde{m}(t)$, asimtotik olarak gönderilen bilgi sinyaline *m(t)* yakınsar. Bu yapıdaki bazı modülasyon sistemlerinde modüleli sinyal *s(t)*, her zaman master sisteme geri besleme olarak verilmez. İkinci nesil kaotik haberleşme sistemi birinci nesile göre güvenlik düzeyi bakımından geliştirilmiştir. Ama bu yapı hala memnun edici düzeyde değildir [135, 154, 155].



Şekil 3.5. Kaotik modülasyonlu haberleşme sisteminin genel yapısı (II. Nesil) [155]



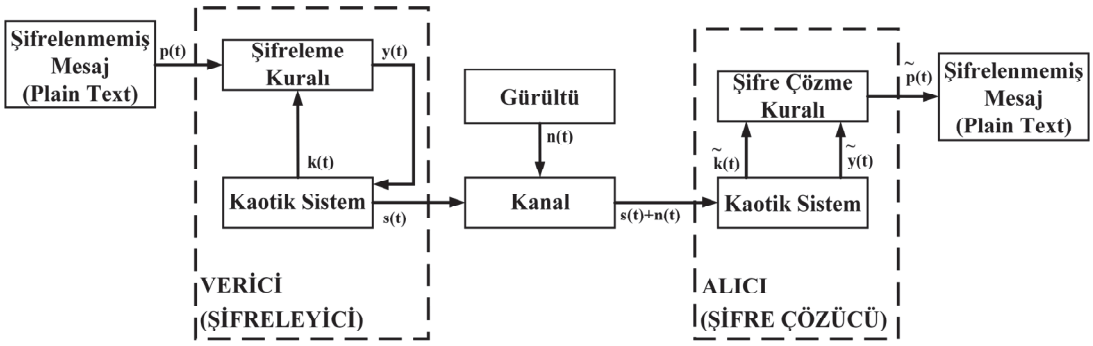
Şekil 3.6. Kaotik parametre modülasyonlu haberleşme sisteminin ayrıntılı yapısı (II. Nesil) [154]



Şekil 3.7. Kaotik direkt modülasyonlu (otonom olmayan modülasyon) haberleşme sisteminin ayrıntılı yapısı (II. Nesil) [154]

3.1.3. III. nesil kaotik haberleşme sistemleri

Üçüncü nesil kaotik haberleşme sistemleri, haberleşme sisteminin güvenlik derecesini ilk iki nesile göre daha üst düzeye getirmek için 1997 yılında tasarlanmıştır. Üçüncü nesil kaotik haberleşme sistemi *"kaotik şifreleme (chaotic cryptosystem)"* olarak isimlendirilmektedir. Bu yöntemde klasik şifreleme tekniği ile kaotik senkronizasyon birleştirilerek güvenlik düzeyi arttırılmıştır. Şekil 3.8.'de kaotik şifreleme olarak isimlendirilen ve III. nesil olan kaotik haberleşme yönteminin yapısı verilmiştir [154].

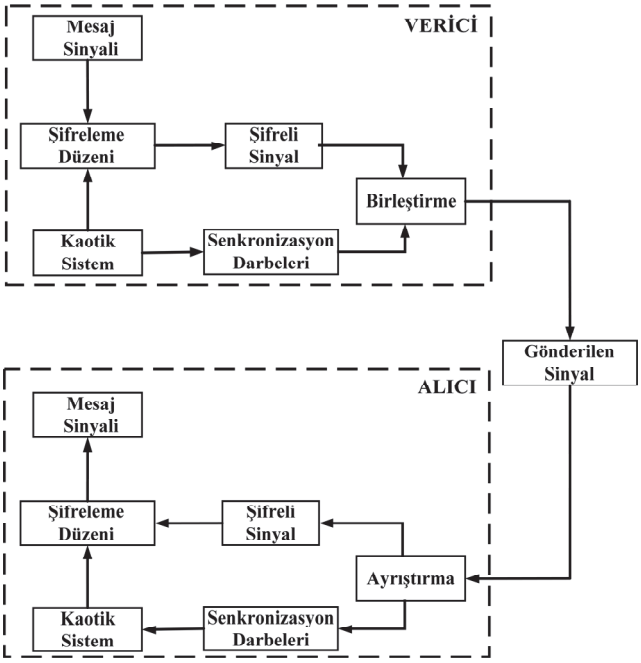


Şekil 3.8. Kaotik şifreleme yönteminin yapısı (III. Nesil) [154]

Kaotik şifreleme yönteminde şifrelenmemiş mesaj (plain text), kaotik sistem tarafından üretilen şifre sinyali *k(t)* kullanılarak şifreleme kuralı çerçevesinde şifrelenir. Şifrelenmiş bilgi *s(t)* kanal üzerinden iletilirken gürültü sinyalleri *n(t)* eklenir. Alıcı tarafa gelen gürültülü sinyal *s(t)+n(t)*, verici taraftaki kaotik sistemin senkronizasyonu için kullanılır. Kaotik sistemin senkronizasyonu sonucunda alıcı tarafta biraz gürültülü olarak $\tilde{k}(t)$ ve $\tilde{y}(t)$ sinyalleri elde edilir. Verici ile uyumlu şifre çözme kuralı ile değerlendirildikten sonra verici tarafından gönderilen şifrelenmemiş mesaj $\tilde{p}(t)$ elde edilir. Kaotik sistemin donanımını ve şifreleme algoritmasını bilmeden şifrelenmiş mesajın çözülmesi zordur [154].

3.1.4. IV. nesil kaotik haberleşme sistemleri

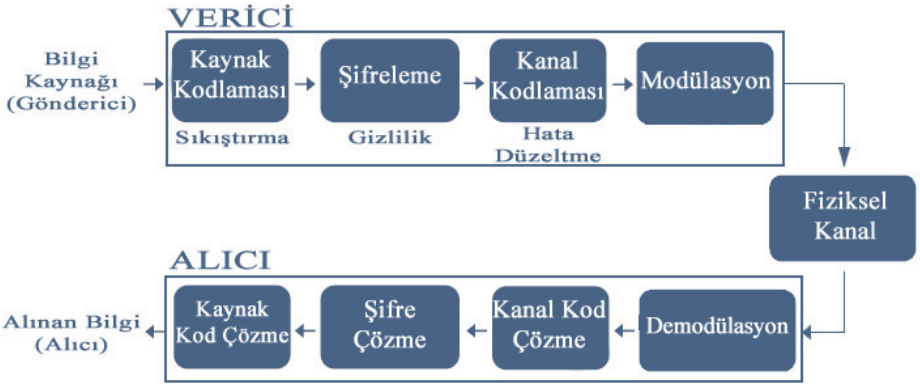
Birinci nesil kaotik haberleşme sistemlerinde geri besleme kontrol teorisine dayanan basit yapıda senkronizasyon kullanılmıştır. İkinci ve üçüncü nesil kaotik haberleşme sistemlerinde ise adaptif kontrol teorisine dayanan adaptif senkronizasyon yöntemleri kullanılmıştır. Prensip olarak ilk üç kaotik haberleşme neslinde devamlı (continuous) bir senkronizasyon durumu vardır. 1997 yılında dürtüsel (impulsive) kontrol teorisine dayanan dürtüsel (impulsive) senkronizasyon ismi ile yeni bir senkronizasyon yöntemi sunulmuştur. Dördüncü nesil kaotik haberleşme sistemleri dürtüsel senkronizasyon yöntemi ile gerçekleştirilmiştir. Dürtüsel senkronizasyon tabanlı güvenli haberleşme sisteminin blok şeması Şekil 3.9.'da verilmiştir [154].



Şekil 3.9. Dürtüsel senkronizasyon tabanlı güvenli haberleşme sistemi (IV. nesil) [154]

3.2. Kaos Tabanlı Sayısal (Dijital) Haberleşme Yöntemleri

Genel olarak bir haberleşme sisteminin verici biriminde, gönderilecek bilgi bir kaynak kodlamasından geçtikten sonra şifrelenerek belirlenen bir yöntemle modülasyon işlemine tabi tutulur. Modülasyonlu sinyal fiziksel ortamdan geçerek alıcı birime gelir. Alıcı birime gelen modülasyonlu sinyal demodülasyon işlemine tabi tutularak şifresi çözülür ve kaynak kod çözme veya eşik dedektörü işleminden sonra da verici birimden gönderilen bilgi elde edilir. Genel bir dijital haberleşme sisteminin yapısı Şekil 3.10.'da verilmiştir [156, 157].



Şekil 3.10. Genel bir haberleşme sisteminin yapısı [157]

Haberleşme sistemlerinde genelde yaygın spektrum (spread spectrum) sistemleri kullanılır. Yaygın spektrum sistemleri, bilgi sinyalinin çeşitli yöntemlerle bant genişliğini arttırıp ileten sistemlerdir. Bu sistemlerde gönderilen modülasyonlu sinyalin bant genişliği bilgi sinyalinden daha fazla olmalıdır. Aynı zamanda yaygın spektrum sistemlerinde gönderilen modülasyonlu sinyalin bant genişliği bilgi sinyalinden bağımsız bir fonksiyon tarafından belirlenmelidir. Kaotik sinyaller de genel olarak geniş bant özelliği gösterdiğinden dolayı doğası gereği yaygın spektrum haberleşmede kullanılabilir. [158].

Geleneksel sayısal haberleşme sistemlerinde gönderilecek her sembol periyodik bir sinüs sinyali ile temsil edilir. Kaos tabanlı (kaotik) sayısal haberleşme sistemlerinde ise gönderilecek her sembol kaotik sinyalin bir parçası ile temsil edilir. Bu sayede aynı sembol bilgisi tekrar tekrar gönderilse bile, kaotik sinyal periyodik olmadığından dolayı gönderilen bilgi sürekli farklı farklı olacaktır. Kaos tabanlı sayısal haberleşme sistemlerinin geleneksel sayısal haberleşme sistemlerine göre verimliliği daha fazladır, daha düşük güç harcaması vardır, tahmin edilme olasılığı daha düşüktür [135, 154-157].

Sayısal haberleşme sistemlerinin performans ölçümünde Bit Hata Oranı - BER (Bit Error Rate) değeri dikkate alınır. Geleneksel haberleşme sistemlerinin BER oranı özellikle kaos tabanlı analog haberleşme sistemlerine göre genelde daha iyidir. Bu nedenle kaos tabanlı analog haberleşme yerine kaos tabanlı sayısal haberleşme sistemleri tercih edilmektedir. Daha iyi BER oranına sahip kaos tabanlı sayısal haberleşme sistemleri için de farklı teknikler üzerinde çalışılmıştır [159, 160].

Kaos tabanlı sayısal haberleşme sistemleri iki grupta incelenir. Bunlar;

- Evre uyumlu/Eş zamanlı (coherent) kaos tabanlı sayısal haberleşme sistemleri
- Evre uyumsuz/Eş zamansız (noncoherent) kaos tabanlı sayısal haberleşme sistemleri

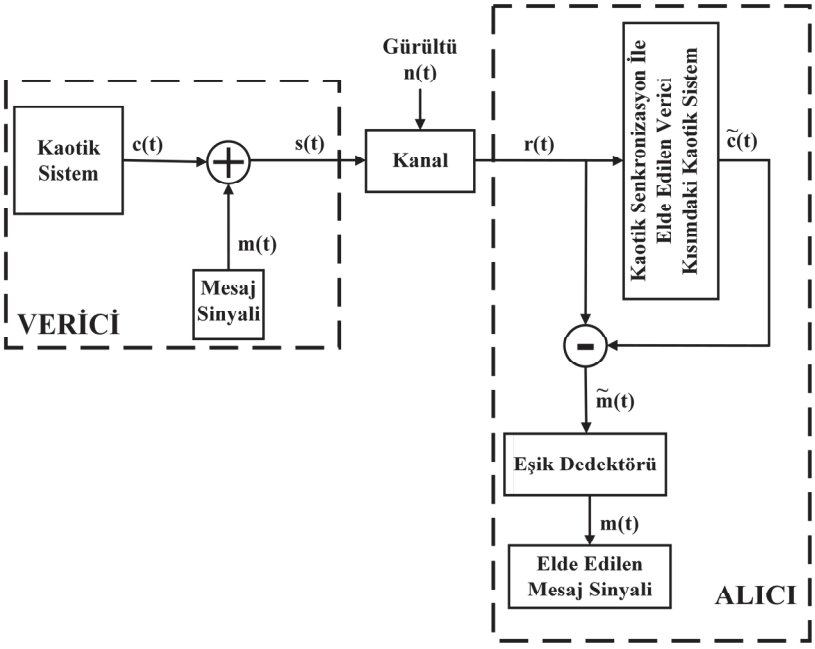
Evre uyumlu/Eş zamanlı (coherent) sistemler, alıcı tarafta verici taraftaki kaotik sistemin yeniden üretilmesini gerektirir. *Evre uyumsuz/Eş zamansız (non-coherent) sistemler* ise, alıcı tarafta verici taraftaki kaotik sistemin yeniden üretilmesine gerek olmayan sistemlerdir. Evre uyumlu/Eş zamanlı (coherent) sistemlerde alıcı tarafta gönderilen mesaj sinyalinin alınması için senkronizasyona ihtiyaç vardır. Evre uyumsuz/Eş zamansız (noncoherent) sistemlerde ise gönderilen mesaj sinyalinin alınması için senkronizasyona gerek yoktur. Bu sistemlerde alınan sinyalin bit enerjisine bakılır. Bit enerjisi belli bir eşik değere göre karşılaştırılarak gelen bilginin lojik-0 veya lojik-1 olduğuna karar verilir [135, 161, 162].

Kaos tabanlı sayısal haberleşme yöntemleri temel olarak, Kaotik Maskeleme (Chaotic Masking - CM), Kaotik Açma-Kapama Anahtarlama (Chaotic On-Off Keying – COOK), Kaos Kaydırmalı Anahtarlama (Chaos Shift Keying - CSK) olmak üzere üç çeşittir. Bu temel yöntemlerden sonra bu yöntemlere dayalı çeşitli varyasyonlar geliştirilmiştir. Kaos tabanlı sayısal haberleşme yöntemleri genel olarak aşağıda verilmiştir [75-86].

- Kaotik Maskeleme (Chaotic Masking – CM)
- Kaos Kaydırmalı Anahtarlama (Chaos Shift Keying – CSK)
- Kaotik Açma-Kapama Anahtarlama (Chaotic On-Off Keying – COOK)
- Farksal Kaos Kaydırmalı Anahtarlama (Differential Chaos Shift Keying – DCSK)
- Frekans Modülasyonlu Farksal Kaos Kaydırmalı Anahtarlama (Frequency Modulated Differential Chaos Shift Keying – FM-DCSK)
- Korelasyon Gecikmeli Kaos Kaydırmalı Anahtarlama (Correlation Delay Shift Keying – CDSK)
- Simetrik Kaos Kaydırmalı Anahtarlama (Symmetric Chaos Shift Keying – SCSK)
- Dörtlü Kaos Kaydırmalı Anahtarlama (Quadrature Chaos Shift Keying – QCSK)
- Permütasyon Tabanlı Farksal Kaos Kaydırmalı Anahtarlama (Permutation Based Differential Chaos Shift Keying – P-DCSK)

3.2.1. Kaotik maskeleme

"Kaotik Maskeleme (KM) (Chaotic masking – CM)" yöntemi hem analog hem de sayısal verilerin gönderilmesinde kullanılan bir kaos tabanlı haberleşme yöntemidir. Şekil 3.11.'de kaotik maskeleme yönteminin temel yapısı verilmiştir. Kaotik maskeleme yönteminde, verici birimde kaotik sinyale c*(t),* gönderilecek bilgi sinyali *m(t)* eklenerek modüleli sinyal *s(t)* elde edilir. Denklem 3.1'de kaotik maskeleme yönteminde verici birimden gönderilen modülasyonlu sinyalin matematiksel ifadesi verilmiştir. Alıcı birime gürültü sinyali *n(t)* eklenmiş modülasyonlu sinyal *r(t)* gelir (Denklem 3.2). Alıcı birimde ilk önce verici kısımdaki kaotik sinyal kaotik senkronizasyon yöntemiyle yeniden elde edilerek kaotik sinyal $\tilde{c}(t)$ elde edilir. Oluşturulan kaotik sistem sinyali $\tilde{c}(t)$, gelen gürültülü sinyalden çıkartılarak gürültülü bilgi sinyali $\tilde{m}(t)$ elde edilir (Denklem 3.3). Gürültülü sinyal eşik dedektöründen geçirilerek gürültüsüz şekilde bilgi sinyali $m(t)$ elde edilir. Kaotik maskeleme yönteminin başarılı olabilmesi için senkronizasyonun güçlü olması ve gönderilen analog bilgi sinyalinin genliğinin kaotik taşıyıcı sinyal genliğinden düşük olması gerekmektedir (20-30db kadar) [75-78, 135, 154, 155].



Şekil 3.11. Kaotik maskeleme haberleşme sistemlerinin temel yapısı [155]

$$s(t) = c(t) + m(t) \tag{3.1}$$

$$r(t) = s(t) + n(t) \tag{3.2}$$

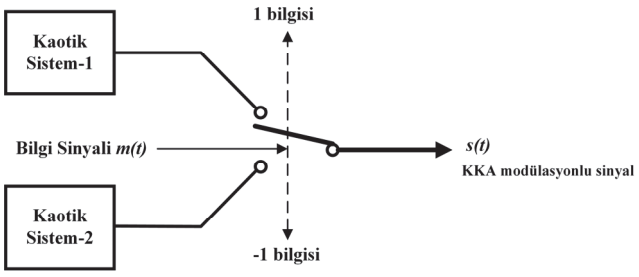
$$\tilde{m}(t) = s(t) + n(t) - x(t) = x(t) + m(t) + n(t) - x(t) = m(t) + n(t) \tag{3.3}$$

3.2.2. Kaos kaydırmalı anahtarlama

"Kaos Kaydırmalı Anahtarlama (KKA) (Chaos Shift Keying - CSK)" modülasyon tekniğinin ilk temelleri 1992 yılında Parlitz ve arkadaşları tarafından kaotik senkronizasyon ile dijital sinyallerin iletimi konulu makalesi ile ortaya atılmıştır [80, 164]. Ardından Hervé Dedieu ve arkadaşları tarafından kaos kaydırmalı anahtarlama tanımlaması kullanılarak Chua devresi ile modülasyon ve demodülasyon çalışması yapılmıştır [81]. KKA modülasyon tekniği sayısal bilgileri kaotik tabanlı sinyaller ile kodlama tekniğidir.

KKA yönteminde iki farklı kaotik sinyal kullanılır. Sayısal "1" bilgisi için bir kaotik sinyal, sayısal "-1" bilgisi için ikinci kaotik sinyal gönderilir (Denklem 3.4). İki farklı kaotik sinyal için iki farklı kaotik sistem (çeker) veya kaotik sistemin farklı durum değişkenleri veya aynı kaotik sistemde parametre değiştirilerek elde edilen farklı sinyallerde kullanılabilir [135, 161, 163]. Şekil 3.12.'de KKA modülatörünün blok şeması verilmiştir [161].

$$s(t) = \begin{cases} c_1(t), & \text{1 bilgisi için} \\ c_2(t), & \text{-1 bilgisi için} \end{cases} \tag{3.4}$$



Şekil 3.12. KKA (CSK) modülatör blok şeması [161]

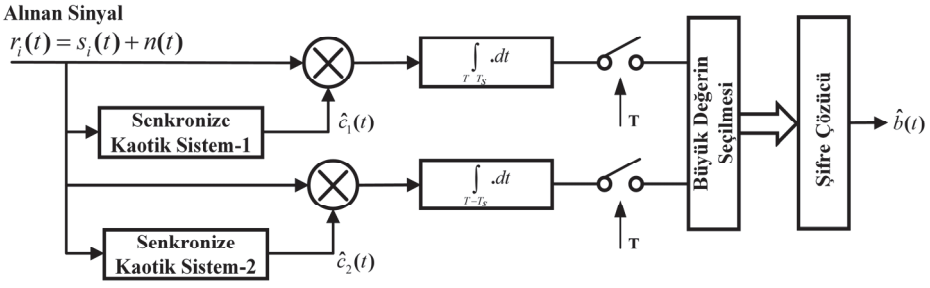
Modülatör biriminde elde edilen modülasyonlu sinyal *s(t)* iletim ortamı üzerinden alıcı birime gönderilir. Bu şekilde elde edilen modülasyonlu sinyalin *s(t)* enerjisi hesaplanarak alıcı devreye iletim ortamı üzerinden gönderilir. Gönderilen her bitin ortalama enerji değeri Denklem 3.5'te verilmiştir [163].

Alıcı tarafta alınan modülasyonlu sinyalden senkronizasyonun gerekmediği *evre uyumsuz (non-cohorent)* veya senkronizasyonun gerektiği *evre uyumlu (coherent)* demodülasyon yöntemleri ile bilgi sinyali elde edilebilir. Alıcı birimde demodülasyon sinyalinin bit enerjisi hesaplanır (Denklem 3.5). Hesaplanan bu enerji değerine göre verici birimden gönderilen bilgi tahmin edilir. Sayısal modülasyon tekniklerinde KKA (CSK) tekniği hem evre uyumlu hem evre uyumsuz olarak gerçekleştirilirken diğer dijital modülasyon teknikleri evre uyumsuz olarak gerçekleştirilir.

$$E_b = \int_{-\infty}^{\infty} x^2(t)dt \tag{3.5}$$

3.2.2.1. Evre uyumlu kaos kaydırmalı anahtarlama alıcı birimi

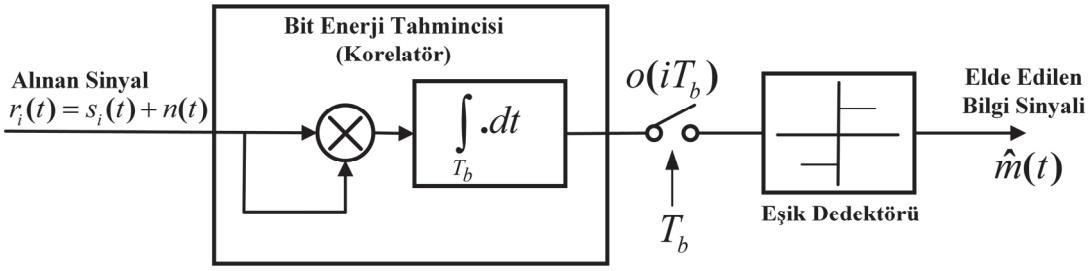
Evre uyumlu KKA sisteminde alıcı tarafta verici taraftaki kaotik sistem sinyalleri senkronizasyon yöntemi ile tekrar elde edilir. Senkronizasyon ile tekrar elde edilen kaotik sistem sinyalleri $\hat{c}_1(t)$ ve $\hat{c}_2(t)$ ve alıcıya gelen $r_i(t)$ modülasyon sinyallerinin bit enerjisi hesaplanır. Hesaplanan bu değerler korelatör devresinde değerlendirilerek bilgi sinyali elde edilir. Aşağıda Şekil 3.13.'te evre uyumlu KKA demodülatörünün blok şeması verilmiştir [161].



Şekil 3.13. Evre uyumlu KKA (CSK) demodülatör blok şeması [161]

3.2.2.2. Evre uyumsuz kaos kaydırmalı anahtarlama

Evre uyumsuz sistemde alıcı tarafta verici taraftaki kaotik sistemle herhangi bir senkronizasyona gerek yoktur. Alıcı tarafta korelatör kısmında alınan sinyalin $r_i(t)$ enerji değeri hesaplanarak eşik dedektöründe belli bir eşik değerine göre bilgi sinyali tahmin edilir. Aşağıda Şekil 3.14.'te evre uyumsuz KKA sisteminin demodülatör kısmının blok şeması verilmiştir [84, 135, 161].



Şekil 3.14. Evre uyumsuz KKA (CSK) demodülatör blok şeması [84, 135, 161]

Alınan sinyalin içinde gürültü sinyalleri de $n(t)$ vardır. Korelatör devresinde *i.* bitin enerjisi hesaplanır ve gözlem sinyali $o(iT_b)$ elde edilir. Gözlem sinyalinin değeri Denklem 3.6'da verilmiştir [135, 161, 163].

$$o(iT_b) = \int_{(i-1)T_b}^{iT_b} r_i^2(t)dt = \int_{(i-1)T_b}^{iT_b} [s_i(t) + n(t)]^2 dt$$
$$= \int_{(i-1)T_b}^{iT_b} s_i(t)^2 dt + 2\int_{(i-1)T_b}^{iT_b} s_i(t).n(t)dt + \int_{(i-1)T_b}^{iT_b} n(t)^2 dt \quad (3.6)$$

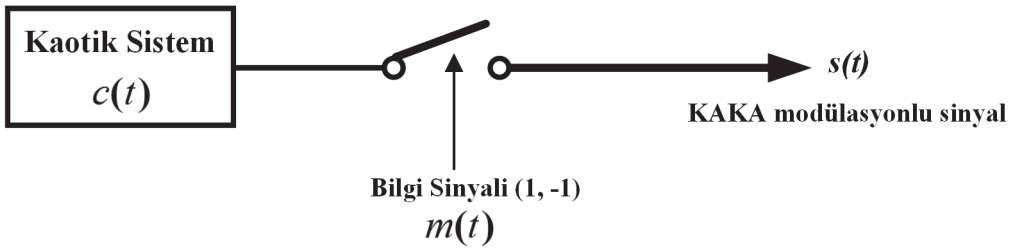
Denklem 3.6'da gürültü sıfır kabul edilirse gözlem sinyalinin son iki teriminin değeri sıfır olur ve geriye sadece *i.* bitin enerji değeri kalır ($\int_{(i-1)T_b}^{iT_b} s_i(t)^2 dt$). Bu değer "1" veya "-1" bilgisinin enerji değeridir ve eşik dedektöründe bilgi tahmini için kullanılır [161].

3.2.3. Kaotik açma-kapama anahtarlama

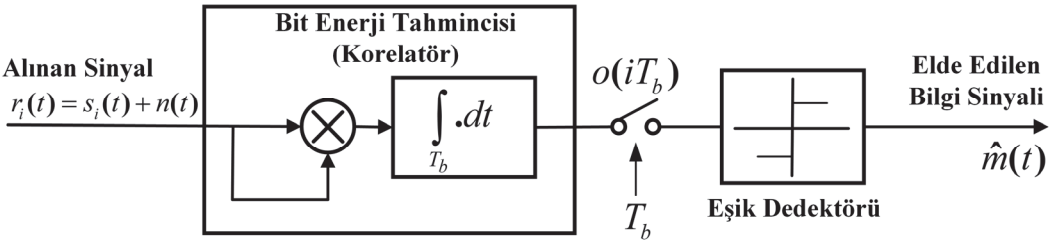
"Kaotik Açma-Kapama Anahtarlama (KAKA) (Chaotic On-Off Keying - COOK)" modülasyon yöntemi evre uyumsuz KKA yönteminin özel bir halidir ve Kolumban ve arkadaşları tarafından 1997 yılında önerilmiştir [83, 84, 135]. KAKA yönteminde "1" bilgisi için alıcıya kaotik sinyal direkt gönderilir, "-1" bilgisi için ise alıcıya hiçbir şey gönderilmez. Denklem 3.7'de bu durum gösterilmiştir. KKA yönteminin ana dezavantajlarından biri karar devresindeki eşik seviyesinin gürültüye bağlı olmasıydı. Bu durum KAKA yönteminde de görülür. Fakat KAKA yönteminde "1" ve "-1" bilgileri arasındaki enerji farkı arttırılmış olmaktadır. Bu da eşik seviye belirlemede kolaylık sağlamaktadır [164].

$$s(t) = \begin{cases} c(t), & \text{1 bilgisi için} \\ 0 \ , & \text{-1 bilgisi için} \end{cases} \tag{3.7}$$

KAKA modülatör devresinde bilgiye göre (1 veya -1) anahtarlama yapılarak kaotik sinyal demodülatöre gönderilir. Şekil 3.15.'te KAKA modülatörünün blok şeması verilmiştir. KAKA demodülatör devresi ise evre uyumsuz KKA demodülatör devresi ile aynıdır. Şekil 3.16.'da KAKA demodülatörünün blok şeması verilmiştir. Demodülatör devresinde gelen gürültülü sinyal kendisi ile çarpılarak integrali alınır yani bit enerjisi hesaplanır ve eşik dedektörü devresine iletilerek eşik seviyeye göre bilgi sinyali elde edilir. [84].



Şekil 3.15. KAKA (COOK) modülatör blok şeması [84]



Şekil 3.16. KAKA (COOK) demodülatör blok şeması [84]

Alınan sinyalin içinde gürültü sinyalleri de $n(t)$ vardır. Korelatör devresinde *i.* bitin enerjisi hesaplanır ve gözlem sinyali $o(iT_b)$ elde edilir. Gözlem sinyalinin değeri Denklem 3.8'de verilmiştir [84].

$$o(iT_b) = \int_{(i-1)T_b}^{iT_b} r_i^2(t)dt = \int_{(i-1)T_b}^{iT_b} [s_i(t) + n(t)]^2 dt$$
$$= \int_{(i-1)T_b}^{iT_b} s_i(t)^2 dt + 2\int_{(i-1)T_b}^{iT_b} s_i(t).n(t)dt + \int_{(i-1)T_b}^{iT_b} n(t)^2 dt \quad (3.8)$$

Gürültüsüz ortamda gözlem vektörü değeri "1" ve "-1" bilgileri için Denklem 3.9'daki gibi olur.

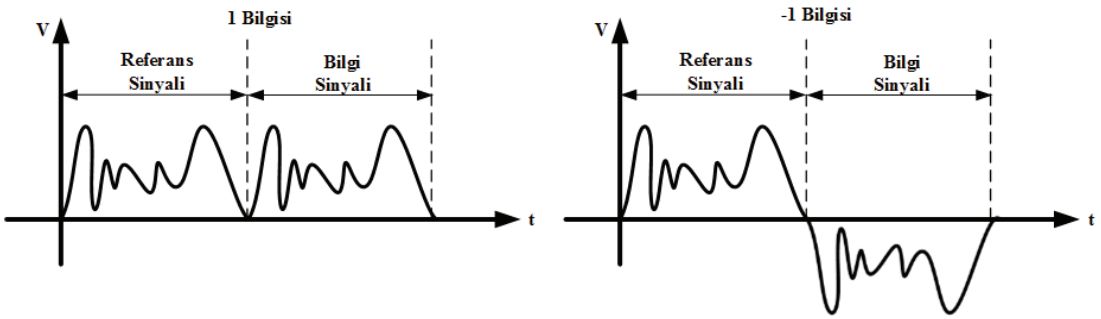
$$s(t) = \begin{cases} \int_{(i-1)T_b}^{iT_b} c^2(t)dt, \text{"1" bilgisi için} \\ 0 \quad ,\text{"-1" bilgisi için} \end{cases} \quad (3.9)$$

Karar devresinin eşik seviyesi, 0 ile $\int_{(i-1)T_b}^{iT_b} c^2(t)dt$ değerinin yarısı arasında seçilebilir. Bit enerji seviyesi bu şekilde belirlenen eşik değerin üstünde ise gelen bilgi "1", altında ise "-1" olarak algılanır.

3.2.4. Farksal kaos kaydırmalı anahtarlama

KKA yönteminde karar devresinin eşik seviyesi değeri gürültüye karşı duyarlıdır ve eşik seviyesini belirlemek zor olabilmektedir. Bu nedenle iletişimde birçok hata meydana gelebilir. KKA yönteminin bu durumuna alternatif bir çözüm olarak Kolumban ve arkadaşları "Farksal Kaos Kaydırmalı Anahtarlama (FKKA) (Diferential Chaos Shift Keying – DCSK)" modülasyon yöntemini önermiştir [84, 135, 165].

FKKA modülasyon yönteminde her bit bilgisi iki kaotik sinyal ile temsil edilir. İlk sinyal *referans sinyali*, ikinci sinyal ise *bilgi sinyalini* temsil etmektedir. "1" bilgisi için bit periyodunun yarısında referans kaotik sinyal ve diğer yarısında da yine aynı referans kaotik sinyal gönderilir. "-1" bilgisi için ise bit periyodunun yarısında referans sinyal diğer yarısında ise referans sinyalinin tersi gönderilir [84, 161]. FKKA modülasyonu işleminde "1" ve "-1" bilgisi için gönderilen sinyaller Şekil 3.17.'de grafiksel olarak [163] ve Denklem 3.10'da ise matematiksel olarak verilmiştir. Denklem 3.10'da $c(t)$ kaotik referans sinyalini, T_b bit süresini, t ise zamanı ifade etmektedir [84, 135, 161].



Şekil 3.17. FKKA (DCSK) yönteminde "1" ve "-1" sinyalleri [163]

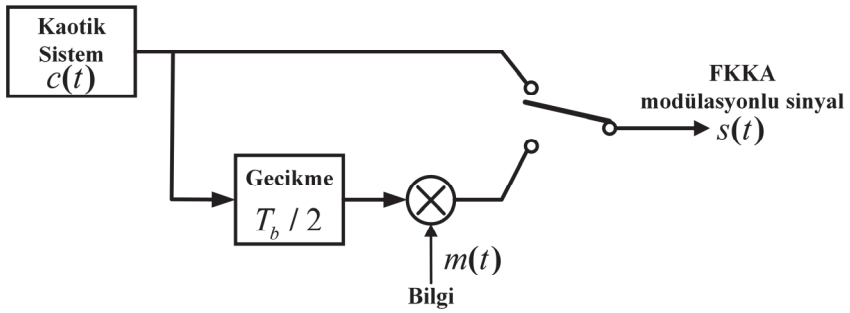
"1" için

$$s(t)=\begin{cases} c(t) & ,(i-1)T_b \le t < (i-1/2)T_b \\ c(t-T_b/2) & ,(i-1/2)T_b \le t < iT_b \end{cases} \tag{3.10}$$

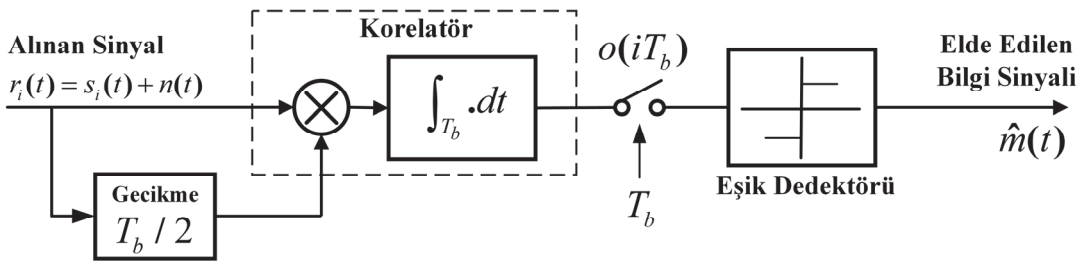
"-1" için

$$s(t)=\begin{cases} c(t) & ,(i-1)T_b \le t < (i-1/2)T_b \\ -c(t-T_b/2) & ,(i-1/2)T_b \le t < iT_b \end{cases}$$

Şekil 3.18.'de FKKA modülatörünün blok şeması verilmiştir [84, 135, 161, 163]. Modülatör devresinde kaotik sistemden gelen referans sinyali eğer gönderilecek bilgi "1" ise iki kere, eğer "-1" ise referans sinyalinin ardından referans sinyali bit süresinin yarısı kadar geciktirilip -1 bilgisi ile çarpılarak gönderilir. FKKA demodülatör kısmında ise Şekil 3.19.'da görüldüğü gibi gelen gürültülü modülasyonlu sinyal ve onun bit süresinin yarısı kadar geciktirilmiş hali çarpılarak korelatör devresine girer. Korelatör devresinde sinyalin enerjisi hesaplanarak eşik dedektöründe belirlenen değere göre bilgi sinyali elde edilir [84, 135, 161, 163, 164].



Şekil 3.18. FKKA (DCSK) modülatör blok şeması [84, 135, 161, 163]



Şekil 3.19. FKKA (DCSK) demodülatör blok şeması [84, 135, 161, 163, 164]

FKKA demodülatöründe gözlem sinyalinin $o(iT_b)$ matematiksel ifadesi i. bit için Denklem 3.11'de verilmiştir [84].

$$\begin{aligned} o(iT_b) &= \int_{(i-1/2)T_b}^{iT_b} r_i(t).r_i(t-T_b/2)dt \\ &= \int_{(i-1/2)T_b}^{iT_b} [s_i(t)+n(t)].[s_i(t-T_b/2)+n(t-T_b/2)]dt \\ &= \int_{(i-1/2)T_b}^{iT_b} s_i(t).s_i(t-T_b/2)dt + \int_{(i-1/2)T_b}^{iT_b} s_i(t).n(t-T_b/2)dt \\ &+ \int_{(i-1/2)T_b}^{iT_b} n(t).s_i(t-T_b/2)dt + \int_{(i-1/2)T_b}^{iT_b} n(t).n(t-T_b/2)dt \end{aligned} \tag{3.11}$$

Gürültüsüz ortamda gürültü terimlerini içeren ifadeler sıfır olacak ve gözlem sinyali $o(iT_b)$ Denklem 3.12'ye eşit olacaktır. Denklem 3.12'den görüldüğü gibi $s_i(t)^2 > 0$ olduğunda "1" bilgisi gönderilmiş, $s_i(t)^2 < 0$ olduğunda ise "-1" bilgisi gönderilmiş demektir. Böylece FKKA eşik dedektörünün eşik seviyesini belirlemek kolay olacaktır. Eşik seviyesi sıfır seçilerek bilgi tahmin edilebilir. Fakat gürültülü ortamda bu değer değişebilir [84]. Bit örnek süresi T_b, çok küçük olduğunda FKKA performansı yeterli olamamaktadır. Yeterince büyük bit süresi T_b için FKKA performansı geleneksel sinüzoid tabanlı modülasyon yöntemleri ile karşılaştırılabilir seviyeye gelebilir [84, 161, 163].

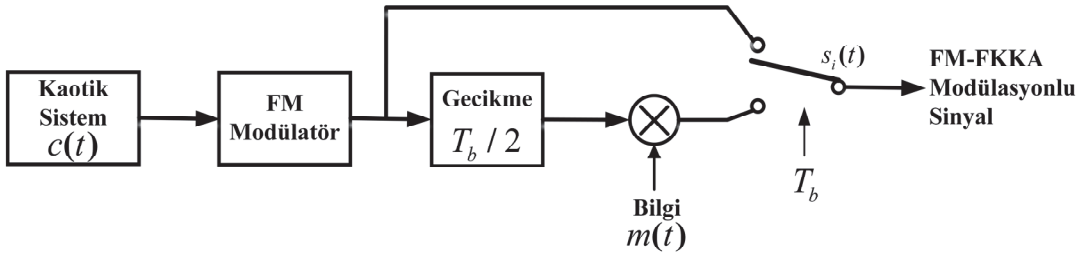
$$o(iT_b) = \int_{(i-1/2)T_b}^{iT_b} s_i(t).s_i(t-T_b/2)dt = \begin{cases} \int_{(i-1)T_b}^{(i-1/2)T_b} c^2(t)dt, \text{"1"bilgisi} \\ -\int_{(i-1)T_b}^{(i-1/2)T_b} c^2(t)dt, \text{"-1"bilgisi} \end{cases} \tag{3.12}$$

3.2.5. Frekans modülasyonlu farksal kaos kaydırmalı anahtarlama

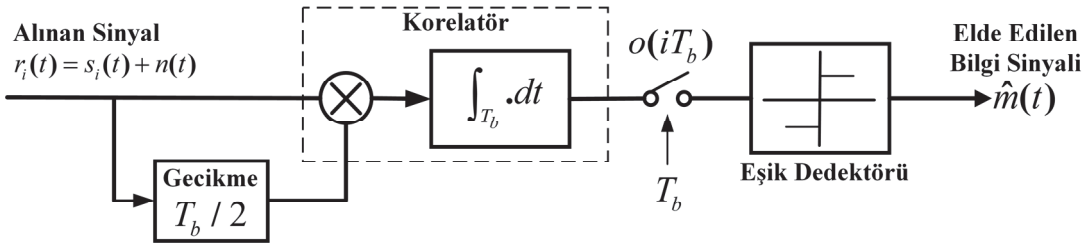
Periyodik olmayan kaotik sistemlerin doğası gereği zamanla değişen bit enerjileri söz konusudur. Gürültüsüz ortamda bile bit enerjisi değişmektedir. Bu da FKKA alıcısındaki eşik dedektörünün seviyesinin seçimini zorlaştırmakta ve gelen bilginin doğru tahminini güçleştirebilmektedir. Kolumban ve arkadaşları bu durumun gelen bilginin bit enerji seviyesinin sabit olması ile düzeltilebileceğini belirtmiş ve "Frekans Modülasyonlu Farksal Kaos Kaydırmalı Anahtarlama (FM-FKKA) (Frequency Modulated Differential Chaos Shift Keying – FM-DCSK)" yöntemini önermiştir [82].

FM-FKKA yönteminde kaotik sinyal bir FM modülatörünün girişine uygulanır. FM modülatöründe taşıyıcı sinyalin değeri belli olduğundan kaotik sinyalin enerjisi sabit tutulmuş olur. Bu sayede gürültü değerinden bağımsız olarak FM-FKKA eşik dedektörü seviyesi sıfır seçilebilir [82, 161]. Gürültü bağışıklığı yüksek olsa da FM-FKKA yönteminde gürültünün değerine göre eşik seviyesini sıfır seçmek her zaman doğru bilgi tahmini sağlamayabilir [135].

FM-FKKA modülasyon ve demodülasyon işlemi, frekans modülasyonu işlemi hariç FKKA yöntemiyle aynıdır. Şekil 3.20.'de FM-FKKA modülatör ve Şekil 3.21.'de demodülatör blok şeması verilmiştir. FM-FKKA yönteminin diğer yöntemlere göre asıl üstünlüğü veri oranının kaotik sinyalin özelliği ile sınırlı olmamasıdır.



Şekil 3.20. FM-FKKA (FM-DCSK) modülatör blok şeması [82, 163]



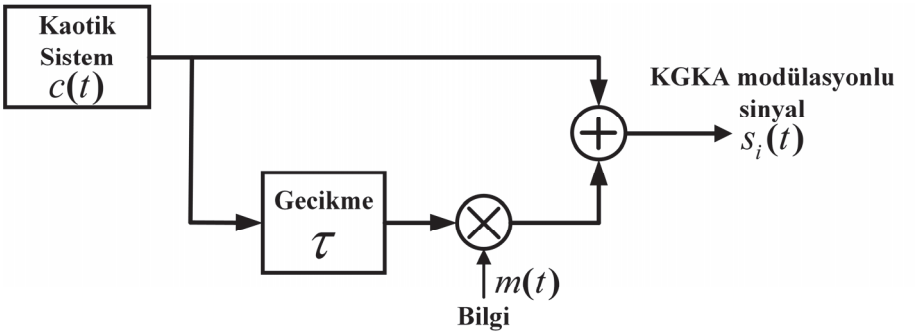
Şekil 3.21. FM-FKKA (FM-DCSK) demodülatör blok şeması [82, 163]

3.2.6. Korelasyon gecikmeli kaydırmalı anahtarlama

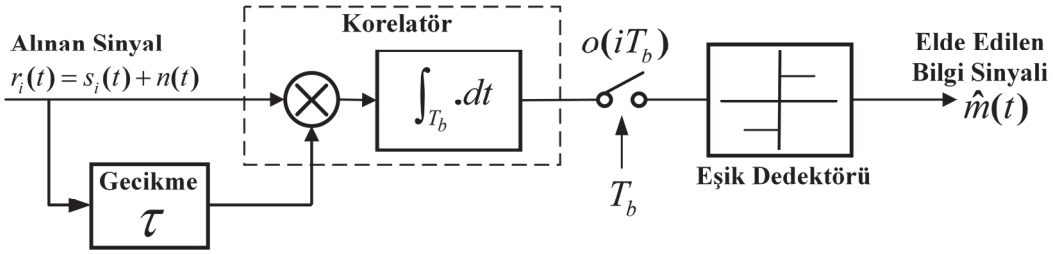
"Korelasyon Gecikmeli Kaydırmalı Anahtarlama (KGKA) (Correlation Delay Shift Keying – CDSK)" modülasyon yöntemi FKKA yönteminin bir türevi olarak görülebilir. Sushchic ve arkadaşları KGKA yöntemini 2000 yılında önermiştir [83]. KGKA yöntemi FKKA yönteminin bir türevi olmakla beraber aşağıda belirtilen farkları vardır [83, 84, 135, 163].

- FKKA yönteminde bit periyodunun yarısında referans sinyali gönderiliyor diğer yarısında ise bilgi sinyali gönderiliyordu. Bu nedenle periyodun yarısında bilgi sinyali hiç taşınmıyordu. KGKA yönteminde ise kaotik sinyal ve bu kaotik sinyalin belirlenen bir süre geciktirilmiş halinin bilgi sinyali ile çarpılmış hali toplanıp o şekilde iletilir. Bu nedenle gönderilen modüleli sinyalde her zaman bilgi sinyali mevcuttur. Bu da bant genişliğinin etkinliğinin artırılmasını sağlamaktadır.
- FKKA'da verilen gecikme bit süresinin yarısı kadardır. Hâlbuki KGKA'da verilen gecikme belirlenen herhangi bir değer de olabilir.
- FKKA yönteminde "1" bilgisi için aynı iki referans sinyal gönderilmekteydi. KGKA yönteminde ise kaotik sinyal ve bu kaotik sinyalin belirlenen bir süre geciktirilmiş halinin bilgi sinyali ile çarpılmış hali toplanıp gönderildiği için gönderilen sinyalde tekrarlama söz konusu değildir.
- KGKA yönteminde FKKA'da bulunan anahtar yerine toplayıcı bulunmaktadır.

Şekil 3.22.'de KGKA modülatör ve Şekil 3.23.'te ise demodülatör blok şemaları verilmiştir [83, 84, 135].



Şekil 3.22. KGKA (CDSK) modülatör blok şeması [83, 84, 135]



Şekil 3.23. KGKA (CDSK) demodülatör blok şeması [83, 84, 135]

KGKA modülatöründe gönderilen modülasyonlu sinyal Denklem 3.13'e eşit olur [84]. Denklemde τ, belirlenen gecikmeyi, $c(t)$ referans sinyali, $\pm c(t-\tau)$ ise taşıyıcı sinyali ifade etmektedir. KGKA alıcısında gürültü sinyali $n(t)$ eklenmiş olarak korelatör çıkışında ki gözlem sinyali $o(iT_b)$ ise Denklem 3.14'e eşit olur [83, 84].

$$s(t) = \begin{cases} c(t) + c(t-\tau), & \text{"1" bilgisi için} \\ c(t) - c(t-\tau), & \text{"-1" bilgisi için} \end{cases} \tag{3.13}$$

$$\begin{aligned} o(iT_b) &= \int_{(i-1)T_b}^{iT_b} r_i(t).r_i(t-\tau)dt = \int_{(i-1)T_b}^{iT_b} [s_i(t)+n(t)].[s_i(t-\tau)+n(t-\tau)]dt \\ &= \int_{(i-1)T_b}^{iT_b} s_i(t).s_i(t-\tau)dt + \int_{(i-1)T_b}^{iT_b} s_i(t).n(t-\tau)dt \\ &+ \int_{(i-1)T_b}^{iT_b} n(t).s_i(t-\tau)dt + \int_{(i-1)T_b}^{iT_b} n(t).n(t-\tau)dt \end{aligned} \tag{3.14}$$

Gürültüsüz ortamda $n(t) = 0$ olacağından Denklem 3.14 ifadesi Denklem 3.15'e eşit olur.

$$o(iT_b) = \int_{(i-1)T_b}^{iT_b} s_i(t).s_i(t-\tau)dt \tag{3.15}$$

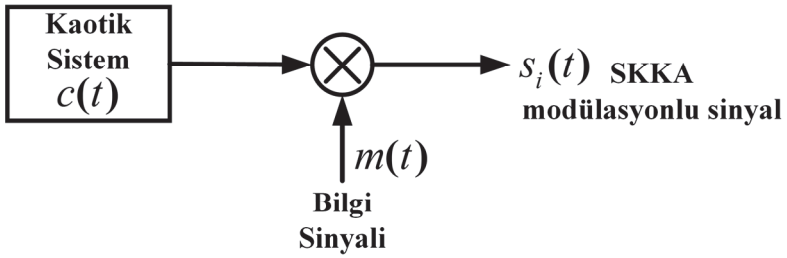
3.2.7. Simetrik kaos kaydırmalı anahtarlama

"Simetrik Kaos Kaydırmalı Anahtarlama (SKKA) (Symmetric Chaos Shift Keying - SCSK)" yöntemi Sushchic ve arkadaşları tarafından KGKA yöntemiyle birlikte 2000 yılında önerilmiştir [83]. SKKA yöntemi evre uyumsuz KKA yönteminin bir alt sınıfı olarak tanımlanabilir. SKKA modülatöründe kaotik referans sinyali bilgi sinyali ile direkt çarpılır ve demodülatöre iletilir. SKKA modülatör blok şeması Şekil 3.24.'te, demodülatör blok şeması ise Şekil 3.25.'te verilmiştir. SKKA demodülatör yapısında vericideki kaotik sistem aynen oluşturulur ve vericiden gelen sinyal ile çarpılarak integrali alınır yani bit enerjisi hesaplanır [83, 84, 135].

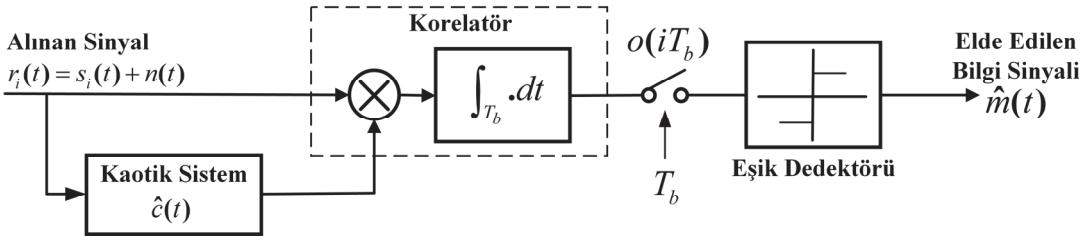
SKKA modülatör ve demodülatör blok şemaları incelendiğinde SKKA yönteminin avantaj ve dezavantajları aşağıda özetlenmiştir [84, 135].

- SKKA yapısı itibariyle referans sinyalin haberleşme kanalı üzerinden iletilmesi ihtiyacı ortadan kalkmıştır.
- Verici tasarımı diğer yöntemlere göre daha basittir.
- SKKA yönteminin FKKA ve KGKA yöntemlerine göre BER performansı 2-3dB daha düşüktür.
- SKKA yönteminde taşıyıcı sinyal tekrarlanmadığından, FKKA yöntemine göre yetkisiz girişimlerin sinyali algılama ihtimali daha düşüktür.

SKKA yönteminde alıcı tarafta verici taraftaki kaotik sistem yeniden oluşturulup eşleştirildiğinden yetkisiz girişimlere karşı daha güvenli bir modülasyon yöntemidir.



Şekil 3.24. SKKA (SCSK) modülatör blok şeması [83, 84, 135]



Şekil 3.25. SKKA (SCSK) demodülatör blok şeması [83, 84, 135]

BÖLÜM 4. YENİ KAOTİK SİSTEM: DİNAMİK ANALİZ, BENZETİM, ELEKTRONİK DEVRE TASARIMI

Bu bölümde tez çalışması kapsamında elde edilen yeni üç boyutlu kaotik sistem tanıtılmıştır. Elde edilen yeni kaotik sistemin dinamik analizleri, nümerik benzetimi ve elektronik devre tasarımı bu kısımda verilmiştir.

4.1. Yeni Kaotik Sistem

Deneme çalışmaları sonucunda elde edilen yeni üç boyutlu otonom kaotik sistemin matematiksel ifadesi Denklem 4.1'de verilmiştir. Sistem (Denklem 4.1), üç adet durum değişkeni (*x, y, z*), üç adet negatif (*a, c, d*) ve bir adet pozitif (*b*) olmak üzere toplam dört adet sabit terim ile dört adet doğrusal olmayan (*xz, xz, yz, xy*) ifade içermektedir.

$$\begin{aligned} \dot{x} &= y + ax + bxz \\ \dot{y} &= cxz + dx + yz + 1 \\ \dot{z} &= 1 + xy \end{aligned} \tag{4.1}$$

Sistemin başlangıç şartları $x(0) = 0, y(0) = 0, z(0) = 0$ 'dır. Sistem parametreleri ise $a = -0,6,\ b = 3,\ c = -10,\ d = -0,3$ şeklindedir. Denklem 4.2'de yeni kaotik sistemin parametreleri yazılmış ifadesi verilmiştir.

$$\begin{aligned} \dot{x} &= y - 0.6x + 3xz \\ \dot{y} &= -10xz - 0.3x + yz + 1 \\ \dot{z} &= 1 + xy \end{aligned} \tag{4.2}$$

4.2. Yeni Kaotik Sistemin Dinamik Analizleri

Bu kısımda yeni kaotik sistemin kaos özelliği gösterip göstermediği Bölüm 2.'de anlatılan çeşitli kaos analiz yöntemleri ile incelenmiştir. Bölüm 4.3.'te ise sistemin (Denklem 4.1) nümerik benzetimi yapılarak sistemin durum değişkenlerinin ve faz uzaylarının yörüngeleri incelenmiştir.

4.2.1. Kararlılık ve denge noktaları analizi

Kaotik bir sistem kararsız bir davranış göstermelidir. Sistemde en az bir pozitif öz değer (eigenvalue) var ise sistem kararsızdır denir. Bir sistemin denge noktalarını (equilibrium points) bulmak için, sistemin durum değişkenlerinin türevi sıfıra eşitlenir (Denklem 4.3).

$$\begin{aligned} & y - 0.6x + 3xz = 0 \\ & -10xz - 0.3x + yz + 1 = 0 \\ & 1 + xy = 0 \end{aligned} \tag{4.3}$$

Denklem 4.3 çözüldüğünde sistemin Denklem 4.4'te verildiği gibi dört adet denge noktasına sahip olduğu görülür.

$$\begin{aligned} E_1 &= (\text{-}0,01606 + 0,31559i, \quad 0,160845 + 3,16048i, \quad \text{-}3,12091 + 0,33891i) \\ E_2 &= (\text{-}0,01606 - 0,31559i, \quad 0.16085 - 3,16048i, \quad \text{-}3,12091 - 0,33891i) \\ E_3 &= (0.23345 + 1.18189i, \quad \text{-}0.16085 + 0.81433i, \quad \text{-}0.01242 - 0.08732i) \\ E_4 &= (0.23345 - 1.18189i, \quad \text{-}0.16085 - 0.81433i, \quad \text{-}0.01242 + 0.08732i) \end{aligned} \tag{4.4}$$

E_1 denge noktası için Jakobiyen matrisi Denklem 4.5'te hesaplanmıştır.

$$J(E_1)=\begin{bmatrix} \partial x/\partial x & \partial x/\partial y & \partial x/\partial z \\ \partial y/\partial x & \partial y/\partial y & \partial y/\partial z \\ \partial z/\partial x & \partial z/\partial y & \partial z/\partial z \end{bmatrix}=\begin{bmatrix} bz+a & 1 & bx \\ cz+d & z & y+cx \\ y & x & 0 \end{bmatrix}$$
$$=\begin{bmatrix} -9,9627+1,0167i & 1 & -0,0482+0,9468i \\ 30,9091-3,3891i & -3,1209+0,3389i & 0,3214+0,0046i \\ 0,1608+3,1605i & -0,0161+0,3156i & 0 \end{bmatrix} \quad (4.5)$$

Sistemin E_1 denge noktasındaki karakteristik denklemi Denklem 4.6 kullanılarak Denklem 4.7'de verildiği gibi bulunur.

$$\det(J-\lambda.I)=\left|J-\lambda.I\right|=0 \quad (4.6)$$

$$\lambda^3+(13,0836+1,3556i)\lambda^2+(2,8457+3,2619i)\lambda+(18,6028+3,1202i) \quad (4.7)$$

Denklem 4.7 çözüldüğünde E_1 denge noktası için sistemin öz değerleri $\lambda_1=-12,9555+1,1239i, \lambda_2=-0,0224+1,3229i, \lambda_3=-0,1057-1,0912i$ olarak bulunur.

Aynı şekilde E_2 denge noktası için Jakobiyen matrisi Denklem 4.8'de hesaplanmıştır.

$$J(E_2)=\begin{bmatrix} -9.9627-1.0167i & 1 & -0.0482-0.9468i \\ 30.9091+3.3891i & -3.1209-0.3389i & 0.3215-0.0046i \\ 0.1608-3.1605i & -0.0161-0.3156i & 0 \end{bmatrix} \quad (4.8)$$

Sistemin E_2 denge noktasındaki karakteristik denklemi Denklem 4.6 kullanılarak Denklem 4.9'da verildiği gibi bulunur.

$$\lambda^3+(13.0836+1.3556i)\lambda^2+(2.8457+3.2619i)\lambda+(18.6028+3.1202i) \quad (4.9)$$

Denklem 4.9 çözüldüğünde E_2 denge noktası için sistemin öz değerleri $\lambda_1 = -12,9555 - 1,1239i, \lambda_2 = -0,0224 - 1,3229i, \lambda_3 = -0,1057 + 1,0912i$ olarak bulunur.

E_3 denge noktası için ise Jakobiyen matrisi Denklem 4.10'da hesaplanmıştır.

$$J(E_3) = \begin{bmatrix} -0.6373 - 0.2620i & 1 & 0.7004 + 3.5457i \\ -0.1758 + 0.8732i & -0.0124 - 0.0873i & -2.4953 - 11.0046i \\ -0.1608 + 0.8143i & 0.2334 + 1.1819i & 0 \end{bmatrix} \quad (4.10)$$

Sistemin E_3 denge noktasındaki karakteristik denklemi Denklem 4.6 kullanılarak Denklem 4.11'de verildiği gibi bulunur.

$$\lambda^3 + (0.6497 + 0.3493i)\lambda^2 + (-9.2628 + 4.7040i)\lambda + (-17.9505 + 4.5935i) \quad (4.11)$$

Denklem 4.11 çözüldüğünde E_3 denge noktası için sistemin öz değerleri $\lambda_1 = -2,4796 + 1,2018i, \lambda_2 = -1,6976 - 0,7606i, \lambda_3 = 3,5275 - 0,7905i$ olarak bulunur.

E_4 denge noktası için ise Jakobiyen matrisi Denklem 4.12'de hesaplanmıştır.

$$J(E_4) = \begin{bmatrix} -0.6373 + 0.2620i & 1 & 0.7004 - 3.5457i \\ -0.1758 - 0.8732i & -0.0124 + 0.0873i & -2.4953 + 11.0046i \\ -0.1608 - 0.8143i & 0.2334 - 1.1819i & 0 \end{bmatrix} \quad (4.12)$$

Sistemin E_4 denge noktasındaki karakteristik denklemi Denklem 4.6 kullanılarak Denklem 4.13'te verildiği gibi bulunur.

$$\lambda^3 + (0.6497 - 0.3493i)\lambda^2 + (-9.2628 - 4.7040i)\lambda + (-17.9505 - 4.5935i) \quad (4.13)$$

Denklem 4.13 çözüldüğünde E_4 denge noktası için sistemin öz değerleri $\lambda_1 = -2,4796 - 1,2018i, \lambda_2 = -1,6976 + 0,7606i, \lambda_3 = 3,5275 + 0,7905i$ olarak bulunur.

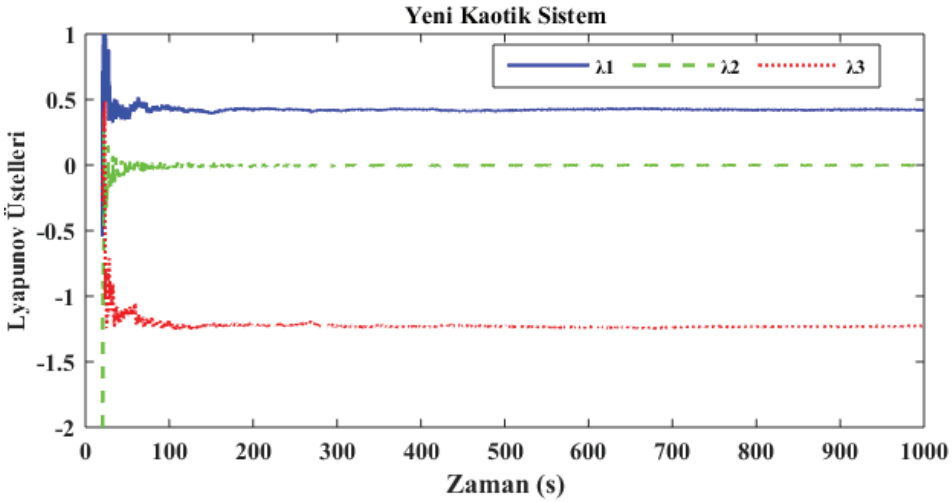
Yeni kaotik sistemin denge noktaları ve öz değerleri Tablo 4.1.'de toplu olarak verilmiştir. Tablo 4.1.'den de görüldüğü üzere sistemin denge noktalarındaki öz değerlerinde pozitif öz değer olduğundan yeni kaotik sistem kararsızdır.

Tablo 4.1. Yeni kaotik sistemin denge noktaları ve öz değerleri

Parametreler	Denge noktaları	Öz değerler
a= –0.6, b= 3, c= –10 d= –0.3	–0.01606±0.31559*i*	–12.9555±1.1239*i*
	0.160845±3.16048*i*	–0.0224±1.3229*i*
	–3.12091±0.33891*i*	–0.1057±1.0912*i*
	0.23345±1.18189*i*	–2.4796±1.2018*i*
	–0.16085±0.81433*i*	–1.6976±0.7606*i*
	–0.01242±0.08732*i*	3.5275±0.7905*i*

4.2.2. Lyapunov üstelleri analizi

Kaotik sistemler başlangıç şartlarına hassas bağımlıdırlar. Bir sistemin başlangıç şartlarına hassas bağımlılığı Lyapunov üstelleri ile belirlenir. Üç boyutlu bir sistemin Lyapunov üstelleri eğer pozitif, sıfır, negatif şeklinde ise bu sistem için kaos özelliği gösteriyor denebilir ve sistemin faz uzayı garip çekici yani kaos şeklindedir [125, 135-140]. Yeni kaotik sistemin hesaplanan Lyapunov üstelleri $\lambda_{L1} = 0,4379, \lambda_{L2} \cong 0, \lambda_{L3} = -1,2288$ şeklindedir. Lyapunov üstellerinin zamana göre grafiği ise Şekil 4.1.'de verilmiştir. Sistemin Lyapunov üstelleri incelendiğinde pozitif, sıfır, negatif şeklinde olduğundan sistemde kaosun varlığından söz edilebilir.



Şekil 4.1. Yeni kaotik sistemin Lyapunov üstelleri grafiği

4.2.3. Lyapunov boyutu hesabı

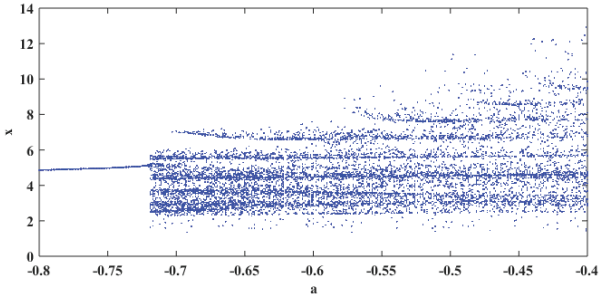
Dinamik bir sistemde de kaosun varlığı araştırılırken sistemin boyutunun fraktal yani kesirli olup olmadığına bakılır. Dinamik sistemlerde sistem boyutunun pratik olarak tespiti için Kaplan ve Yorke tarafından sunulan metot ile Lyapunov üstelleri kullanılarak Lyapunov boyutu hesaplanır. Eğer sistemin Lyapunov boyutu kesirli yani fraktal çıkarsa sistemin için kaotik özellik gösterdiği söylenebilir. Denklem 4.14'de yeni kaotik sistemin Lyapunov boyutu hesabı verilmiştir. Sistemin Lyapunov boyutu $D_L = 2,3564$ olarak hesaplanmıştır. Sistemin boyutu kesirli yani fraktal olduğundan sistemin kaos özelliği gösterdiği söylenebilir [140, 152, 153].

$$D_L = j + \frac{\sum_{i=1}^{j} \lambda_{L_i}}{\left|\lambda_{L_{J+1}}\right|} = 2 + \frac{\sum_{i=1}^{2} (\lambda_{L_1} + \lambda_{L_2})}{\left|\lambda_{L_{2+1}}\right|} = 2 + \frac{\sum_{i=1}^{2} (\lambda_{L_1} + \lambda_{L_2})}{\left|\lambda_{L_3}\right|} = 2,3564 \tag{4.14}$$

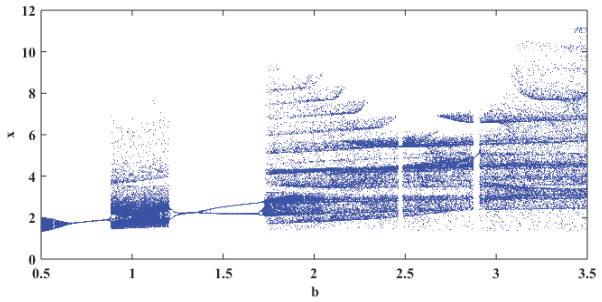
4.2.4. Çatallaşma diyagramları analizi

Dinamik bir sistemin akışı yani çekici davranışı sistemin parametre değerlerine göre değişir. Dinamik sistemdeki bu değişiklik *çatallaşma* olarak isimlendirilir. Dinamik sistemin bir parametresinin alacağı değişik değerlere göre sistem durum değişkeninin aldığı değerlerin birbirlerine göre çizdirilmesi ile elde edilen grafik *çatallaşma*

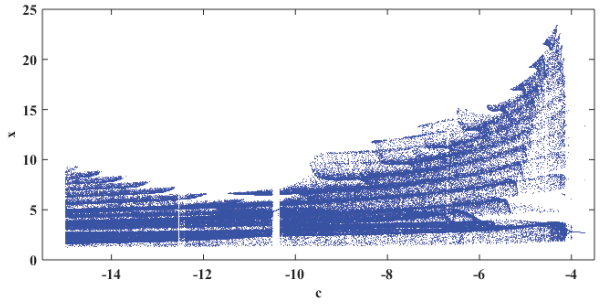
diyagramı olarak tanımlanır. Çatallaşma diyagramları sistemin hangi parametre değerlerinde kaosa girdiğini gösterir [2, 148]. Yeni kaotik sistemin *a, b, c, d* parametrelerinin çatallaşma diyagramı Şekil 4.2.'de verilmiştir. Şekil 4.2. incelendiğinde *a, b, c, d* parametrelerinin hangi değerlerinde sistemin kaos durumunda olduğu veya kaos durumundan çıktığı görülmektedir.



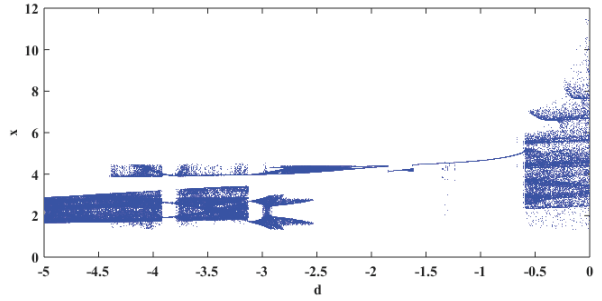
(a)



(b)



(c)

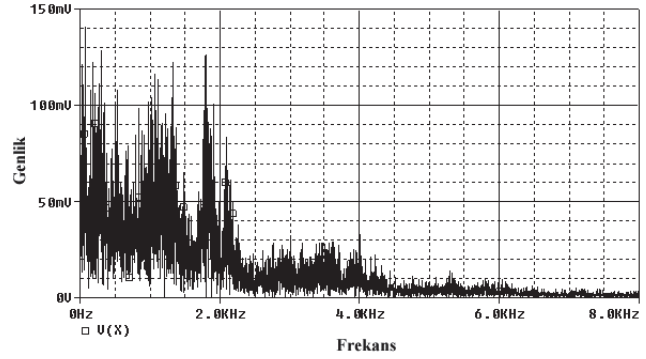


(d)

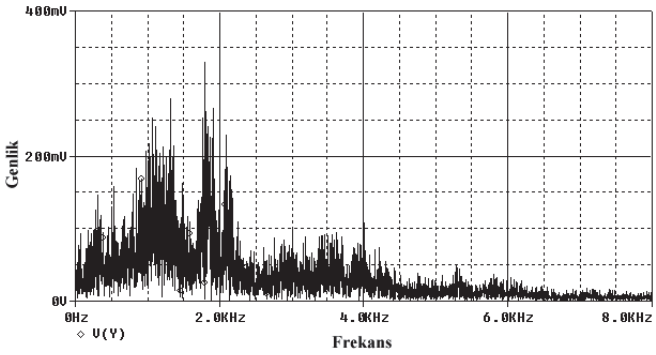
Şekil 4.2. Yeni kaotik sistemin çatallaşma diyagramları (a) a-x (b) b-x (c) c-x (d) d-x

4.2.5. Frekans spektrumu analizi

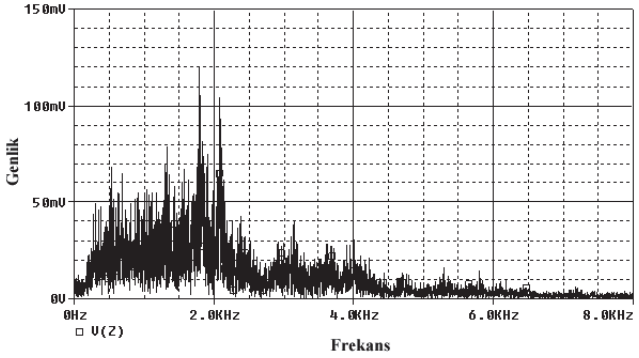
Yeni kaotik sistemin frekans spektrum analizi yapılarak hangi frekans aralıklarına sahip olduğu incelenmiştir. Şekil 4.3.'te yeni kaotik sistemin *x, y, z* durum değişkenlerinin Genlik-Frekans spektrum analizi grafikleri verilmiştir. Grafiklerden görüleceği üzere sistem çıkışları yaklaşık 0-6KHz arasında bir dağılım göstermektedir.



(a)



(b)

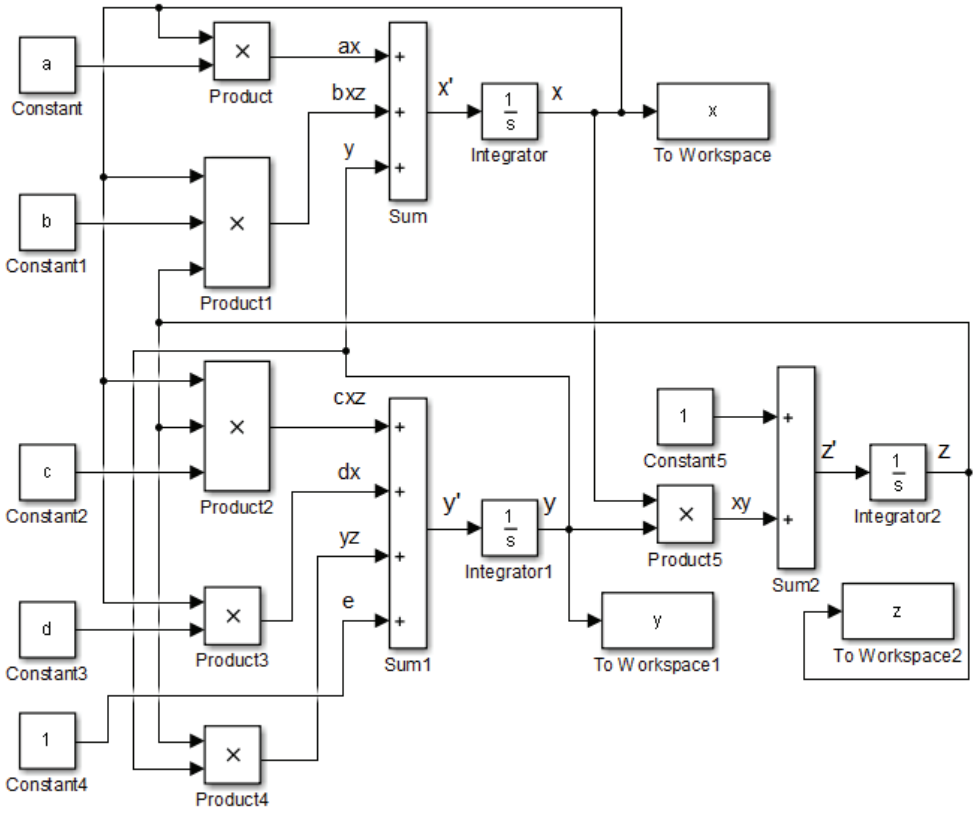


(c)

Şekil 4.3. Yeni kaotik sistemin durum değişkenlerinin frekans spektrumları (a) |X(f)| (b) |Y(f)| (c) |Z(f)|

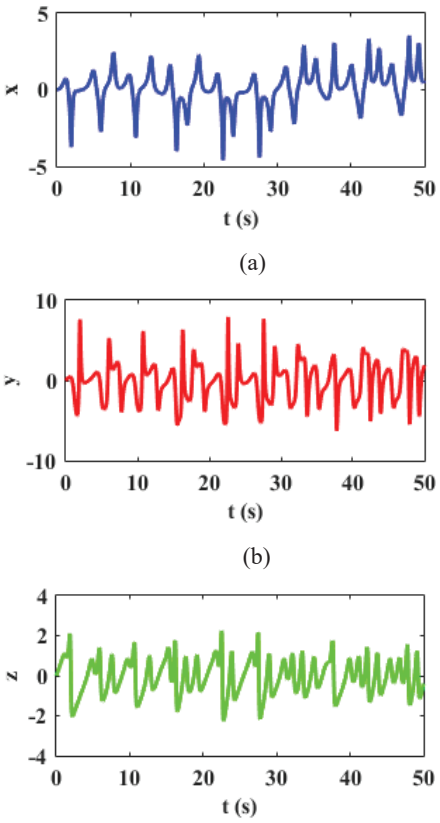
4.3. Yeni Kaotik Sistemin Nümerik Benzetimi

Yeni kaotik sistemin Matlab-Simulink ortamında gerçekleştirilen benzetim çalışmasının blok şeması Şekil 4.4.'te verilmiştir.

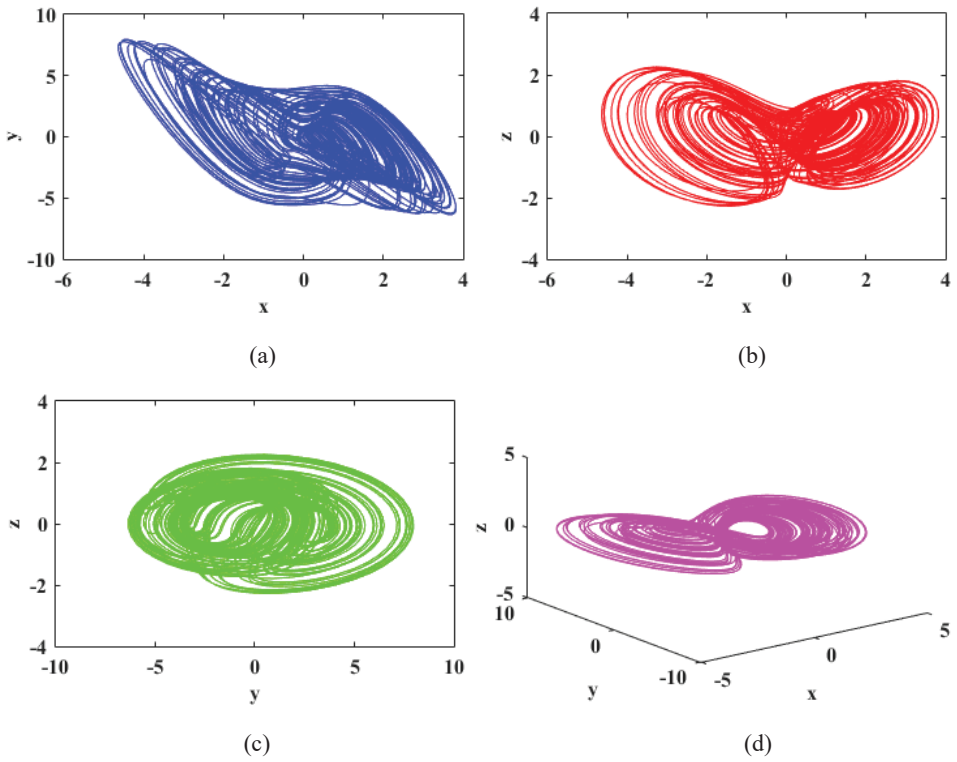


Şekil 4.4. Yeni kaotik sistemin Matlab-Simulink nümerik benzetimi blok şeması

Matlab-Simulink benzetimi sonucu yeni kaotik sistemin *x, y, z* durum değişkenlerinin zamana göre grafiği Şekil 4.5.'te verilmiştir. Aynı şekilde yeni kaotik sistemin *x-y, x-z, y-z* ve *x-y-z* faz uzayları (portreleri) ise Şekil 4.6.'da verilmiştir. Sistemin faz uzaylarına bakıldığında sistem acayip (garip) çekici formunda kaotik davranış göstermektedir.



Şekil 4.5. Yeni kaotik sistemin durum değişkenlerinin zamana göre grafikleri (a) x (b) y (c) z

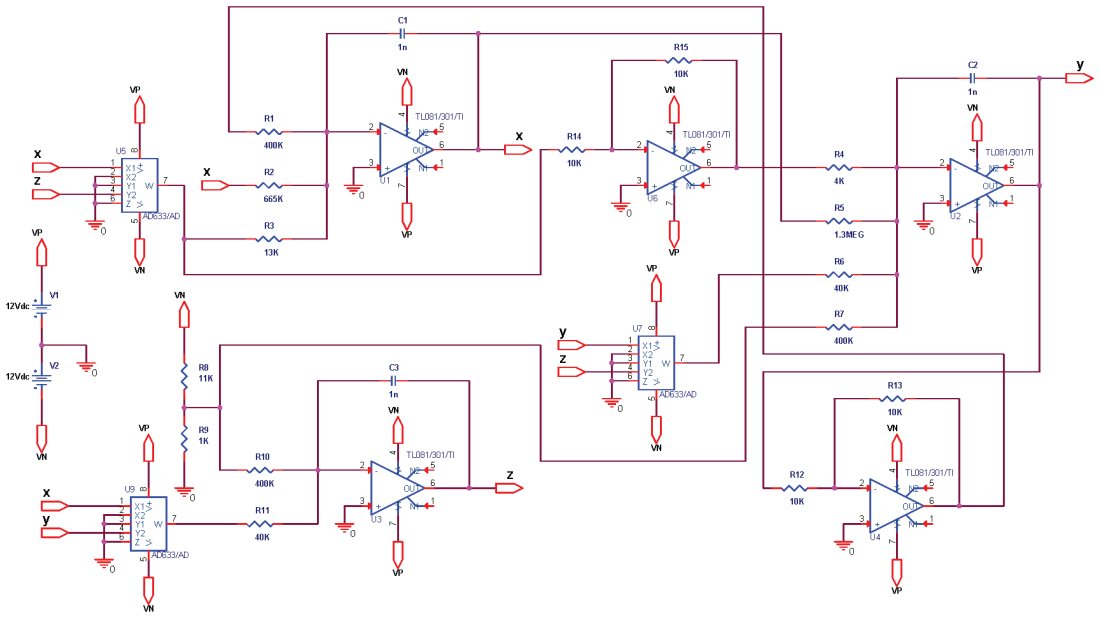


Şekil 4.6. Yeni kaotik sistemin nümerik benzetim sonucu faz uzayları (a) x-y (b) x-z (c) y-z (d) x-y-z

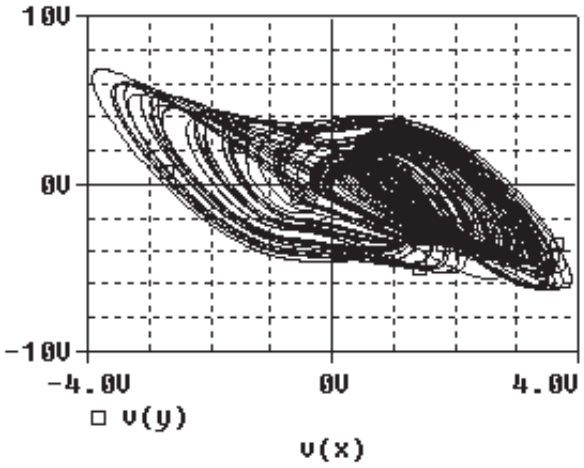
4.4. Yeni Kaotik Sistemin Elektronik Devre Tasarımı

Kaotik sistemlerin nümerik benzetiminin yanında pratik uygulamalar için elektronik devre tasarımlarına da ihtiyaç vardır. Bu nedenle nümerik analizden sonra yeni kaotik sistemin elektronik devre tasarımı Orcad-PSPICE® programında gerçekleştirilmiştir. Elektronik devre tasarımında (Şekil 4.7.) TL081 opamp, AD633JN analog çarpıcı ve uygun değerlerde direnç ve kondansatörler kullanılmıştır. Sistemin kaotik diferansiyel denklemleri Denklem 4.15'te verilmiştir. Tasarlanan elektronik devrenin *x, y, z* çıkışlarının faz portreleri Şekil 4.8.'de verilmiştir. Şekil 4.6. ve Şekil 4.8.'den görüldüğü gibi nümerik benzetim ile elektronik devre çıkışları birbirinin aynısı olduğundan tasarlanan elektronik devre pratik uygulamalar da kullanılabilir.

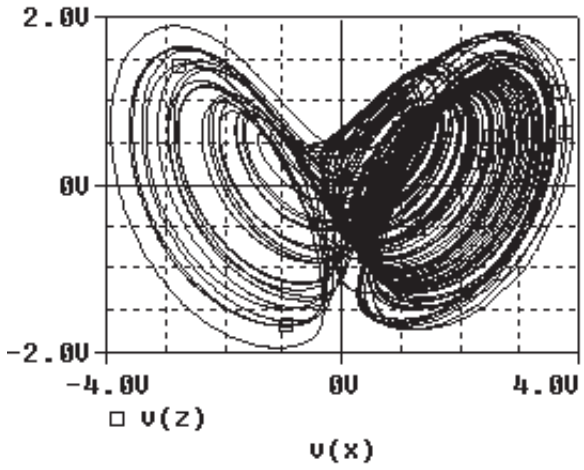
$$\begin{aligned}\dot{x} &= \frac{1}{R_1C_1}y - \frac{0.6}{R_2C_1}x + \frac{3}{R_3C_1}xz \\ \dot{y} &= -10\frac{10}{R_4C_2}xz - \frac{0.3}{R_5C_2}x + \frac{3}{R_6C_2}yz + 1 \\ \dot{z} &= 1 + \frac{1}{R_7C_3}xy\end{aligned} \tag{4.15}$$



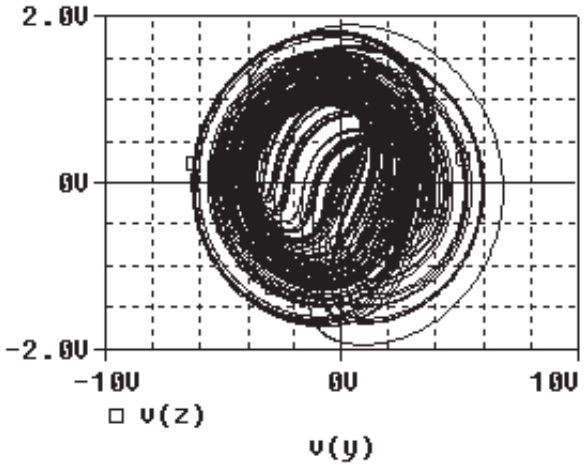
Şekil 4.7. Yeni kaotik sistemin elektronik devre tasarımı



(a)



(b)



(c)

Şekil 4.8. Yeni kaotik sistemin elektronik devre tasarımı çıkışları faz portreleri (a) x-y (b) x-z (c) y-z

BÖLÜM 5. YENİ KAOTİK SİSTEMİN FPGA TABANLI TASARIMI

Bu bölümde tez çalışması kapsamında elde edilen yeni kaotik sistemin ilk önce Euler algoritması ile nümerik olarak hesaplanması incelenmiştir. Ardından yeni kaotik sistemin Euler algoritması ile VHDL donanım tanımlama dili kullanılarak FPGA üzerinde tasarımı gerçekleştirilmiştir.

5.1. Yeni Kaotik Sistemin Euler Algoritması İle Nümerik Olarak Hesaplanması

Yeni kaotik sistemin ayrık matematiksel modelinin çıkartılmasında basit olması açısından Euler algoritması seçilmiştir. Euler yöntemiyle diferansiyel denklemi çözebilmek için $[x_i, x_{i+1}]$ aralığındaki bilgilerden ve y_i'den yararlanılmaktadır. Euler yöntemi tek adımlı yöntem olarak tanımlanmaktadır. Euler yöntemi açık ve kapalı Euler yöntemi olarak iki çeşittir. Bu çalışmada açık Euler yöntemi kullanılmıştır. Açık Euler yönteminde Denklem 5.1'de verildiği gibi sistemin eğim bilgisi kullanılmaktadır [166].

$$f(x, y) = \frac{dy}{dx} \tag{5.1}$$

Sistem değişkeninin her yeni değeri adım büyüklüğüne (h) bağlı olarak Denklem 5.2'deki gibi hesaplanmaktadır [166].

$$x_{k+1} = x_k + h \qquad \text{k} = 1, 2, 3, \ldots, n \tag{5.2}$$

Denklem 5.1'de verilen türev ifadesi ileri doğru sonlu fark karşılığı şeklinde Denklem 5.3'te verildiği gibi yazılır [166].

$$f(x_k, y_k) = \frac{y_{k+1} - y_k}{h} \tag{5.3}$$

Sonuç olarak Denklem 5.3'ten y_{k+1} ifadesi çekilirse Euler algoritmasının genel yapısı Denklem 5.4'teki gibi olur [166, 167, 168].

$$y_{k+1} = y_k + h.f(x_k, y_k) \tag{5.4}$$

Denklem 5.4'teki h ifadesi Euler algoritmasının adım büyüklüğüdür. Bu değer ne kadar küçük olursa hesaplama o kadar hassas olur. Adım büyüklüğü yarıya düşürülürse hesaplamanın hata oranı da yarıya düşer [166, 167, 168].

Denklem 5.5'te de yeni kaotik sisteminin Euler algoritması kullanılarak ayrıklaştırılmış modeli verilmiştir. Denklemlerdeki x_k, y_k, z_k ifadeleri o anki sistem durum değişkenlerinin değerlerini, x_{k+1}, y_{k+1}, z_{k+1} ifadeleri ise bir sonraki adımın sistem durum değişkenlerinin değerlerini ifade etmektedir.

$$\begin{aligned} x_{(k+1)} &= x_{(k)} + h.\dot{x} = x_{(k)} + h(y_{(k)} + a.x_{(k)} + b.x_{(k)}.z_{(k)}) \\ y_{(k+1)} &= y_{(k)} + h.\dot{y} = y_{(k)} + h(c.x_{(k)}.z_{(k)} + d.x_{(k)} + y_{(k)}.z_{(k)} + 1) \\ z_{(k+1)} &= z_{(k)} + h.\dot{z} = z_{(k)} + h(1 + x_{(k)}.y_{(k)}) \end{aligned} \tag{5.5}$$

Euler algoritması ile nümerik hesaplama için ilk önce Matlab ortamında yeni kaotik sistemin tanıtıldığı Yeni_Kaotik_Sistem_Euler.m adlı bir dosya oluşturulmuştur. Ardından yeni kaotik sistemin Euler algoritması ile hesaplanmasını sağlayacak Euler.m adlı Matlab dosyası oluşturulmuştur. Şekil 5.1.'de bu dosyaların içerikleri verilmiştir.

Matlab ortamında Euler algoritması ile toplam 300.000 adet veri alınmıştır (Şekil 5.2.). Euler algoritması ile Matlab programında yapılan hesaplamada Euler adım büyüklüğü, hesaplamanın çok hassas olması için *h=0,001* alınmıştır. Euler nümerik analizi sonucu elde edilen verilerin zamana göre ve birbirlerine göre faz uzayları (portreleri) çizdirilmiştir. Şekil 5.3.'te *x, y, z* sistem durum değişkenlerinin zamana

karşı çizimleri, Şekil 5.4.'te ise *x, y, z* durum değişkenlerinin birbirlerine göre çizdirilmiş faz uzayları verilmiştir.

Yeni_Kaotik_Sistem_Euler.m

```
function yp=Yeni_Kaotik_Sistem_Euler(t,y)


a=-0.6;
b=3;
c=-10;
d=-0.3;


yp=[y(2)+a*y(1)+b*y(1)*y(3);c*y(1)*y(3)
+d*y(1)+y(2)*y(3)+1;1+y(1)*y(2)];
```

Euler.m

```
clc
clear
h=0.001;%Artış değeri(Zaman)
time_son=300.0;%(Son zaman degeri)
ye=[];
y0=[0,0,0];%Başlangıç değerleri x1(0) x2(0)
for time=0:h:time_son
ye=[ye; y0];
yturev=Yeni_Kaotik_Sistem_Euler(time,y0);
yt1=y0+h*yturev';
y0=yt1;
end
time=0:h:time_son;
yk=ye(:,:,:);
time=time';
figure
plot(time,yk,'r');
title('EULER')
```

Şekil 5.1. Matlab ortamında yeni kaotik sistemin Euler algoritması ile nümerik olarak hesaplanması

x — 300000x1 double

	1
1	5.8775e-39
2	1.0000e-06
3	2.9996e-06
4	5.9979e-06
5	9.9947e-06
6	1.4989e-05
7	2.0981e-05
8	2.7969e-05
9	3.5952e-05
10	4.4933e-05
11	5.4906e-05
12	6.5880e-05

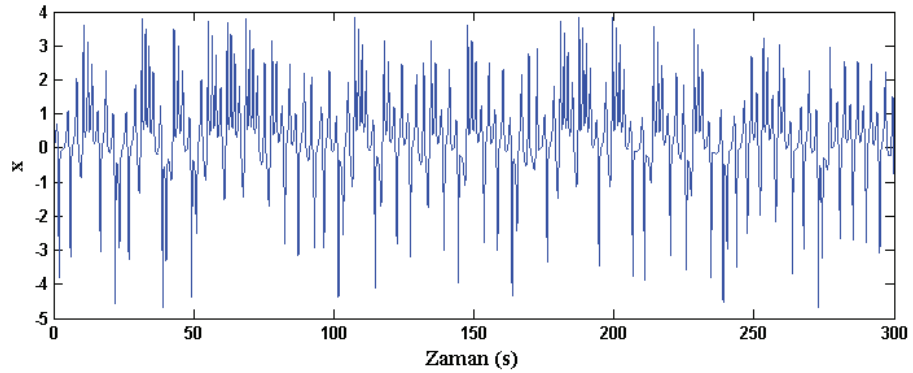
y — 300000x1 double

	1
1	0.0010
2	0.0020
3	0.0030
4	0.0040
5	0.0050
6	0.0060
7	0.0070
8	0.0080
9	0.0090
10	0.0100
11	0.0110
12	0.0120

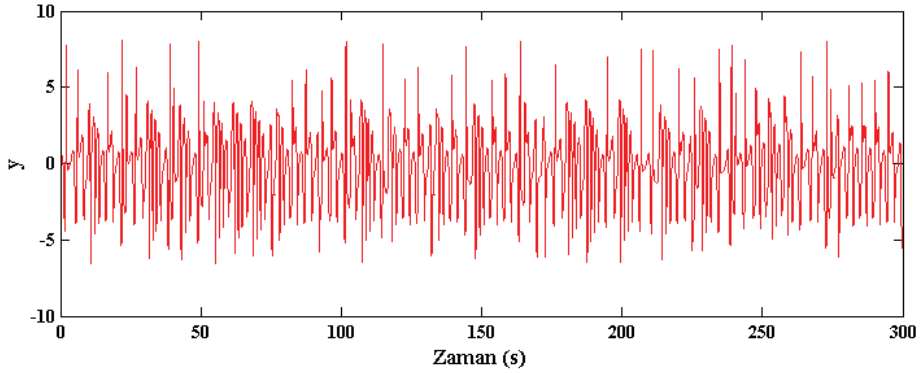
z — 300000x1 double

	1	2
101213	-0.0088	
101214	-0.0157	
101215	-0.0226	
101216	-0.0294	
101217	-0.0362	
101218	-0.0431	
101219	-0.0498	
101220	-0.0566	
101221	-0.0633	
101222	-0.0700	
101223	-0.0767	
101224	-0.0834	

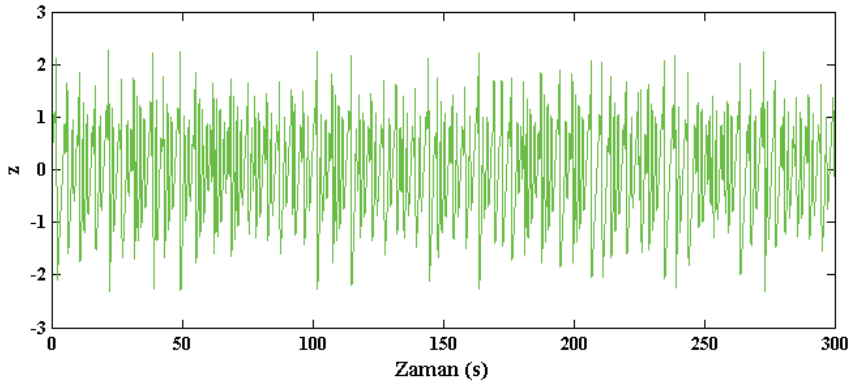
Şekil 5.2. Matlab ortamında yeni kaotik sistemin Euler algoritması ile elde edilen sonuçlarından bir görüntü



(a)

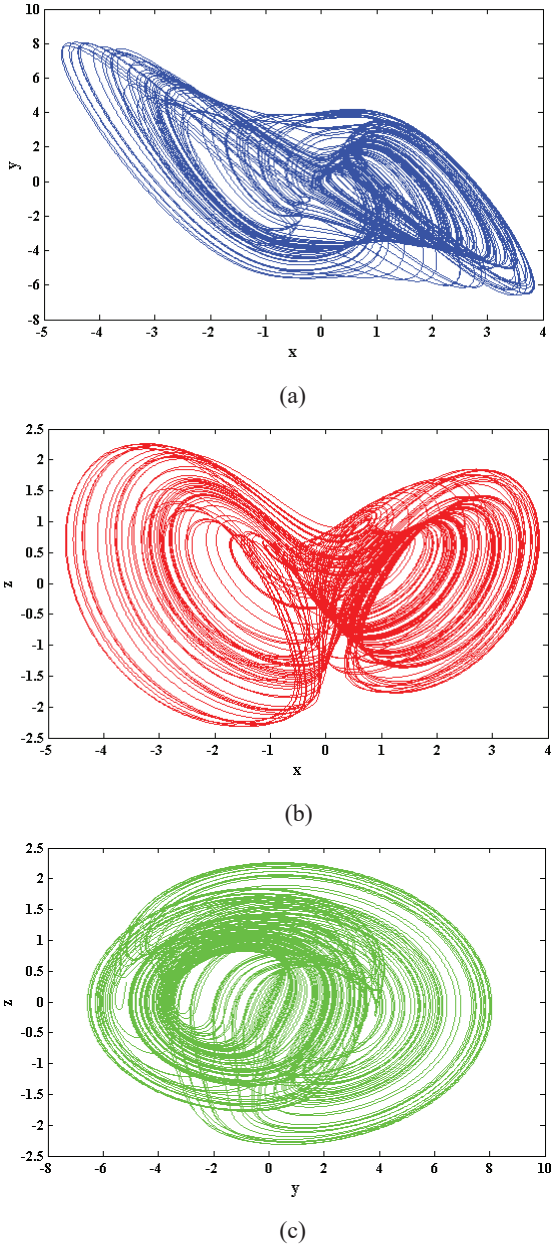


(b)



(c)

Şekil 5.3. Matlab ortamında yeni kaotik sistemin Euler algoritmasına göre nümerik olarak hesaplatılmış durum değişkenlerinin zamana göre grafikleri (a) x (b) y (c) z



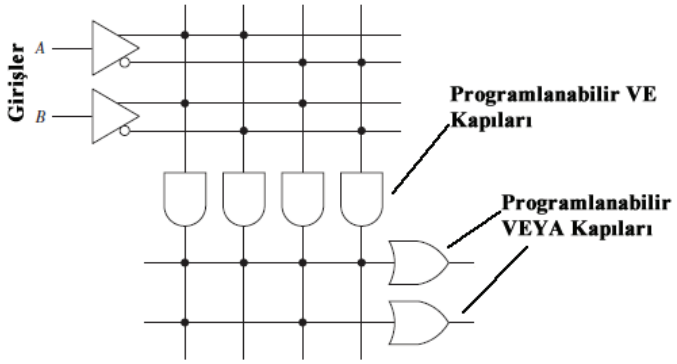
Şekil 5.4. Matlab ortamında yeni kaotik sistemin Euler algoritmasına göre nümerik olarak hesaplatılmış faz uzayları (a) x-y (b) x-z (c) y-z

5.2. Yeni Kaotik Sistemin FPGA İle Tasarımı

Tez çalışmasının bu bölümünde, Euler algoritması ile nümerik olarak elde edilen yeni kaotik sistemin FPGA üzerinde tasarımı gerçekleştirilmiştir. İlk olarak FPGA ve kayan noktalı sayı sistemi formatı üzerine bilgi verilmiş ve ardından yeni kaotik sistemin FPGA tasarımı hakkında bilgi verilmiştir.

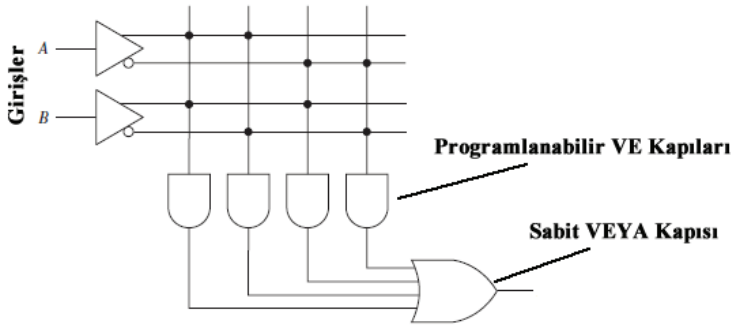
5.2.1. FPGA

Tasarımcıların hızlı tasarım, düşük maliyet, çok az yer kaplama gibi isteklerinden dolayı tümleşik devreler (Integrated Circuit - IC) geliştirilmiştir. Bu gelişimle birlikte uygulamaya özel tümleşik devre (Application Specific Integrated Circuit - ASIC) elemanı geliştirilmiştir. ASIC elemanı, sadece tek bir uygulaya özel olarak tasarlandığından, ihtiyaca göre tasarım değişikliği, tekrar programlanabilme özellikleri yoktur. ASIC elemanlarının bu dezavantajlarından dolayı tekrar programlanabilir elemanlara ihtiyaç doğmuştur. 1970'lerde ilk programlanabilir mantık elemanları (Programmable Logic Device - PLD) üretilmiştir. PLD elemanları sadece bir kez programlanabilen VE/VEYA (VE/OR) kapıları içerir. Daha yüksek hız, daha büyük kapasite ve daha hızlı tasarım için daha sonra programlanabilir mantık dizi (Programmable Logic Array – PLA) elemanı geliştirilmiştir. PLA elemanları, programlanabilir VE ve programlanabilir VEYA kapılarından oluşur (Şekil 5.5.). PLA'lar içlerinde iki adet programlanabilir yapı barındırdığından devre tasarım karmaşıklığı ve gecikme süresi fazla olmaktadır [169-171, 175].



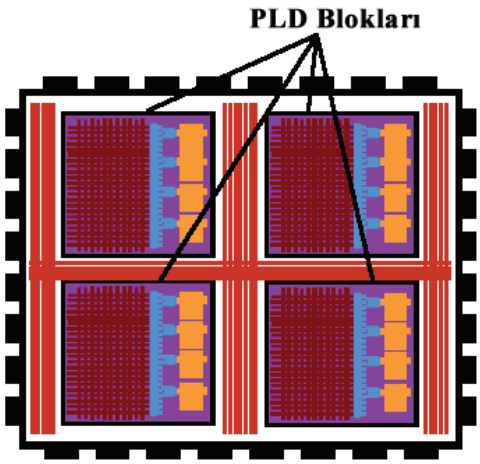
Şekil 5.5. PLA iç yapısı [175]

PLA'ların bu dezavantajını gidermek için programlanabilir dizi mantık (Programmable Array Logic – PAL) elemanı geliştirilmiştir. PAL elemanlarında programlanabilir VE kapısı ile sabit yani programlanamayan VEYA kapılarından oluşur (Şekil 5.6.) [170-172, 175].



Şekil 5.6. PAL iç yapısı [175]

PLD, PLA, PAL gibi basit programlanabilir elemanların büyük tasarımlar için yetersiz kalmasından dolayı karmaşık programlanabilir mantık devreleri (Complex Programmable Logic Device – CPLD) geliştirilmiştir. PLD, PLA, PAL elemanlarının kapasitesi yaklaşık yüz mantık kapısı eşdeğerinde iken, CPLD elemanları binler düzeyinde mantık kapısı eşdeğerine sahiptir. CPLD elemanları içlerinde çok sayıda PLD barındıran elemanlar olarak da tanımlanabilir (Şekil 5.7.) [170-172, 175].

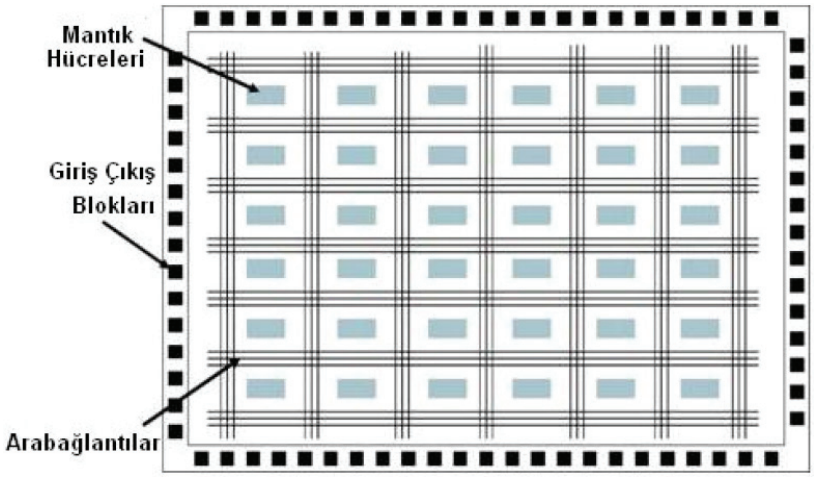


Şekil 5.7. CPLD iç yapısı [172]

CPLD'ler yüksek kapasiteli tasarımlarda kullanıldığında, mantık hücrelerinin diziliş şekli ve tek bir genel bağlantı yapısından dolayı bağlantı sorunları meydana gelir. Bu durum da CPLD elemanlarının çok büyük tasarımlarda kullanılmasını zorlaştırmaktadır. Bundan dolayı CPLD'ler yerine FPGA (Field Programmable Gate Array) elemanı geliştirilmiştir. FPGA, "Alanda Programlanabilir Kapı Dizileri" anlamındadır. Alanda programlanabilir ismi FPGA'nın üretimden sonra

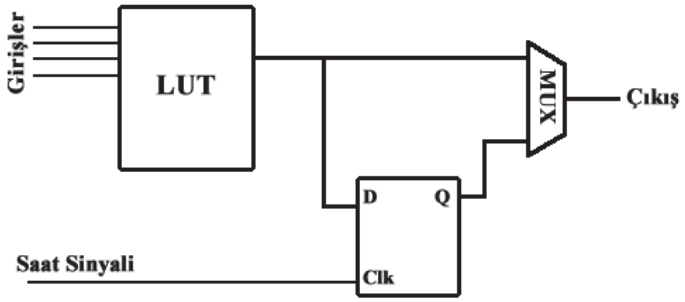
programlanabilme özelliğinden gelmektedir. FPGA elemanları ile istenen bir sayısal devre donanımsal olarak tasarlanabilir. Tasarım için bir tasarım yazılımı ile oluşturulmuş bir veri dosyası kullanılır. FPGA elemanı 1984 yılında Xilinx şirketi kurucularından olan Ross Freeman tarafından icat edilmiştir. İlk FPGA elemanını üreten lider Xilinx firması yanında Altera, Lattice, Semiconductor, ActelQuick gibi firmalar da FPGA üreten firmalardır. FPGA elemanları yüksek hız ve özellikle paralel işlem yetenekleri sayesinde uzay ve savunma sanayii, otomasyon ve kontrol sistemleri, haberleşme, şifreleme, otomotiv, sinyal işleme, tıbbi cihazlar ve tüketici elektroniği gibi birçok alanda kullanılmaktadır [170-174].

FPGA elemanı programlanabilir mantık blokları (Configurable Logic Block - CLB) , giriş-çıkış blokları (Input/Output Block – IOB) ve ara bağlantılar (Interconnection) veya diğer ismi ile ara bağlantı blokları (In Connection Block - ICB) olmak üzere temel olarak üç yapıdan oluşur (Şekil 5.8.) [170-174].



Şekil 5.8. FPGA iç yapısı [173]

CLB birimi FPGA içinde mantık işlemlerini gerçekleştiren birimdir (Şekil 5.9.). CLB yapısı temel olarak bir başvuru tablosu LUT (Look Up Table), Flip-flop (FF) ve çoklayıcı (Multiplexer – MUX) içerir. IOB birimi ise FPGA elemanının dış dünya ile iletişiminin sağlandığı birimdir. ICB birimi, mantık bloklarının birbirlerine bağlantısını sağlar [170-175].

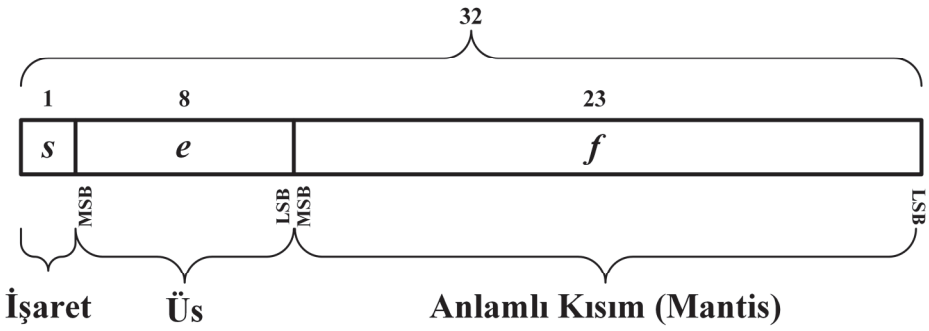


Şekil 5.9. CLB iç yapısı [170]

FPGA elemanlarının programlanması için donanım tanımlama dilleri (Hardware Description Language – HDL) veya şematik yöntemler kullanılır. HDL dilleri olarak genelde VHDL (Very High Speed Integrated Circuit Hardware Description Language – Çok Yüksek Hızlı Entegre Devre Donanımı Tanımlama Dili) veya Verilog donanım tanımlama dilleri kullanılmaktadır [170-172].

5.2.2. IEEE-754 kayan noktalı sayı formatı

32 bitlik IEEE 754-1985 kayan nokta (floating point) standartının (single format) gösterimi Şekil 5.10.'da verilmiştir. Kayan noktalı gösterim gerçel sayıların sayısal sistemlerde kullanılması için bir gösterim şeklidir. 32 bitlik kayan nokta gösteriminde sayı ikilik (binary) düzende temsil edilir. Sayının ilk biti işaret bitidir ve *s* ile gösterilir. Ondan sonraki 8 bit ise sayının üs değeridir ve *e* ile gösterilir. Geri kalan 23 bit ise kesirli kısım (fraction) veya anlamlı kısım veya mantis olarak tanımlanır ve *f* ile gösterilir. IEEE-754 formatında 32 bitlik (single) bir sayının genel gösterim ifadesi Denklem 5.6'da verilmiştir [176].



Şekil 5.10. 32 bitlik (single) IEEE 754-1985 standartında sayı gösterimi [176]

$$X = (-1)^s 2^{e-bias}(f) \tag{5.6}$$

Eğer sayının işareti negatif (-) ise işaret biti *s=1*, pozitif ise işaret biti *s=0* olur. Kayan nokta gösteriminde, sayının anlamlı kısmı (mantis) kaydırılarak *1,...* haline getirilir. Bu işleme *normalizasyon* denir. Böylece virgülün solunda hep 1 kalır. Virgülün sağında kalan kısım ise anlamlı kısım (mantis) *f* olarak alınır. Normalizasyon işlemi sonunda virgülün solunda kalan 1 değeri 32 bitlik kayan nokta gösteriminde gösterilmez. Denklem 5.6'daki bias değeri 127'dir [166, 176].

Örnek olarak onluk tabandaki $(65.125)_{10}$ sayısını 32 bitlik IEEE 754 kayan nokta formatında göstermek için öncelikle sayı virgülden önce 1 kalacak şekilde kesirli hale getirilerek normalizasyon işlemi yapılır. Denklem 5.7'de normalizasyon işlemi ve sonucu verilmiştir.

$$(65.125)_{10} = (1000001.001)_2 = (1,000001001x2^6)_2 \tag{5.7}$$

Denklem 5.7'ye bakıldığında normalizasyon değerinde virgülün sağında kalan kısım anlamlı kısmı verir. Kalan kısım toplam 23 bit olacak şekilde sonu sıfır ile tamamlanarak anlamlı kısım *f=00000100100000000000000* olur. Yine Denklem 5.7'ye göre üs değeri Denklem 5.8'de gösterildiği gibi *e=133* bulunur.

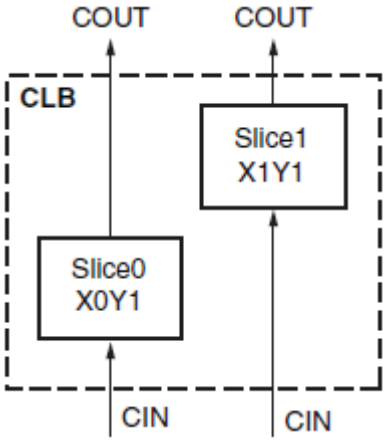
$$\begin{aligned} &e - bias = 6 \Rightarrow e - 127 = 6 \Rightarrow e = 127 + 6 \Rightarrow e = 133 \\ &e = 133 \text{ değerinin ikilik karşılığı } e = 10000101 \end{aligned} \tag{5.8}$$

Verilen örnekteki sayı pozitif olduğundan işaret biti de *s=0* olur. Buna göre bulunan parametreler Şekil 5.10.'daki sıraya göre birleştirildiğinde $(65.125)_{10}$ sayısını 32 bitlik IEEE 754 kayan nokta formatında gösterimi Denklem 5.9'da verildiği gibi olur.

$$(65.125)_{10} \Rightarrow (01000010100000100100000000000000)_2 \tag{5.9}$$

5.2.3. Yeni kaotik sistemin FPGA tabanlı tasarımı ve sonuçları

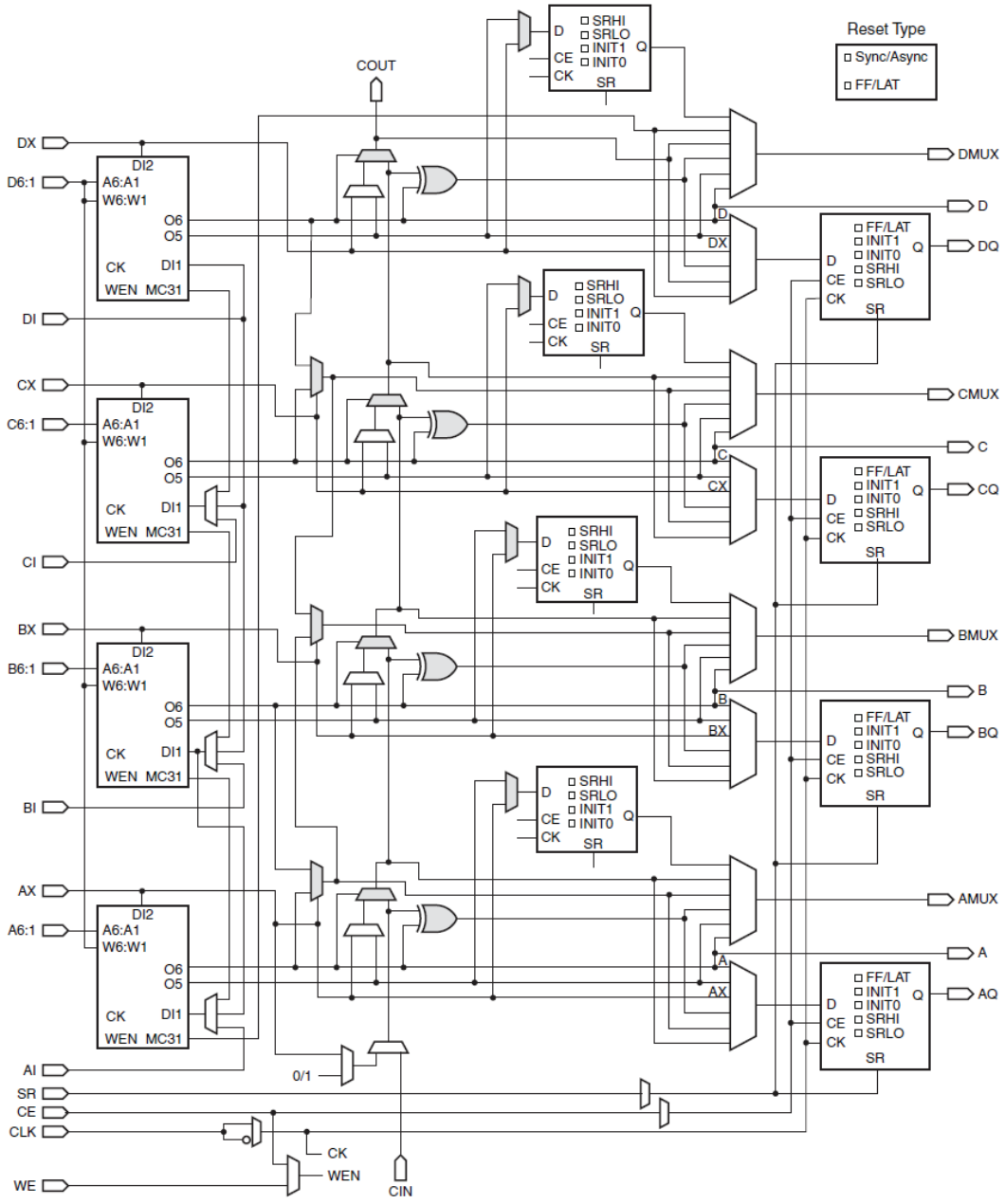
FPGA tasarımında Xilinx firmasının Artix-7 ailesinden xc7a100tcsg324-1 modeli kullanılmıştır. Bir FPGA elemanının kapasitesi genellikle sahip olduğu mantıksal hücre sayısı (Logic Cell) ile ifade edilir. Mantıksal hücreler, tüm mantıksal işlemlerin gerçekleştirilmesi için kullanılan birimlerdir. FPGA içinde bulunan programlamlanabilir mantık blokları (CLB) ile istenen mantıksal işlem için gerekli olan mantıksal hücreler birbirine bağlanarak tasarım elde edilir. FPGA içinde bulunan mantıksal hücre sayısı, CLB birimi sayısının 1,6 katıdır. CLB birimi, *Slice0* ve *Slice1* birimlerinden oluşur. CLB birimi içindeki her *Slice* biriminde temel olarak, dört adet altı girişli LUT, sekiz adet depolama elemanı (Flip-Flop) ve çoklayıcılar (multiplexer) bulunur. CLB birimi iç yapısı Şekil 5.11.'de verilmiştir [177, 178].



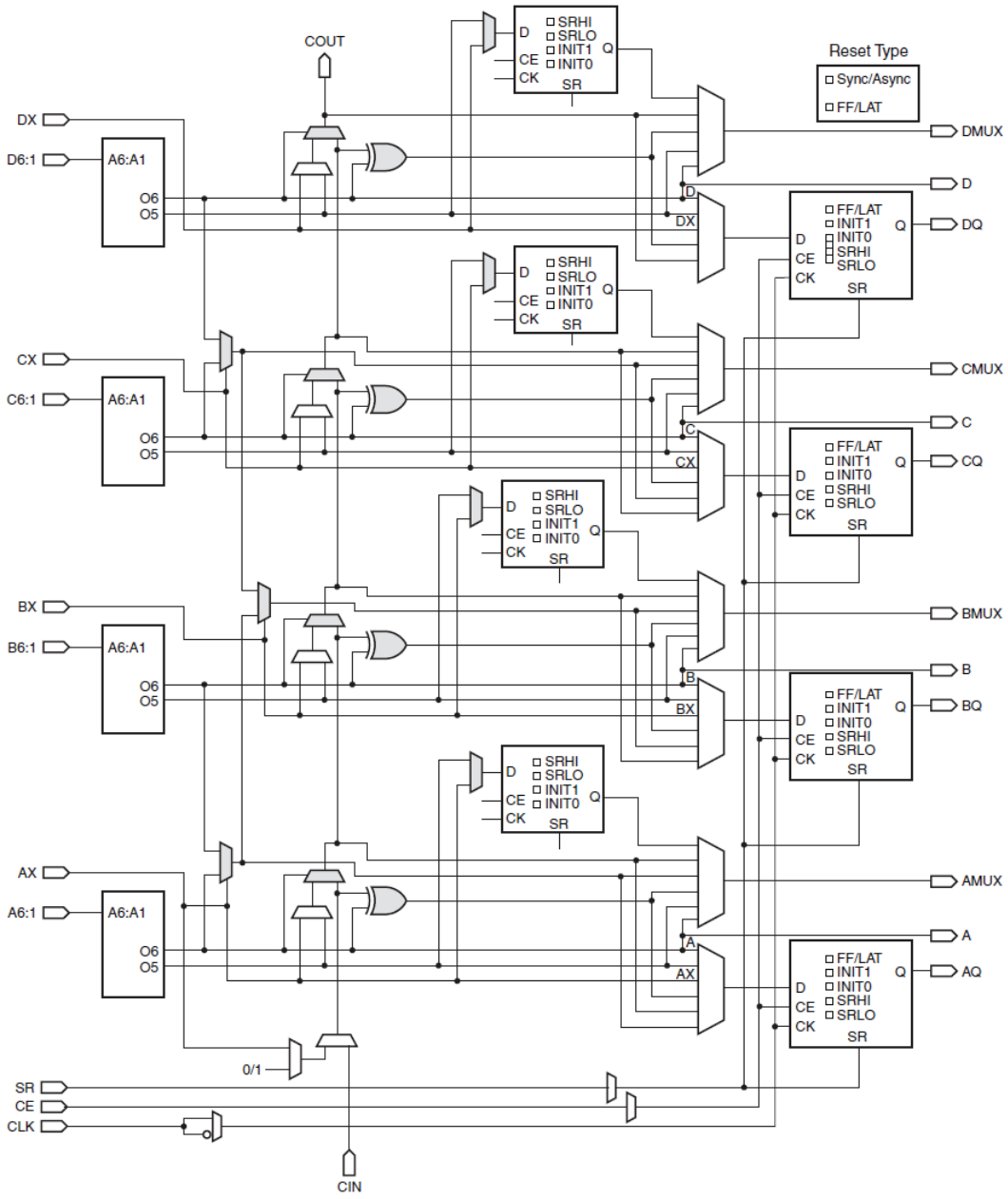
Şekil 5.11. CLB birimi iç yapısı [177]

Slice birimleri FPGA içinde tüm mantıksal, aritmetiksel ve ROM fonksiyonlarını gerçekleştirmek için kullanılırlar. *Slice* birimleri bu işlemleri gerçekleştirmek için yapısındaki *SLICEM* ve *SLICEL* bileşenlerini kullanır. *SLICEM* bileşeni iç şeması Şekil 5.12.'de ve *SLICEL* bileşeni iç şeması ise Şekil 5.13.'te verilmiştir. Her CLB birimi ya iki adet *SLICEL* birimi ya da birer adet *SLICEL* ve *SLICEM* birimleri içerir. Her LUT birimi iki adet FF içerir. FPGA içinde çeşitli işlemlerde kullanılmak üzere *SLICEM* içinde bulunan LUT'lar dağıtılmış RAM (Distributed RAM) hafızası ve kaymalı kaydedici (Shift Register) olarak kullanılmaktadır. Yine FPGA içinde çarpma, toplama, matematiksel fonksiyonlar, mantıksal işlemler, sayıcı gibi sayısal

işlemlerin hızlı ve verimli şekilde yapılmasını sağlayan hazır DSP (Digital Signal Processing- Sayısal Sinyal İşleme) birimi vardır. Tablo 5.1.'de Artix-7 xc7a100tcsg324-1 FPGA modelinin kaynakları verilmiştir [177-179].



Şekil 5.12. SLICEM bileşeni iç şeması [177]



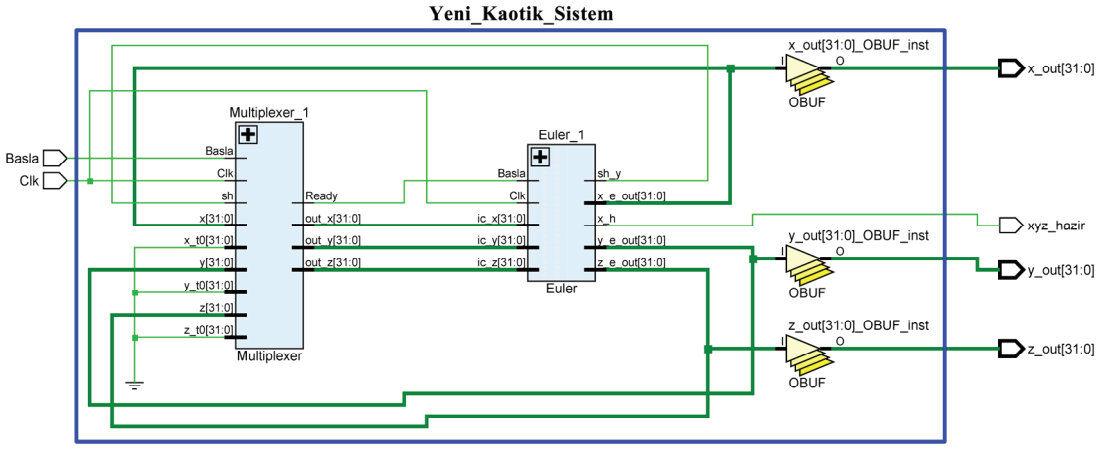
Şekil 5.13. SLICEL bileşeni iç şeması [177]

Tablo 5.1. Artix-7 xc7a100tcsg324-1 FPGA modeli kaynakları [177-179]

Mantık Hücresi	Slice	SLICEL	SLICEM	LUT	Dağıtılmış RAM (Kb)	Kaymalı Kaydedici (Kb)	Flip-Flop	DSP-48E1	Kullanıcı I/O Pin Sayısı
101440	15850	11100	4750	63400	1188	594	126800	240	210

Tasarımın kodlanmasında VHDL dili kullanılmıştır. Tasarımdaki işlem sonuçlarının daha hassas olması için işlemlerdeki sayılarda 32 bit IEEE 754-1985 kayan noktalı sayı formatı tercih edilmiştir. Tasarım, Xilinx firmasının Vivado Design Suite v2015.4 programında yapılmıştır. Yapılan tasarımda kullanılan çarpıcı ve toplayıcı üniteleri, Xilinx firmasının IP Core ürünlerinden IEEE-754 32 bit kayan noktalı sayı formatına uygun modülleri kullanılarak oluşturulmuştur.

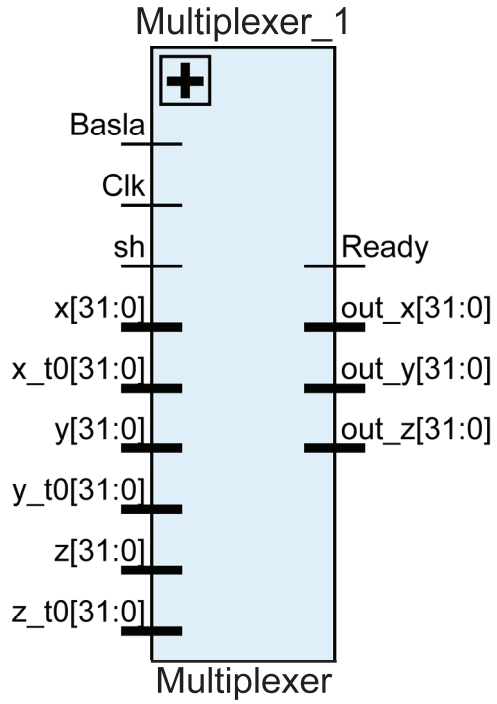
Tasarlanan sistemin en üst seviye blok diyagramı Şekil 5.14.'te verilmiştir. Sistemin giriş kısmında bir bitlik *Basla* ve *Clk* pinleri vardır. *Clk* pini ünitelerin içerisindeki alt ünitelerin zamanlaması ve ünitelerin bağlı bulunduğu sistem ile arasındaki senkronizasyonu sağlamak amacıyla kullanılan saat sinyali giriş pinidir. Sistemin çıkışında ise yeni kaotik sistemin durum değişkenleri çıkışını oluşturan 32 bitlik *x_out*, *y_out*, *z_out* pinleri ile çıkışın hazır olduğunu belirtmek için kullanılan bir bitlik *xyz_hazir* pini vardır.



Şekil 5.14. Yeni kaotik sistemin FPGA tasarımı en üst seviye blok diyagramı

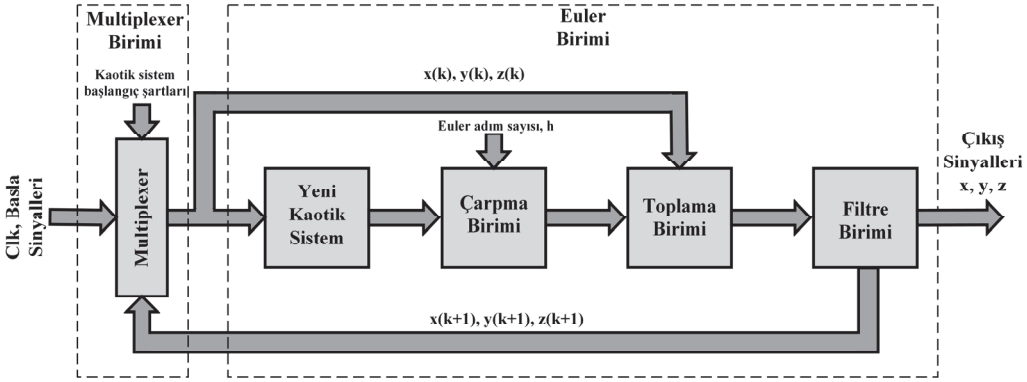
Sistemin ilk çalışması anında ihtiyaç duyduğu başlangıç şartları yeni kaotik sistem için sabit olduğundan tasarımda kullanılan FPGA çipinin kaynaklarını azaltmak amacıyla tasarımın içerisine gömülmüştür. Ancak ihtiyaç duyulduğunda bu sinyaller, 32-bitlik 3 farklı sinyal tanımlaması yapılarak tasarımda küçük değişiklikler ile değerleri kullanıcı tarafından ayarlanacak şekilde de tasarlanabilir.

Tasarımın en üst düzeyinde *Multiplexer* ve *Euler* birimleri bulunmaktadır. Tasarımda *Multiplexer* ünitesi (Şekil 5.15.) kullanılmasının amacı, başlangıç koşulu değerlerini ilk çalışma anında, kullanıcı tarafından atanan başlangıç sinyalleri olan 32-bit kayan noktalı sayı formatında *x_t0, y_t0 ve z_t0* sinyallerinden almasını ve bundan sonraki tüm aşamalar için bu değerlerin *Euler* birimi çıkışından (*x_e_out, y_e_out, z_e_out*) alınmasını sağlamaktır. *Euler* biriminden gelen bir bitlik *sh_y* sinyali kaotik sistem sonuç ürettiği durumlarda '1', bunun dışındaki tüm durumlarda '0' değerini vermektedir. Bu şekilde *Euler* birimi ilk değerlerini ürettiğinde, *sh_y* sinyali '1' olmakta ve bu sinyali *Multiplexer* birimine göndererek kullanıcı tarafından atanan başlangıç değerleri yerine kaotik sistemin ürettiği değerleri kullanmasını sağlamaktadır. *Multiplexer* ünitesinin işlevi *Basla* sinyali geldiğinde sistemin ihtiyaç duyduğu başlangıç şartlarının atanmasını sağlamaktır. Diğer bir ifade ile sisteme kullanıcı tarafından atanan başlangıç şartları ile sistemin çıkışından elde edilen ve bir sonraki algoritmanın hesaplanmasında başlangıç şartları olarak kullanılan *x(k+1)*, *y(k+1)* ve *z(k+1)* sinyalleri arasında seçim yaparak bu sinyalleri sisteme göndermektir.



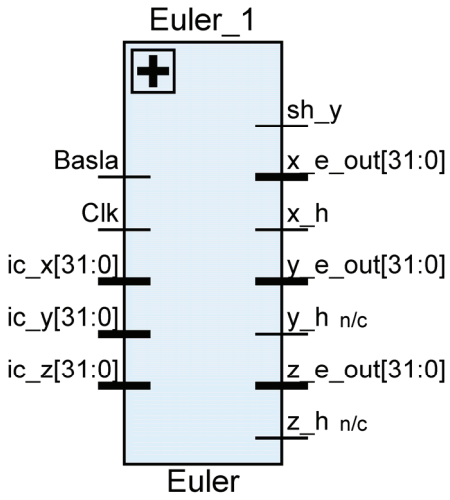
Şekil 5.15. Multiplexer biriminin birinci seviye blok diyagramı

Tasarımda *Euler* biriminin içinde yeni kaotik sistemin Euler algoritması ile sayısal çözümü yaptırılmaktadır. Bölüm 5.1.'de Euler algoritmasında bahsedilen işlemler *Euler* birimi içinde yaptırılmaktadır. Şekil 5.16.'da *Euler* biriminin çalışması daha anlaşılır şekilde blok şema olarak verilmiştir.

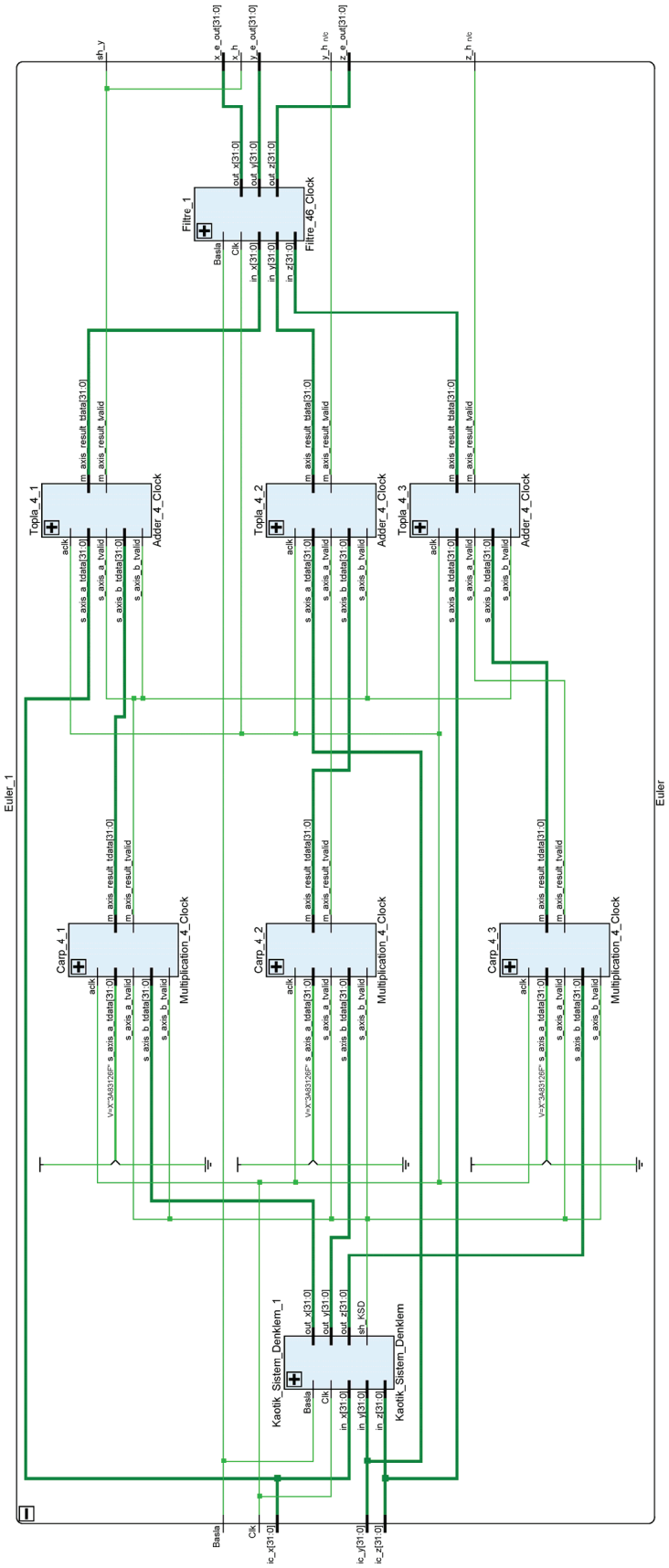


Şekil 5.16. FPGA tasarımında Euler algoritması blok şeması

Euler birimi, *Çarpma (Multiplication)*, *Toplama (Adder)*, *Filtre_46_Clock (Filter)* ve *Kaotik_Sistem_Denklem* birimlerini içermektedir. Şekil 5.17.'de *Euler* biriminin birinci seviye şeması, Şekil 5.18.'de ise *Euler* birimin ikinci seviye şeması yani iç yapısı verilmiştir.

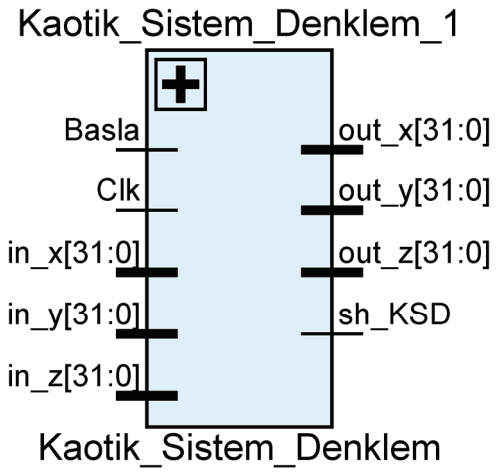


Şekil 5.17. Euler birimi birinci seviye şeması

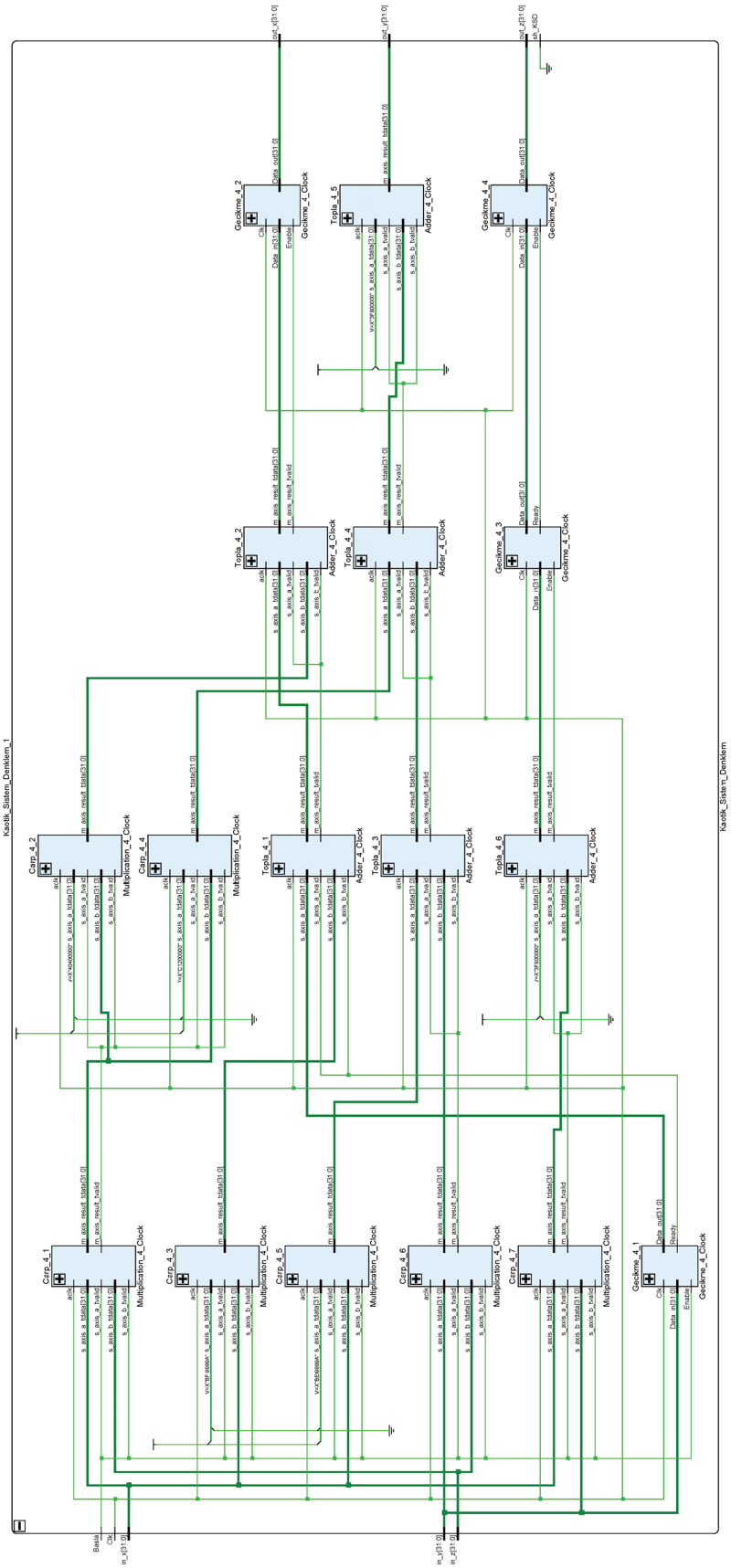


Şekil 5.18. Euler birimi iç yapısı

Multiplexer biriminin girişlerinden birisi olan *"Başlangıç Şartları"* yeni kaotik sistemin başlangıç şartlarını ifade etmektedir. Sistemdeki *Kaotik_Sistem_Denklem* birimi *Multiplexer* ünitesinden gelen sinyaller ile yeni kaotik sistem denklemlerinin hesaplanmasını sağlamaktadır. *Kaotik_Sistem_Denklem* biriminin birinci seviye ve ikinci seviye yani iç yapısı şemaları sırayla Şekil 5.19. ve Şekil 5.20.'de verilmiştir. Bu üniteden çıkan sinyaller ile algoritma adım sayısı olan *h* değeri *Çarpma* (Multiplication) birimi tarafından çarpılmaktadır. Ardından *Toplama* (Adder) birimi başlangıç şartı değerlerini toplayarak sonuçları *Filtre* birimine (Filtre_46_Clock) göndermektedir.

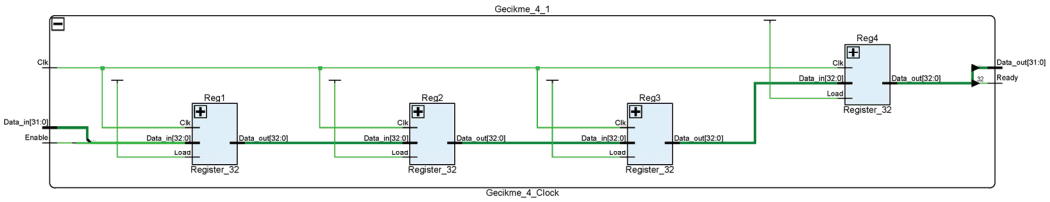


Şekil 5.19. Kaotik_Sistem_Denklem birimi birinci seviye şeması



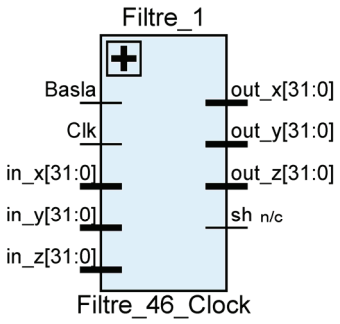
Şekil 5.20. Kaotik_Sistem_Denklem birimi iç yapısı

Şekil 5.20.'de iç yapısı verilen *Kaotik_Sistem_Denklem* birimi içinde yeni kaotik sistemin matematiksel hesabı işlemlerinin paralel olarak gerçekleştirilmesi esnasında gerekli olan dört saat darbesi süresi kadar gecikme sağlayan *Gecikme_4_Clock* birimi tasarlanmıştır. *Gecikme_4_Clock* biriminin yapısı Şekil 5.21.'de verilmiştir. Gecikme işlemi için her saat sinyalinde bilgi bir kaydediciden diğerine aktarılmaktadır. Bu şekilde dört saat darbesi sonunda giren bilgi çıkışa geciktirilerek aktarılmış olur.

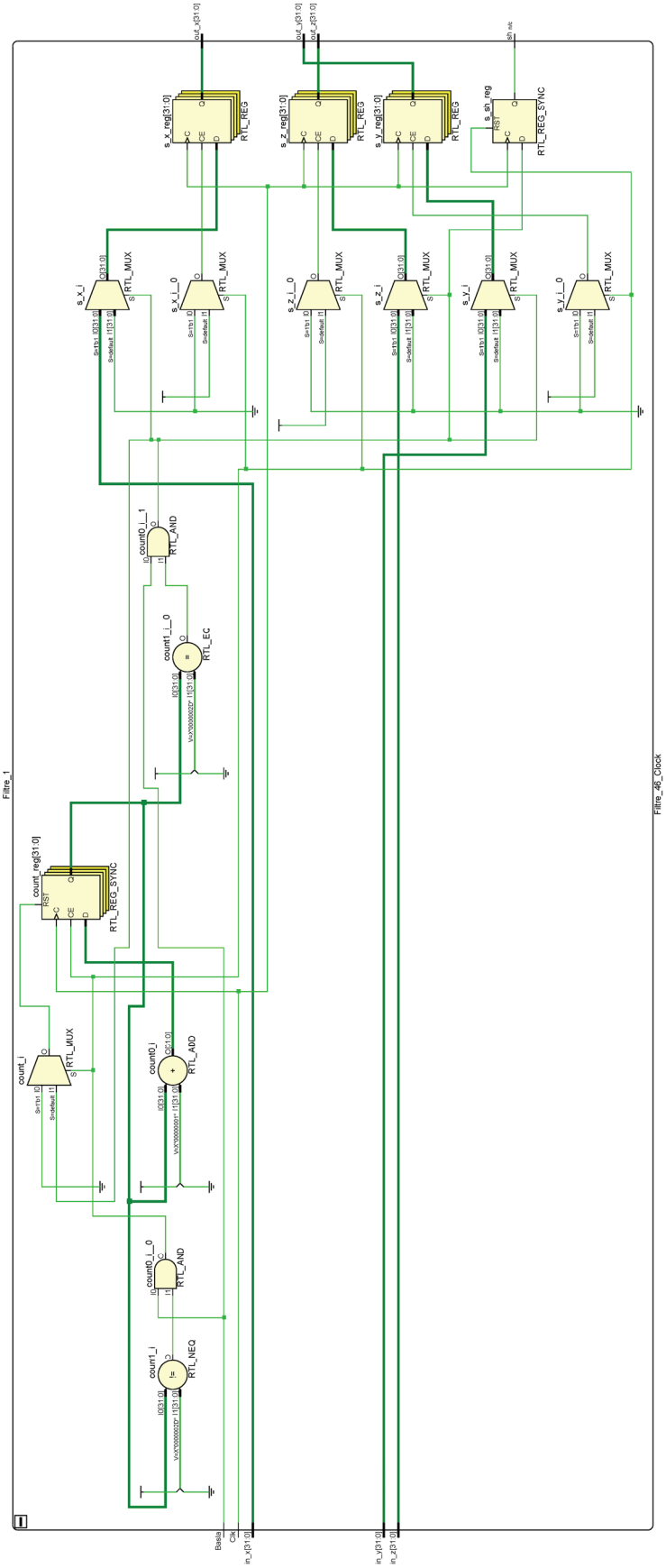


Şekil 5.21. Gecikme_4_Clock birimi iç yapısı

Sistemde *Filtre* birimi, kaotik sistemin istenmeyen sinyaller üretmesini engellemek diğer bir ifade ile filtrelemek amacıyla kullanılmaktadır. *Filtre* biriminin birinci seviye ve ikinci seviye yani iç yapısı şemaları sırayla Şekil 5.22. ve Şekil 5.23.'te verilmiştir. Sistem paralel olarak çalışmakta ve *Euler* birimi 46 saat darbesi (clock) sonunda ilk değerini üretmektedir. Filtre birimi 45 saat darbesi (clock) süresince sistem çıkışına değer göndermemekte, 46. saat darbesinde giriş sinyallerini çıkışa aktarmaktadır. Bu sayede Euler hesabı yapılana kadar sistem çıkışına istenmeyen değerlerin gönderilmesi engellenmiş olmaktadır. Diğer bir ifadeyle *Filtre* birimi her 45 saat darbesi boyunca önceki değeri hep çıkışa vermekte, 46. saat darbesinde ise yeni hesaplanan sonucu çıkışa aktarmaktadır.

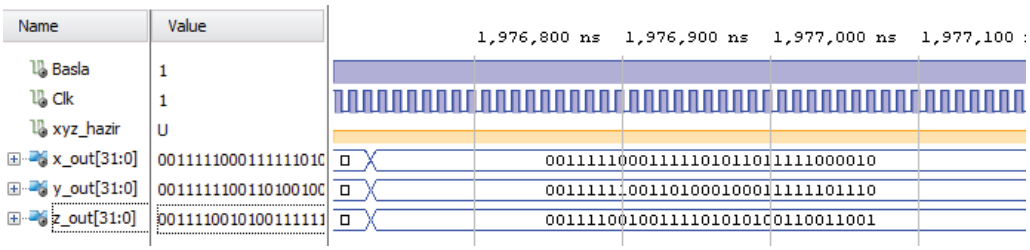


Şekil 5.22. Filtre birimi birinci seviye şeması



Şekil 5.23. Filtre birimi iç yapısı

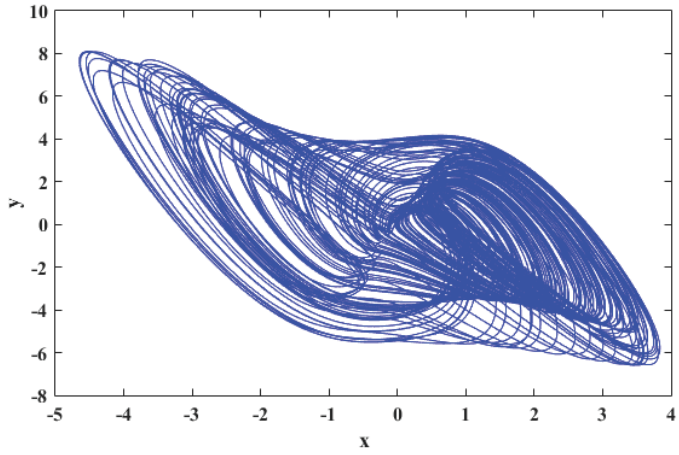
FPGA üzerinde gerçekleştirilen tasarım sentezleme (Synthesis) ve gerçekleme (Implementation) işlemlerine tabi tutulmuş ve çıkış değerleri ile FPGA çip kaynak kullanımları ve çalışma saat hızına ait parametrelerin değerleri incelenmiştir. Tasarımın benzetimi, Xilinx Vivado Design Suite 2015.4 programı simülatöründe yapılmıştır. Benzetim işleminden örnek bir görüntü Şekil 5.24.'te verilmiştir.



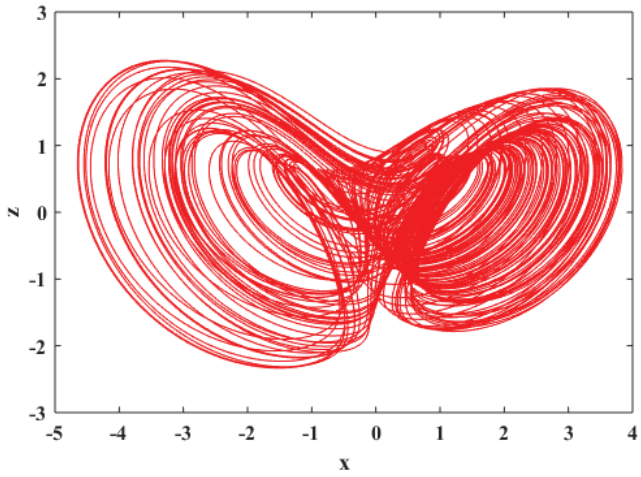
Şekil 5.24. Tasarlanan FPGA tabanlı yeni kaotik sistemin çıkış değerlerinden örnek bir görüntü

Yeni kaotik sistemin FPGA üzerinde gerçeklemesinden elde edilen *x_out*, *y_out* ve *z_out* sinyallerinin zaman serilerine ait 32-bit IEEE–754 formatındaki ikilik değerler benzetim test aşamasında dosyaya kaydedilmiştir. Kaydedilen değerler gerçek sayı sistemine dönüştürüldükten sonra kaotik osilatörün ürettiği ilk 300,000 adet veri alınarak *x_out*, *y_out* ve *z_out* sinyallerinin faz portreleri MATLAB ortamında çizdirilmiştir.

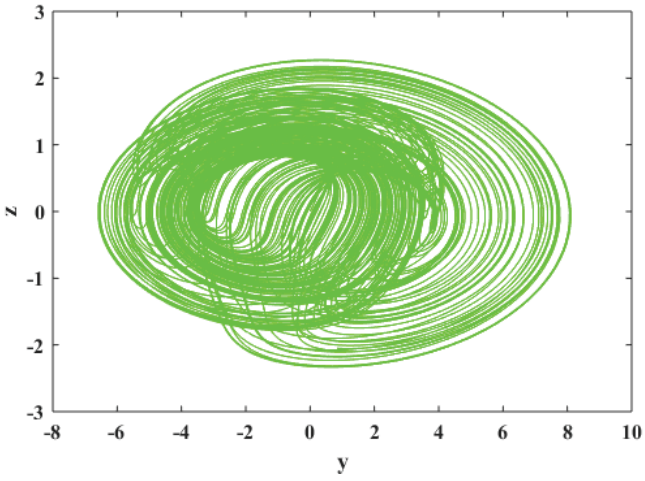
Şekil 5.25.'te yeni kaotik sistemin FPGA üzerinde gerçeklemesinden elde edilen faz portreleri verilmiştir. Faz portrelerinden de görüleceği üzere FPGA ile elde edilen sistem çıkışları ile (Şekil 5.25.) nümerik olarak elde edilen sistem çıkışları (Şekil 5.4.) birbirinin aynısıdır.



(a)



(b)



(c)

Şekil 5.25. Yeni kaotik sistemin FPGA tabanlı tasarımından elde edilen durum değişkenleri faz portreleri (a) x-y (b) x-z (c) y-z

Tablo 5.2.'de yeni kaotik sistemin Xilinx Artix-7 ailesi xc7a100tcsg324-1 modeli üzerinde gerçeklenen FPGA tasarımının çip istatistikleri verilmiştir. Sistemde maksimum çalışma frekansı 392.927 MHz ve minimum çalışma periyodu 2,545ns olarak elde edilmiştir. Tablo 5.2.'de verilen kısaltmaların açılımları şu şekildedir: LUT (Look-up Table, Başvuru Tablosu), LUTRAM (Look-up Table RAM), FF (Flip-Flop), DSP (Digital Signal Processor, Sayısal Sinyal İşleyici), IO (Input/Output, Giriş/Çıkış pin sayısı).

Tablo 5.2. Yeni kaotik sistemin FPGA tasarımı sonucu çip istatistikleri

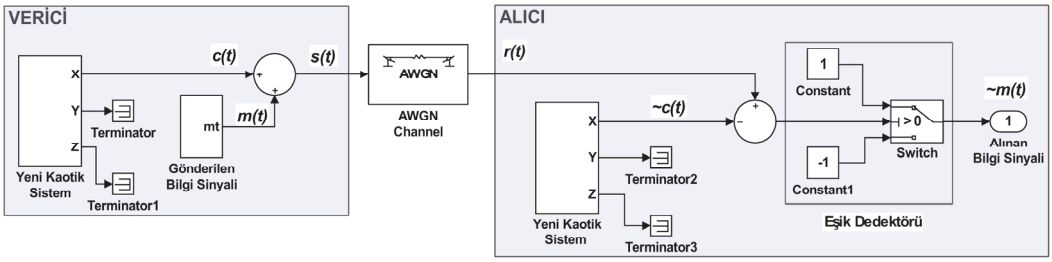
Kaynaklar	Mevcut	Kullanılan	Kullanım Oranı (%)
LUT	63400	2567	4,05
LUTRAM	19000	50	0,26
FF	126800	2229	1,76
DSP	240	48	20
IO	210	99	47,14

BÖLÜM 6. YENİ KAOTİK SİSTEM İLE KAOS TABANLI SAYISAL HABERLEŞME SİSTEMİ ÇALIŞMALARI

Bu bölümde tez çalışması kapsamında elde edilen yeni kaotik sistem kullanılarak Bölüm 3.'te verilen çeşitli kaos tabanlı sayısal haberleşme yöntemleri ile kaotik haberleşme sistemi tasarımları üzerine yapılan benzetim çalışmaları incelenmiştir.

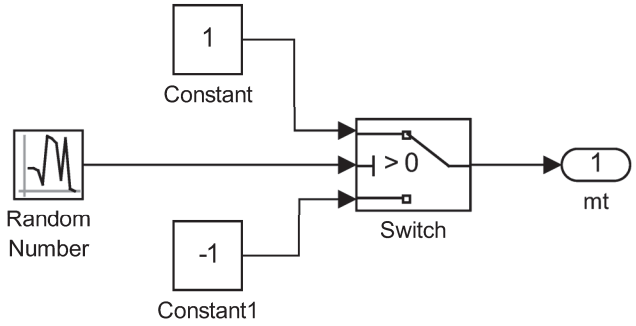
6.1. Yeni Kaotik Sistemin Kaotik Maskeleme Yöntemi İle Haberleşme Sistemi Tasarımı

Yeni kaotik sistem kullanılarak tasarlanan evre uyumlu (coherent) kaotik maskeleme (KM) (Chaotic Masking - CM) yöntemiyle tasarlanan haberleşme sisteminin Matlab-Simulink blok şeması Şekil 6.1.'de verilmiştir. KM yöntemi ile ilgili ayrıntılı teorik bilgi Bölüm 3.2.1.'de verilmişti. Şekil 6.1.'de görüldüğü üzere verici biriminde yeni kaotik sistem çıkışı $c(t)$, rastgele üretilen bilgi sinyali $m(t)$ ile KM yöntemine göre modülasyona tabi tutulmuştur. Modülasyon sonucu alıcı birimine KM modülasyonlu sinyal $s(t)$ gönderilmiştir. Alıcı biriminde verici birimindeki kaotik sistemin aynısı tekrardan oluşturulmuş ve verici birimden gelen gürültülü KM modülasyonlu sinyalden $\tilde{r}(t)$, alıcı birimdeki kaotik sistem çıkış sinyali $\tilde{c}(t)$ çıkartılmıştır. Elde edilen demodülasyon sinyali eşik dedektörü devresinden geçirilerek verici taraftan gönderilen bilgi sinyali $\tilde{m}(t)$ elde edilmiştir.



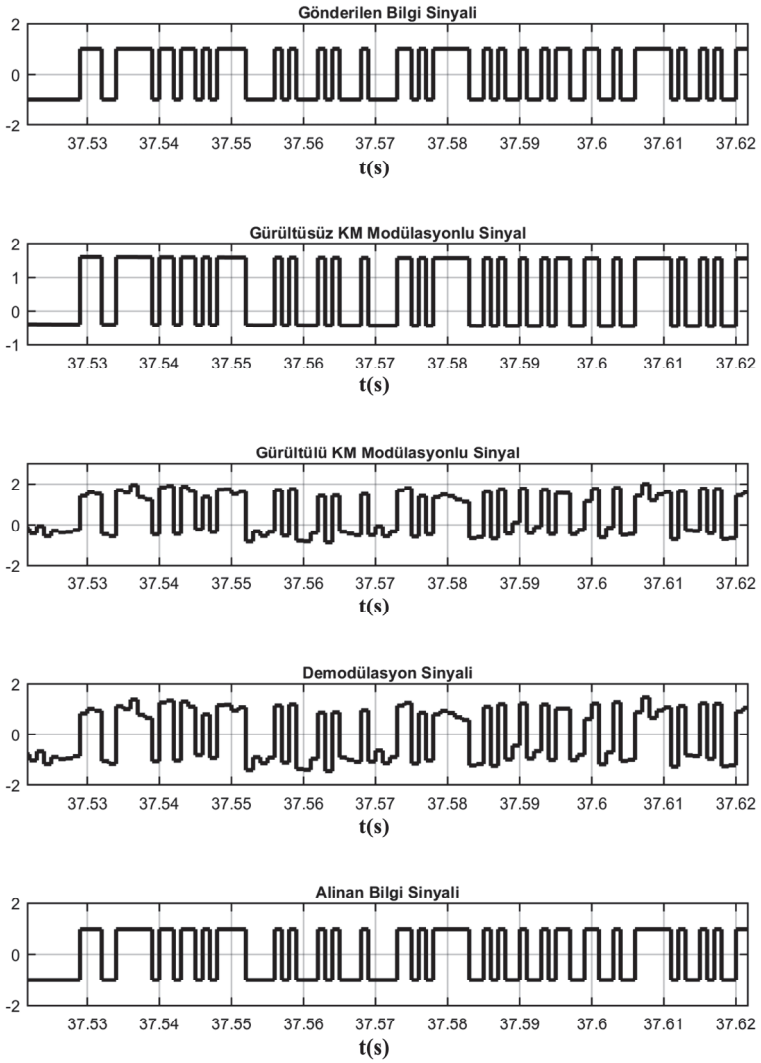
Şekil 6.1. Yeni kaotik sistemin KM yöntemi ile haberleşme sistemi tasarımının Matlab-Simulink blok şeması

Bu bölümdeki tüm haberleşme benzetim çalışmalarında rastgele bilgi sinyalinin üretilmesi için Şekil 6.2.'de verilen yapı kullanılmıştır. Bu yapı ile rastgele sayı üretecinden gelen değerler eşik dedektöründen geçirilerek çıkışta rastgele "1" ve "-1" bilgileri elde edilmiştir.

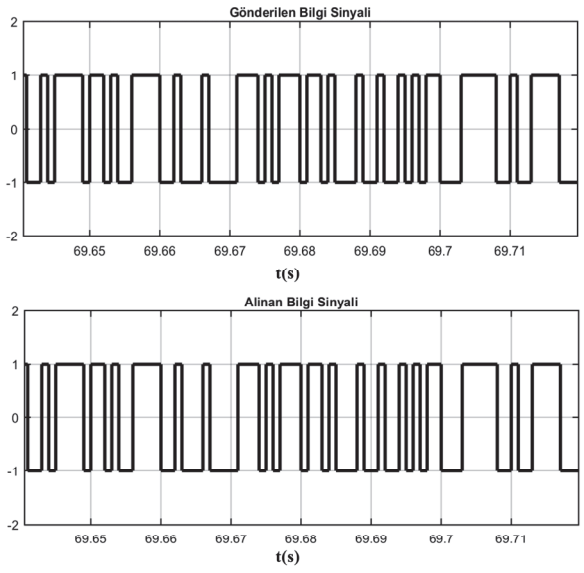


Şekil 6.2. Haberleşme benzetim çalışmalarında kullanılan rastgele bilgi sinyali üreteci Matlab-Simulink blok şeması

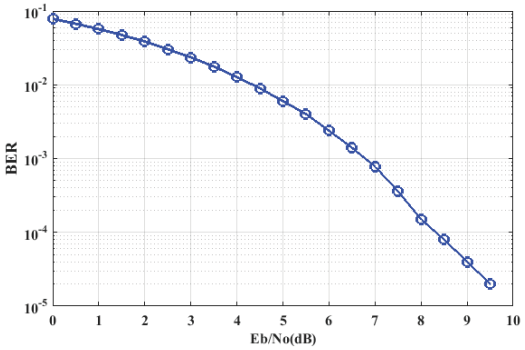
Şekil 6.1.'de verilen KM yöntemiyle yapılan kaos tabanlı haberleşme sisteminin analizi 0-20dB arasındaki E_b/N_0 (Bit enerjisinin gürültü enerjisine oranı) değerlerinde AWGN (Additive White Gaussian Noise – Toplanır Beyaz Gauss Gürültüsü) kanal modeli altında test edilmiştir. Benzetim 100 saniye süre ile çalıştırılmıştır. Sistemde bit süresi $T_b = 0,001$ saniye alınmıştır. Şekil 6.3.'te 10dB AWGN kanal gürültüsü altında benzetimi yapılan haberleşme sisteminin, sırayla gönderilen bilgi sinyali, gürültüsüz KM modülasyonlu sinyal, gürültülü KM modülasyonlu sinyal, demodülasyon sinyali ve alınan bilgi sinyalleri verilmiştir. Şekil 6.4.'te ise verici birimden gönderilen bilgi sinyali ile alıcı birimden alınan bilgi sinyali daha net olarak görülmektedir. Şekil 6.4. incelendiğinde verici taraftan gönderilen bilginin aynen alıcı taraftan alındığı görülmektedir. Tasarlanan haberleşme sisteminin 0-20dB arasındaki E_b/N_0 değerlerindeki AWGN kanal modeli altındaki BER (Bit Error Rate – Bit Hata Oranı) performansı grafiği Şekil 6.5.'te verilmiştir.



Şekil 6.3. Yeni kaotik sistemin KM yöntemi ile haberleşme sistemi tasarımının sinyalleri



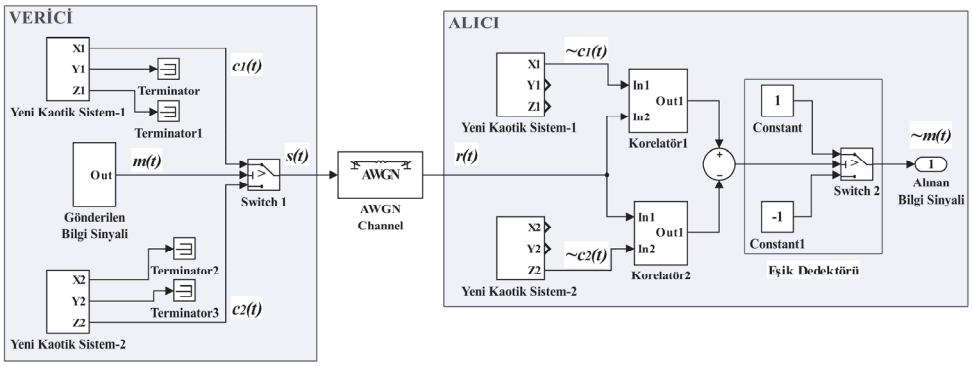
Şekil 6.4. Yeni kaotik sistemin KM yöntemi ile haberleşme sisteminde gönderilen ve alınan bilgi sinyalleri



Şekil 6.5. Yeni kaotik sistemin KM yöntemi ile tasarlanan haberleşme sistemi BER performansı

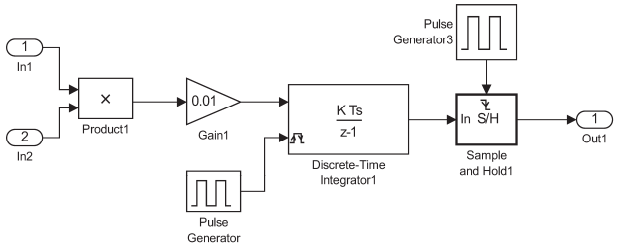
6.2. Yeni Kaotik Sistemin Evre Uyumlu Kaos Kaydırmalı Anahtarlama Yöntemi İle Haberleşme Sistemi Tasarımı

Yeni kaotik sistem kullanılarak tasarlanan evre uyumlu kaos kaydırmalı anahtarlama (KKA) (Chaos Shift Keying - CSK) yöntemiyle tasarlanan haberleşme sisteminin Matlab-Simulink blok şeması Şekil 6.6.'da verilmiştir. KKA yöntemi ile ilgili ayrıntılı teorik bilgi Bölüm 3.2.2.'de verilmişti. Şekil 6.6.'da görüldüğü üzere verici biriminde iki farklı kaotik sistem için yeni kaotik sistemden iki tane kullanılmıştır. İki farklı kaotik sinyal için birinci kaotik sistemin x çıkışı $c_1(t)$, ikinci kaotik sistemin ise z çıkışı $c_2(t)$ kullanılmıştır. Aynı kaotik sistemden iki farklı kaotik çıkış elde etmek için ayrıca birinci kaotik sistemde x çıkışı 70, ikinci kaotik sistemde z çıkışı 0,1 ile çarpılarak çıkışa aktarılmıştır. Rastgele üretilen bilgi sinyali $m(t)$ "1" değerinde ise alıcı birime $c_1(t)$ kaotik sinyali, "-1" değerinde ise alıcı birime $c_2(t)$ kaotik sinyali gönderilmiştir.



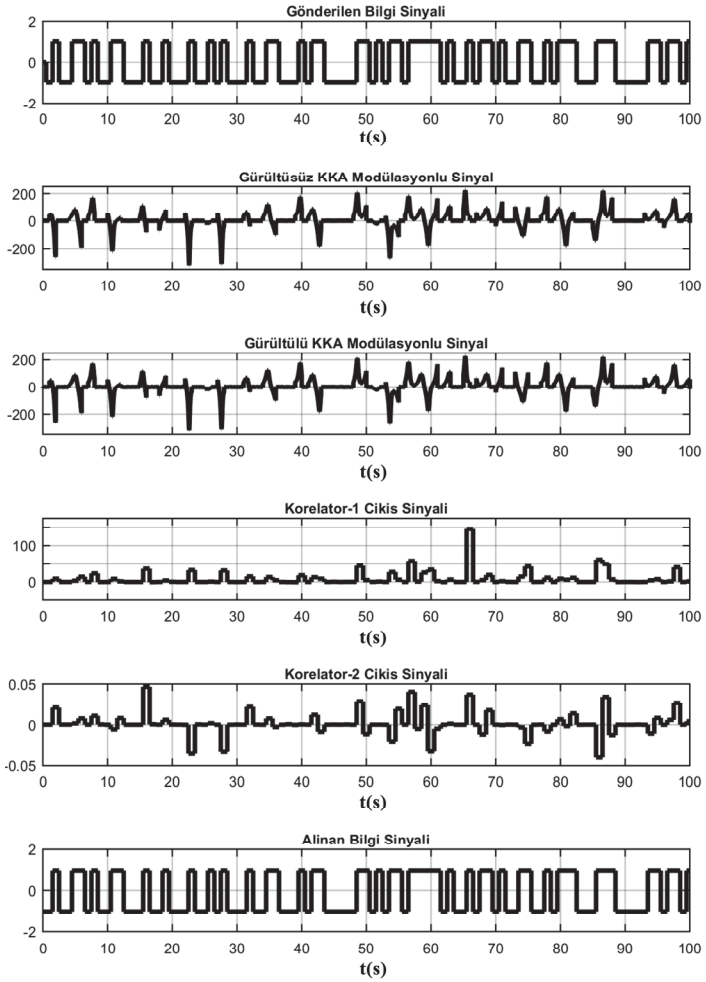
Şekil 6.6. Yeni kaotik sistemin evre uyumlu KKA yöntemi ile haberleşme sistemi tasarımının Matlab-Simulink blok şeması

Alıcı birimde verici birimdeki iki kaotik sistem aynı şekilde oluşturulmuştur. Alıcı birime gelen gürültülü modülasyonlu sinyal $r(t)$, her iki kaotik sistemden gelen kaotik sinyaller ile ($\tilde{c}_1(t)$, $\tilde{c}_2(t)$) korelatör birimlerine girerek burada sinyallerin bit enerjisi hesaplanmıştır. Bu bölümdeki tüm haberleşme benzetim çalışmalarında korelatör birimi olarak Şekil 6.7.'de verilen yapı kullanılmıştır. Korelatör-1 ve Korelatör-2 çıkışlarının farkı alınmış ve bu fark değeri gelen bilginin "1" veya "-1" olup olmadığının tespiti için eşik dedektörü birimine gönderilmiştir. Eşik dedektörü devresinin çıkışından elde edilen sinyal ile verici taraftan gönderilen bilgi sinyali $\tilde{m}(t)$ elde edilmiştir.

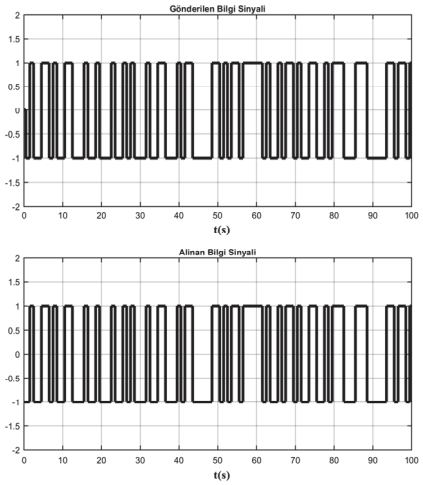


Şekil 6.7. Haberleşme benzetim çalışmalarında kullanılan korelatör birimi Matlab-Simulink blok şeması

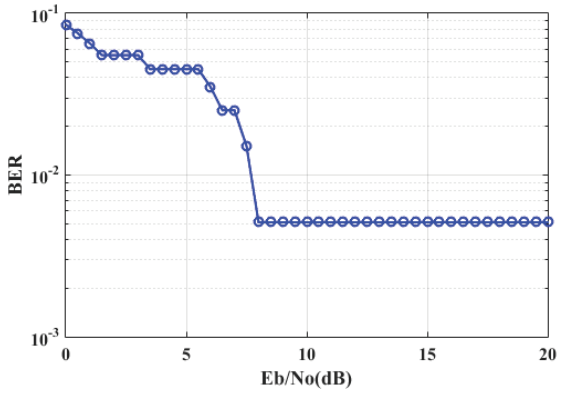
Şekil 6.6.'da verilen evre uyumlu KKA yöntemiyle yapılan kaos tabanlı haberleşme sisteminin analizi 0-20dB arasındaki E_b/N_0 değerlerinde AWGN kanal modeli altında test edilmiştir. Benzetim 100 saniye süre ile çalıştırılmıştır. Sistemde bit süresi $T_b = 0{,}01$ saniye alınmıştır. Şekil 6.8.'de 10dB AWGN kanal gürültüsü altında benzetimi yapılan haberleşme sisteminin, sırayla gönderilen bilgi sinyali, gürültüsüz KKA modülasyonlu sinyal, gürültülü KKA modülasyonlu sinyal, demodülasyon sinyali ve alınan bilgi sinyalleri verilmiştir. Sonuçlar alınırken alınan sinyal 0,5 saniye geciktirilmiştir. Bu fark alınan sinyalin işlenmesi esnasında geçen işlem süresi içindir. Şekil 6.9.'da ise verici birimden gönderilen bilgi sinyali ile alıcı birimden alınan bilgi sinyali daha net olarak görülmektedir. Şekil 6.9. incelendiğinde verici taraftan gönderilen bilginin aynen alıcı taraftan alındığı görülmektedir. Tasarlanan haberleşme sisteminin 0-20dB arasındaki E_b/N_0 değerlerindeki AWGN kanal modeli altındaki BER performansı grafiği Şekil 6.10.'da verilmiştir.



Şekil 6.8. Yeni kaotik sistemin evre uyumlu KKA yöntemi ile haberleşme sistemi tasarımının sinyalleri



Şekil 6.9. Yeni kaotik sistemin evre uyumlu KKA yöntemi ile haberleşme sisteminde gönderilen ve alınan bilgi sinyalleri

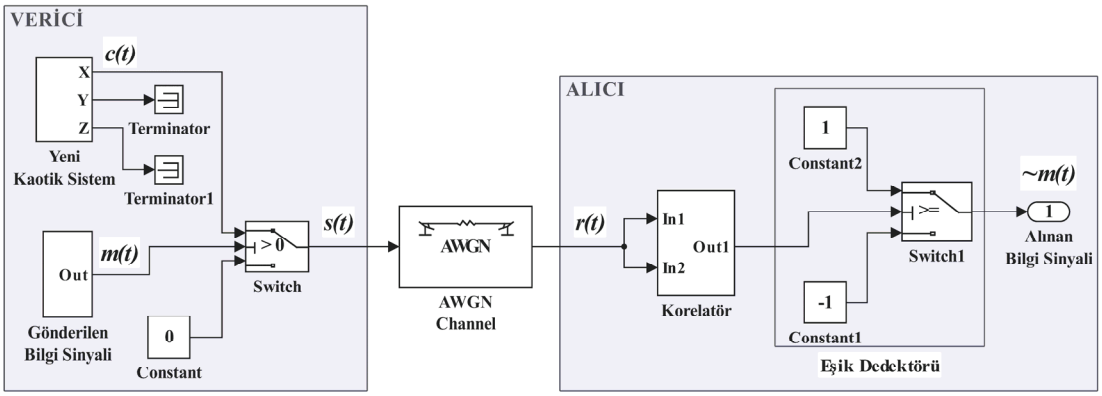


Şekil 6.10. Yeni kaotik sistemin evre uyumlu KKA yöntemi ile tasarlanan haberleşme sistemi BER performansı

6.3. Yeni Kaotik Sistemin Kaotik Açma Kapama Anahtarlama Yöntemi İle Haberleşme Sistemi Tasarımı

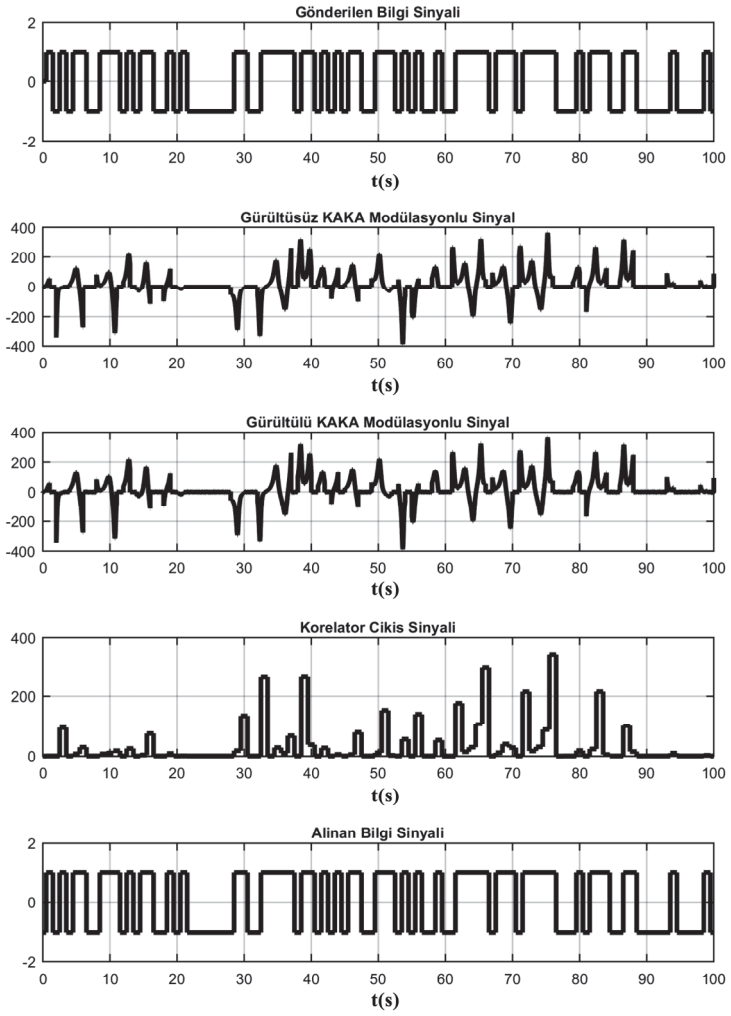
Yeni kaotik sistem kullanılarak tasarlanan kaotik açma kapama anahtarlama (KAKA) (Chaotic On-Off Keying - COOK) yöntemiyle tasarlanan haberleşme sisteminin Matlab-Simulink blok şeması Şekil 6.11'de verilmiştir. KAKA yöntemi ile ilgili ayrıntılı teorik bilgi Bölüm 3.2.3.'te verilmişti. Şekil 6.11.'de görüldüğü üzere verici biriminde gönderilen bilgi sinyali eğer "1" ise alıcı birimine kaotik sistem çıkış sinyali $c(t)$ gönderilmiş, eğer bilgi sinyali "-1" ise alıcı birime "0" sinyali gönderilmiştir. Bu şekilde alıcı birime KAKA modülasyonlu sinyal $s(t)$ gönderilmektedir. Kaotik sinyal çıkış değerleri çok küçük olduğu için x durum değişkeni hesaplatıldıktan sonra 100 ile çarpılarak genliği artırılmış ve bu şekilde çıkışa aktarılmıştır.

Alıcı birime gelen gürültülü KAKA modülasyonlu sinyalin $r(t)$ korelatör biriminde sinyalin enerji değeri hesaplanmıştır. Korelatör birimi daha önce Şekil 6.7.'de verilen yapının aynısıdır. Korelatör çıkış sinyali eşik dedektörüne gönderilmiş ve eşik seviyesine göre verici birimden gönderilen bilgi sinyali $\tilde{m}(t)$ elde edilmiştir.

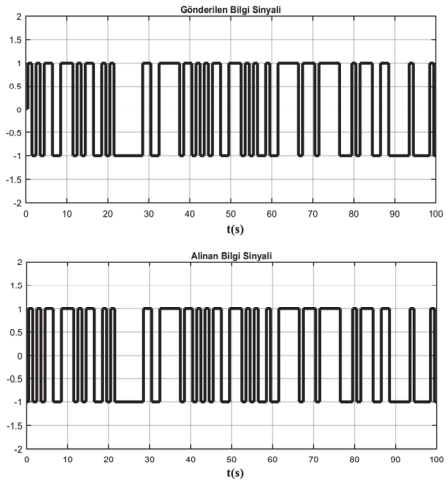


Şekil 6.11. Yeni kaotik sistemin KAKA yöntemi ile haberleşme sistemi tasarımının Matlab-Simulink blok şeması

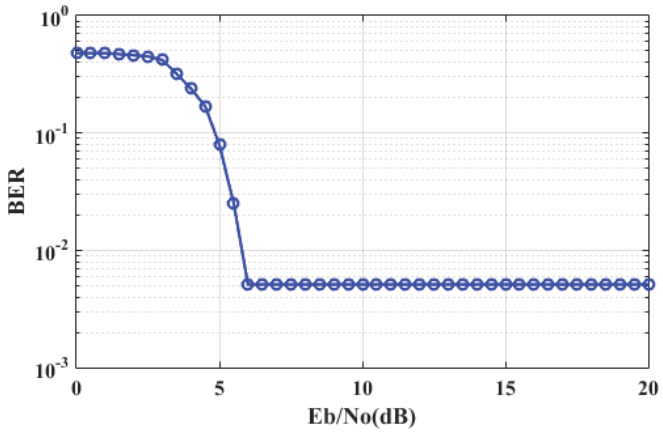
Şekil 6.11.'de verilen KAKA yöntemiyle yapılan kaos tabanlı haberleşme sisteminin analizi 0-20dB arasındaki E_b/N_0 değerlerinde AWGN kanal modeli altında test edilmiştir. Benzetim 100 saniye süre ile çalıştırılmıştır. Sistemde bit süresi $T_b = 0,01$ saniye alınmıştır. Şekil 6.12.'de 10dB AWGN kanal gürültüsü altında benzetimi yapılan haberleşme sisteminin, sırayla gönderilen bilgi sinyali, gürültüsüz KAKA modülasyonlu sinyal, gürültülü KAKA modülasyonlu sinyal, demodülasyon sinyali ve alınan bilgi sinyalleri verilmiştir. Sonuçlar alınırken alınan sinyal 0,5 saniye geciktirilmiştir. Bu fark alınan sinyalin işlenmesini esnasında geçen işlem süresi içindir. Şekil 6.13.'te ise verici birimden gönderilen bilgi sinyali ile alıcı birimden alınan bilgi sinyali daha net olarak görülmektedir. Şekil 6.13. incelendiğinde verici taraftan gönderilen bilginin aynen alıcı taraftan alındığı görülmektedir. Tasarlanan haberleşme sisteminin 0-20dB arasındaki E_b/N_0 değerlerindeki AWGN kanal modeli altındaki BER performansı grafiği Şekil 6.14.'te verilmiştir.



Şekil 6.12. Yeni kaotik sistemin KAKA yöntemi ile haberleşme sistemi tasarımının sinyalleri



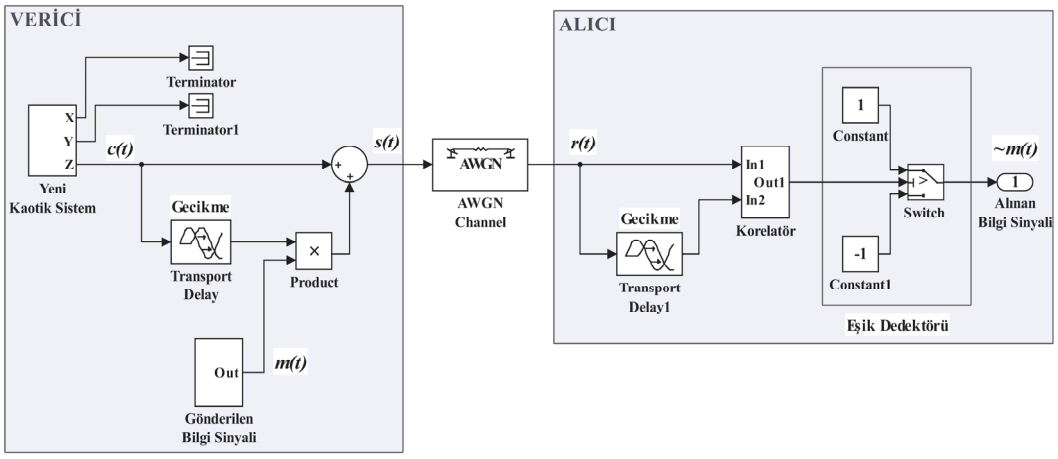
Şekil 6.13. Yeni kaotik sistemin KAKA yöntemi ile haberleşme sisteminde gönderilen ve alınan bilgi sinyalleri



Şekil 6.14. Yeni kaotik sistemin KAKA yöntemi ile tasarlanan haberleşme sistemi BER performansı

6.4. Yeni Kaotik Sistemin Korelasyon Gecikmeli Kaydırmalı Anahtarlama Yöntemi İle Haberleşme Sistemi Tasarımı

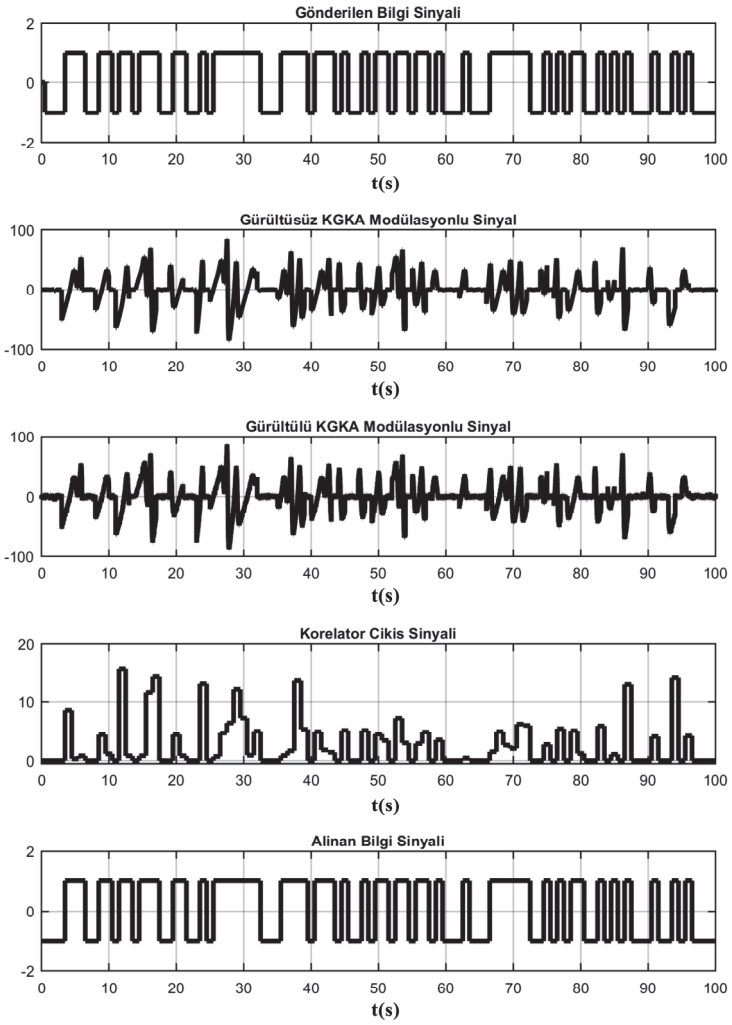
Yeni kaotik sistem kullanılarak tasarlanan korelasyon gecikmeli kaydırmalı anahtarlama (KGKA) (Correlation Delay Shift Keying - CDSK) yöntemiyle tasarlanan haberleşme sisteminin Matlab-Simulink blok şeması Şekil 6.15.'te verilmiştir. KGKA yöntemi ile ilgili ayrıntılı teorik bilgi Bölüm 3.2.6.'da verilmişti. Şekil 6.15.'te görüldüğü üzere verici biriminde yeni kaotik sistem oluşturulmuş ve bu kaotik sistemin z durum değişkeni sinyali haberleşme için kullanılmıştır. Kaotik sinyal çıkış değerleri çok küçük olduğu için z durum değişkeni hesaplatıldıktan sonra 20 ile çarpılarak genliği artırılmış ve bu şekilde çıkışa aktarılmıştır. Verici biriminde, kaotik sinyalin bit periyodunun yarısı süresi kadar ($T_b/2$) geciktirilmiş haliyle çarpılan bilgi sinyali $m(t)$ ile kaotik sinyal $c(t)$ toplanmıştır. Bu şekilde elde edilen modülasyonlu sinyal $s(t)$ alıcı birimine gönderilmiştir.



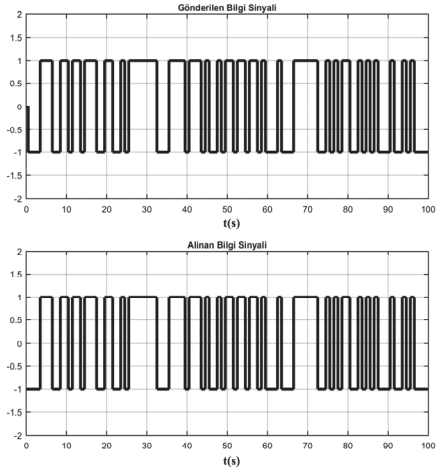
Şekil 6.15. Yeni kaotik sistemin KGKA yöntemi ile haberleşme sistemi tasarımının Matlab-Simulink blok şeması

Alıcı birime gelen gürültülü modülasyon sinyali $r(t)$ ve bu sinyalin bit periyodunun yarısı süresi kadar ($T_b/2$) geciktirilmiş hali korelatör birimine gönderilmiş ve burada gelen sinyalin enerji değeri hesaplanmıştır. Korelatör birimi daha önce Şekil 6.7.'de verilen yapının aynısıdır. Korelatör çıkış sinyali eşik dedektörüne gönderilmiş ve eşik seviyesine göre verici birimden gönderilen bilgi sinyali $\tilde{m}(t)$ elde edilmiştir.

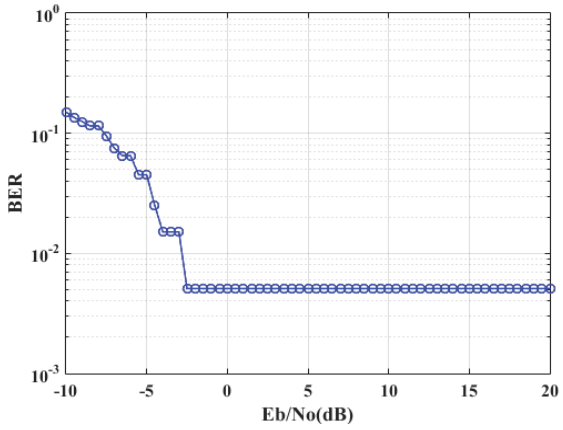
Şekil 6.15.'te verilen KGKA yöntemiyle yapılan kaos tabanlı haberleşme sisteminin analizi -10dB ile 20dB arasındaki E_b/N_0 değerlerinde AWGN kanal modeli altında test edilmiştir. Benzetim 100 saniye süre ile çalıştırılmıştır. Sistemde bit süresi $T_b = 0,01$ saniye alınmıştır. Şekil 6.16'da 10dB AWGN kanal gürültüsü altında benzetimi yapılan haberleşme sisteminin, sırayla gönderilen bilgi sinyali, gürültüsüz KGKA modülasyonlu sinyal, gürültülü KGKA modülasyonlu sinyal, demodülasyon sinyali ve alınan bilgi sinyalleri verilmiştir. Sonuçlar alınırken alınan sinyal 0,5 saniye geciktirilmiştir. Bu fark alınan sinyalin işlenmesini esnasında geçen işlem süresi içindir. Şekil 6.17.'de ise verici birimden gönderilen bilgi sinyali ile alıcı birimden alınan bilgi sinyali daha net olarak görülmektedir. Şekil 6.17. incelendiğinde verici taraftan gönderilen bilginin aynen alıcı taraftan alındığı görülmektedir. Tasarlanan haberleşme sisteminin -10dB ile 20dB arasındaki E_b/N_0 değerlerindeki AWGN kanal modeli altındaki BER performansı grafiği Şekil 6.18.'de verilmiştir.



Şekil 6.16. Yeni kaotik sistemin KGKA yöntemi ile haberleşme sistemi tasarımının sinyalleri



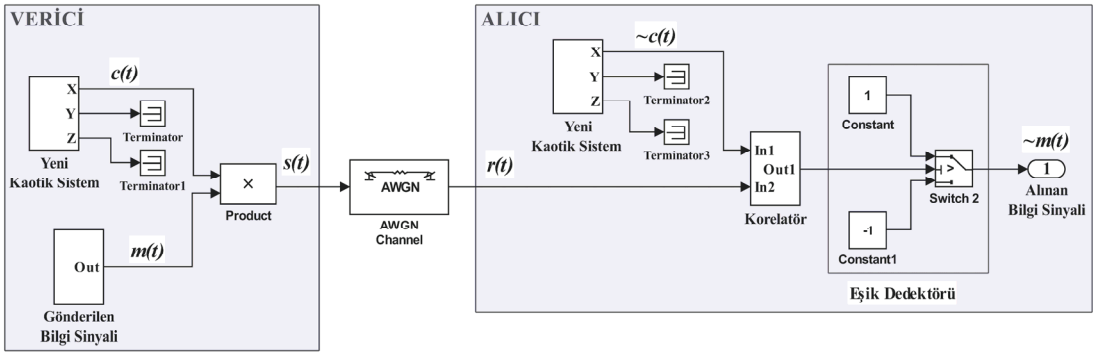
Şekil 6.17. Yeni kaotik sistemin KGKA yöntemi ile haberleşme sisteminde gönderilen ve alınan bilgi sinyalleri



Şekil 6.18. Yeni kaotik sistemin KGKA yöntemi ile tasarlanan haberleşme sistemi BER performansı

6.5. Yeni Kaotik Sistemin Simetrik Kaos Kaydırmalı Anahtarlama Yöntemi İle Haberleşme Sistemi Tasarımı

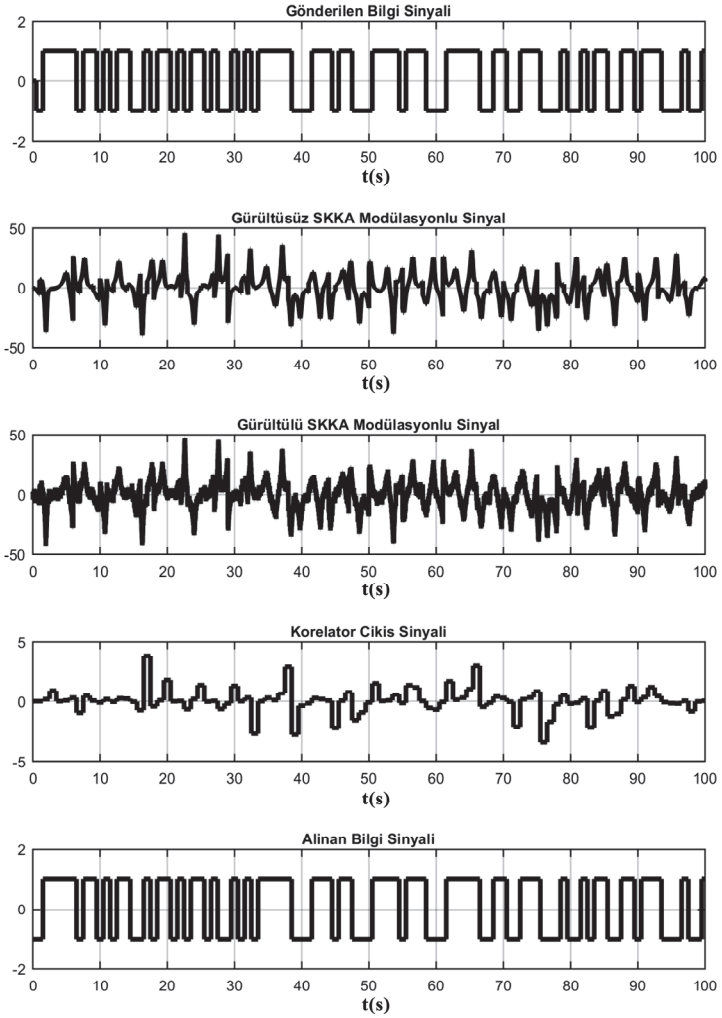
Yeni kaotik sistem kullanılarak tasarlanan simetrik kaos kaydırmalı anahtarlama (SKKA) (Symmetric Chaos Shift Keying - SCSK) yöntemiyle tasarlanan haberleşme sisteminin Matlab-Simulink blok şeması Şekil 6.19.'da verilmiştir. SKKA yöntemi ile ilgili ayrıntılı teorik bilgi Bölüm 3.2.7.'de verilmişti. Şekil 6.19.'da görüldüğü üzere verici biriminde yeni kaotik sistem oluşturulmuş ve bu kaotik sistemin x durum değişkeni sinyali haberleşme için kullanılmıştır. Kaotik sinyal çıkış değerleri çok küçük olduğu için x durum değişkeni hesaplatıldıktan sonra 10 ile çarpılarak genliği artırılmış ve bu şekilde çıkışa aktarılmıştır. Verici birimde bilgi sinyali $m(t)$ ile kaotik sistem sinyali $c(t)$ çarpılarak modülasyonlu sinyal $s(t)$ elde edilmiştir. Bu şekilde elde edilen modülasyonlu sinyal $s(t)$ alıcı birimine gönderilmiştir.



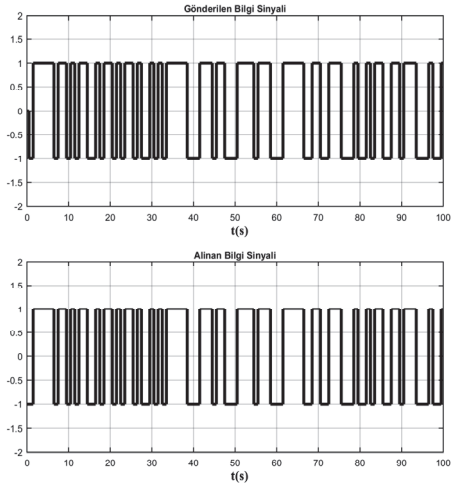
Şekil 6.19. Yeni kaotik sistemin SKKA yöntemi ile haberleşme sistemi tasarımının Matlab-Simulink blok şeması

Alıcı birimde verici birimdeki kaotik sistemin aynısı tekrar oluşturulmuştur. Alıcı birime gelen gürültülü modülasyon sinyali $r(t)$ ve kaotik sistem çıkışı sinyali $\tilde{c}(t)$ korelatör birimine girmiş ve burada gelen sinyalin enerji değeri hesaplanmıştır. Korelatör birimi daha önce Şekil 6.7.'de verilen yapının aynısıdır. Korelatör çıkış sinyali eşik dedektörüne gönderilmiş ve eşik seviyesine göre verici birimden gönderilen bilgi sinyali $\tilde{m}(t)$ elde edilmiştir.

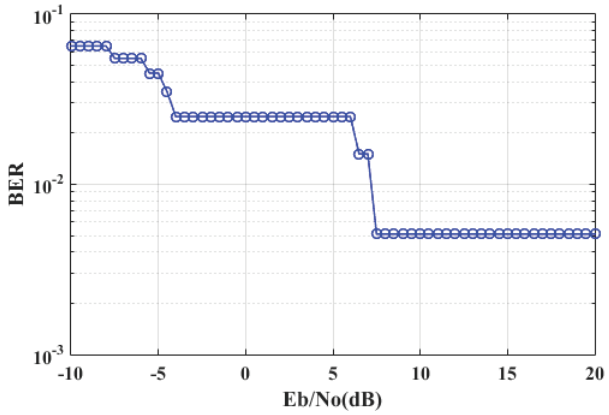
Şekil 6.19.'da verilen SKKA yöntemiyle yapılan kaos tabanlı haberleşme sisteminin analizi -10dB ile 20dB arasındaki E_b/N_0 değerlerinde AWGN kanal modeli altında test edilmiştir. Benzetim 100 saniye süre ile çalıştırılmıştır. Sistemde bit süresi $T_b = 0,01$ saniye alınmıştır. Şekil 6.20.'de 10dB AWGN kanal gürültüsü altında benzetimi yapılan haberleşme sisteminin, sırayla gönderilen bilgi sinyali, gürültüsüz SKKA modülasyonlu sinyal, gürültülü SKKA modülasyonlu sinyal, demodülasyon sinyali ve alınan bilgi sinyalleri verilmiştir. Sonuçlar alınırken alınan sinyal 0,5 saniye geciktirilmiştir. Bu fark alınan sinyalin işlenmesini esnasında geçen işlem süresi içindir. Şekil 6.21.'de ise verici birimden gönderilen bilgi sinyali ile alıcı birimden alınan bilgi sinyali daha net olarak görülmektedir. Şekil 6.21. incelendiğinde verici taraftan gönderilen bilginin aynen alıcı taraftan alındığı görülmektedir. Tasarlanan haberleşme sisteminin -10dB ile 20dB arasındaki E_b/N_0 değerlerindeki AWGN kanal modeli altındaki BER performansı grafiği Şekil 6.22.'de verilmiştir.



Şekil 6.20. Yeni kaotik sistemin SKKA yöntemi ile haberleşme sistemi tasarımının sinyalleri



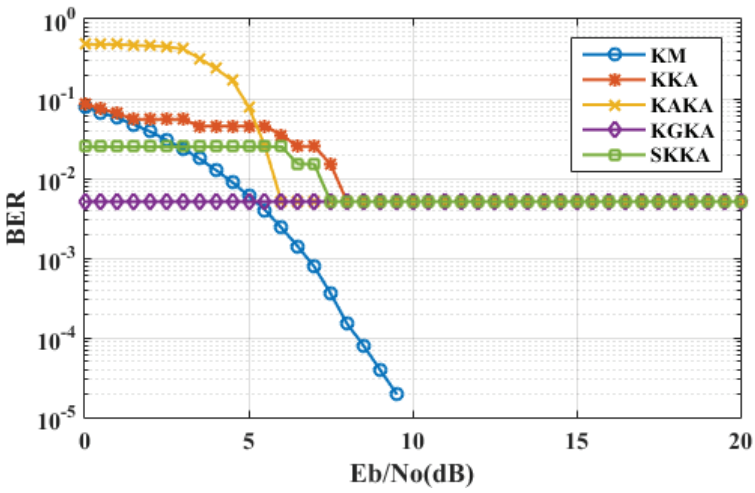
Şekil 6.21. Yeni kaotik sistemin SKKA yöntemi ile haberleşme sisteminde gönderilen ve alınan bilgi sinyalleri



Şekil 6.22. Yeni kaotik sistemin SKKA yöntemi ile tasarlanan haberleşme sistemi BER performansı

6.6. Tasarlanan Kaos Tabanlı Sayısal Haberleşme Sistemlerinin BER Performanslarının Karşılaştırılması

Yeni kaotik sistem ile tasarlanan kaos tabanlı KM, KKA, KAKA, KGKA ve SKKA yöntemli haberleşme sistemlerinin 0dB ile 20dB arasındaki E_b/N_0 değerlerinde AWGN kanal modeli altındaki BER performansları karşılaştırılmıştır. Yapılan karşılaştırma sonucu Şekil 6.23.'te verilmiştir. Şekil 6.23.'ten de görüleceği üzere aynı şartlar altında KM yönteminin BER performansı diğer yöntemlere göre daha iyidir.



Şekil 6.23. Yeni kaotik sistem ile tasarlanan KM, KKA, KAKA, KGKA, SKKA yöntemli haberleşme sistemlerinin BER performanslarının karşılaştırılması

BÖLÜM 7. YENİ KAOTİK SİSTEMDEN FPGA TABANLI GERÇEK RASTGELE SAYI ÜRETECİ TASARIMI

Bu bölümde tez çalışmasının amacı olan FPGA tabanlı şifreli kaotik haberleşme sisteminde kullanılacak şifreleme biriminde kullanılmak üzere gerçek rastgele sayı üreteci (GRSÜ) tasarımı çalışmaları gerçekleştirilmiştir. Tez çalışması kapsamında elde edilen yeni kaotik sistem kullanılarak ilk önce Matlab-Simulink ortamında GRSÜ tasarımı yapılmıştır. Tasarlanan GRSÜ çıkışından elde edilen verilerin rastgelelik kontrolü için FIPS 140-1 ve NIST 800-22 testleri gerçekleştirilmiştir. En son olarak da Matlab-Simulink ortamında yapılan GRSÜ tasarımı VHDL donanım tanımlama dili kullanılarak FPGA ortamında tasarlanmıştır. FPGA tabanlı tasarlanan GRSÜ biriminden elde edilen verilerin rastgelelik kontrolü için de aynı şekilde FIPS 140-1 ve NIST 800-22 testleri gerçekleştirilmiştir.

7.1. Rastgele Sayı Üreteçleri

Rastgele kelimesi Türk Dil Kurumu (TDK) sözlüğüne göre gelişigüzel anlamındadır. Rastgele sayı üreteçleri de çıkışında gelişigüzel yani rastgele sayı üreten, yazılım tabanlı veya donanım tabanlı olarak tasarlanabilen yapılardır. Rastgele sayıların aralarında herhangi bir ilişki bulunmaz ve dolayısıyla istatiksel olarak birbirinden bağımsızdırlar [180-182]. Rastgele sayılar; benzetim (simülasyon) ve modelleme, örnekleme, nümerik analiz, bilgisayar programlama, karar verme, şans oyunları, çekilişler, bilgisayar oyunları ve şifreleme (Kriptoloji) gibi alanlarda kullanılmaktadır [183].

Rastgele sayı üreteçleri üç ayrı sınıfta incelenebilir. Bunlar;

- Sözde Rastgele Sayı Üreteçleri (SRSÜ)
  a. Saf b. Hibrit
- Gerçek Rastgele Sayı Üreteçleri (GRSÜ)

a. Fiziksel b. Fiziksel Olmayan

– Hibrit Rastgele Sayı Üreteçleri (HRSÜ)

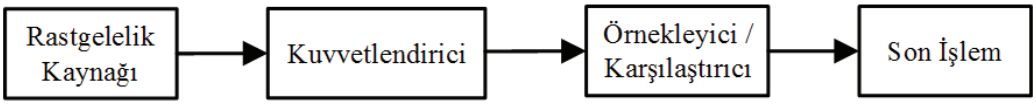
SRSÜ, belli bir başlangıç yani tohum (seed) değerine bağlı olarak deterministik bir denkleme ve algoritmaya göre rastgele sayı üretirler. SRSÜ'ler düşük maliyetle kolay şekilde gerçeklenebilirler. Fakat belli bir algoritmaya bağlı sayı ürettiklerinden dolayı belli bir zamandan sonra rastgelelik değeri düşmekte ve üretim algoritması öğrenildiğinde bir sonraki üreteci sayı değerleri kolayca tahmin edilebilmektedir. Bu durum şifreleme uygulamalarında bir dezavantaj olmaktadır. SRSÜ yapıları saf ve hibrit olmak üzere iki çeşittir. Saf SRSÜ yapılarında bir tohum değerinden hesapsal olarak diziler üretilir. Belli bir süre sonra üretilen diziler birbirinin tekrarı olabilir. Hibrit SRSÜ yapısında ise saf SRSÜ yapısına ek olarak tohum değeri girişine ek bir parametre değeri girdisi vardır. Bu ek parametre değeri sistemin tahmin edilmesini zorlaştırır ve rastgelelik özelliğini arttırır [180-184].

GRSÜ, deterministik olmayan bir kaynağı kullanarak rastgele sayı üretirler. Deterministik olmayan bu kaynaklar bir elektronik elemanın termal gürültüsü, bir osilatörün faz gürültüsü, bir bilgisayar işlemcisinin çalışma yükü değeri, bilgisayar kullanıcısının klavye veya fare kullanım değerleri, elektronik elemanların gürültüleri olabilmektedir. Bu sayede GRSÜ'lerinin ürettiği sayı değerlerinin rastgelelik derecesi daha yüksek olmaktadır. GRSÜ yapıları fiziksel ve fiziksel olmayan olmak üzere iki farklı yapıda olabilir. Fiziksel GRSÜ yapılarında rastgelelik kaynağı için bir donanım birimi parametresi kullanılır. Fiziksel olmayan GRSÜ yapılarında ise rastgelelik kaynağı için donanım yerine bir yazılımdan elde edilen veriler kullanılır [180-184].

HRSÜ, gerçek ve sözde rastgele sayı üreteçlerinin birlikte kullanılmasıyla elde edilirler [183].

Rastgele sayı üreteçlerin temel blok şeması Şekil 7.1.'de verilmiştir. Buna göre RSÜ'leri, bir rastgelellik kaynağı, örnekleyici ve son işlem biriminden oluşur. Rastgelelik kaynağı yukarıda belirtilen herhangi bir gerçek veya sözde bir kaynak

olabilir. Örnekleme biriminde ise rastgelelik kaynağından gelen sinyallerden sayısal bilgiler elde edilir. Son işlem birimi, çeşitli farklı yapılar ile üretilen bilgilerin rastgelelik derecesini arttırmak için kullanılır. Son işlem (post-process) biriminde Von Neumann doğrultucusu yöntemi, XOR yöntemi, karıştırma algoritmaları (bir özet fonksiyonundan geçirme) işlemlerinden biri uygulanabilir [182-185].



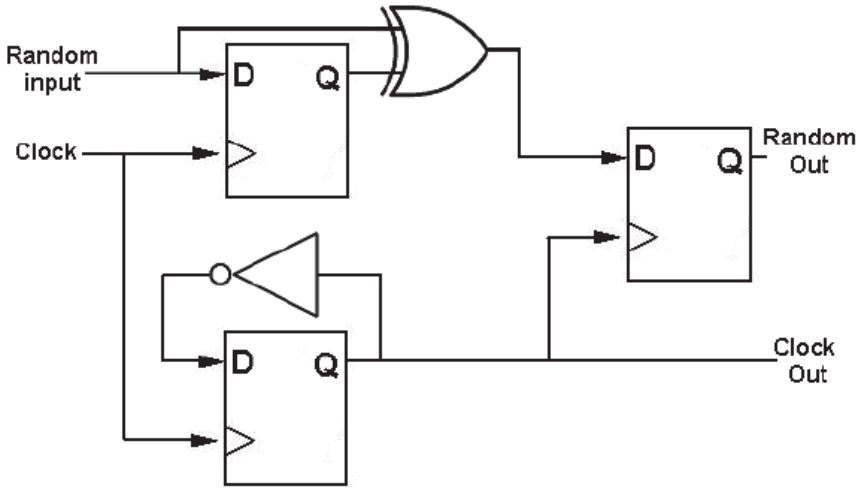
Şekil 7.1. Rastgele sayı üreteci temel blok şeması [182]

Von Neumann son işlem yönteminde, üretilen bitlerdeki "00" ve "11" ikili bit çiftleri silinir, "10" olan bit çiftlerine "1" değeri, "01" olan bit çiftlerine "0" değeri verilir. Tablo 7.1.'de Von Neumann son işlem yöntemi özetlenmiştir. Bu yöntemde gelen aynı çift bitler atıldığı ve farklı değerli bit çifti tek bite dönüştürüldüğü için bit üretim hızı düşmektedir [182-185].

Tablo 7.1. Von Neumann son işlem yöntemi çıkış değerleri [183]

Üretilen Bit Çiftleri	Von Neumann İşlemi Çıkışı
00	Çıkış yok
01	0
10	1
11	Çıkış yok

EXOR son işlem yönteminde, bit dizisi istenen n bitlik bloklara ayrılır. Ayrılan bloklar kendi içinde EXOR (özel veya) işlemine tabi tutularak tek bir bit rastgele sayı üretilir. Bu yöntemde çıkış bit hızı *1/n* oranında düşer. Şekil 7.2'de n=2 bitlik EXOR son işlem uygulaması verilmiştir [182, 183, 185].



Şekil 7.2. EXOR son işlem uygulaması (n=2 bit için) [185]

Karıştırma algoritmaları son işlem yönteminde, bit dizisi doğrusal ve/veya doğrusal olmayan fonksiyon veya algoritma üzerinde işlenir. Bu yöntemde hiçbir veri kaybı oluşmaz fakat bit dizilerinin işlenmesi için zaman ve bellek ihtiyacı gereklidir [182].

7.2. Rastgelelik Testleri

Rastgele sayı üreteçlerinden üretilen bitlerin (0 veya 1) rastgelelilik derecesini ölçen çeşitli testler vardır. Rastgelelik testleri için en çok kullanılan ve geçerli olan iki tane test FIPS (Federal Information Processing Standards – Federal Bilgi İşleme Standardı) tarafından belirlenen FIPS 140-1 ve NIST (National Institute of Standarts and Technology – Ulusal Standartlar ve Teknoloji Enstitüsü) tarafından belirlenen NIST 800-22 testleridir [180, 182, 184, 185].

7.2.1. FIBS 140-1 testi

FIPS 140-1 testi rastgele sayı üretecinden elde edilen bilgilerin rastgelelik kalitesini belirlemede kullanılan bir testtir. Veri blok uzunluğu küçük olan veriler için kullanılır. Bu test için rastgele sayı üreteçlerinden oluşturulan 20.000 adet bitlik bir X dizisi *monobi*t, *poker*, *runs* ve *long runs* testi olmak üzere dört adet teste tabi tutulur. Bu testlerden herhangi birinden geçilemediği takdirde rastgele sayı üreteci başarısız kabul edilir [180, 184-188].

7.2.1.1. Monobit testi

Bu testten başarılı olmak için üretilen 20.000 bitlik *X* dizisindeki "1"'lerin sayısının 9654<*X*<10346 arasında olması gereklidir [180, 184-188].

7.2.1.2. Poker testi

Bu test de 20.000 adet bitlik veri 4 bitlik olarak 5.000 segmente (parçaya) bölünür. Bu parçalarda 4 bitin 16 olası durumu sayılır ve kaydedilir. Denklem 7.1'de verilen formül ile *X* değeri hesaplanır. Denklemde *f(i)* ifadesi, 16 olası durum değerindeki *(i)* sayılan sayı değerini ifade eder. Denklem 7.1 ile hesaplanan X değeri $1.03 < X \leq 57.4$ aralığında ise test başarılı sayılır [180, 184-188].

$$X = (16/5000)\left(\sum_{i=0}^{15}[f(i)]^2\right) - 5000 \tag{7.1}$$

7.2.1.3. Runs testi

Bu test ile üretilen bit dizisinde art arda gelen "1" veya "0" bitlerinden oluşan blokların sayısının Tablo 7.2.'de verilen değer aralıklarında olup olmadığı test edilir. 6 bitten uzun bloklar 6 bitlik olarak kabul edilir. Bu testin amacı "0" ve "1" bitleri arasındaki salınımın çok hızlı veya çok yavaş olup olmadığının belirlenmesidir [180, 184-188].

Tablo 7.2. Runs testi kriterleri [180, 185, 188]

Blok Uzunluğu	Blok Sayısı Aralığı
1	2267-2733
2	1079-1421
3	502-748
4	223-402
5	90-223
6+	90-223

7.2.1.4. Long runs testi

Run testi ile aynıdır. Fakat bu Long Runs testinin başarılı olabilmesi için üretilen 20000 bitlik bir dizi içindeki "1" veya "0" lardan oluşan tüm blokların sayısı 34'ten küçük olmalıdır [180, 184-188].

7.2.2. NIST 800-22 testi

NIST 800-22 testi üretilen rastgele sayıların rastgelelik derecesini ölçmek için geliştirilmiş istatiksel bir testtir. NIST 800-22 testi en az 1000000 adet veri ile gerçekleştirilir. NIST 800-22 testi kendi içinde 15 ayrı test içerir. Bu 15 testin içerikleri aşağıda açıklanmıştır. NIST 800-22 testi her alt testte bir *p* değeri (probability) hesaplar. Testlerin başarılı kabul edilmesi için bu *p* değerinin 0,01'den büyük olması gerekir [180, 183-185, 189].

7.2.2.1. Frekans testi (Frequency monobit test)

Bu test üretilen bit dizisi içindeki "1" ve "0" bitlerinin sayısının oranını inceler. Yani bit dizisindeki "1" ve "0" bit sayılarının yaklaşık olarak aynı olup olmadığının incelendiği bir testtir [180, 184, 189].

7.2.2.2. Blok frekans testi (Frequency test within a block)

Bu test üretilen bit dizisi içindeki *m* bitlik bir blok içindeki "1" ve "0" bitlerinin sayısının oranını inceler [180, 184, 189].

7.2.2.3. Yinelemeler testi (Runs test)

Bu test üretilen bit dizisi içindeki art arda gelen "1" ve "0" bloklarının sayısını inceleyen bir testtir [180, 184, 189].

7.2.2.4. Blok içinde en uzun bir yinelemesi testi (Tests for the longest-run-of-ones in a block test)

Bu test üretilen bit dizisi içindeki art arda gelen en uzun "1" sayısını inceleyen bir testtir [180, 184, 189].

7.2.2.5. İkili matris rankı testi (Binary matrix rank test)

Bu test ile sabit uzunluklu bit blokları kullanılarak, her biri bir satırı belirtecek şekilde, bir matris oluşturulur ve matrisin rankı hesaplanarak bloklar arasındaki lineer bağımlılık incelenir [180, 184, 189].

7.2.2.6. Ayrık Fourier dönüşümü testi (Discrete Fourier transform test)

Bu test ile üretilen bit dizisinin ayrık Fourier dönüşümü alınarak dizinin periyodikliği incelenir [180, 184, 189].

7.2.2.7. Örtüşmeyen şablon eşleştirme testi (Non-overlapping template matching test)

Bu test üretilen bit dizisinin içindeki m bitlik bir bloğun dizi içindeki tekrarını inceleyen bir testtir [180, 184, 189].

7.2.2.8. Örtüşen şablon eşleştirme testi (Overlapping template matching test)

Örtüşmeyen şablon eşleştirme testi ile aynıdır. Farkı, eğer bir örüntü (dizi tekrarı) bulunursa pencere bulunan örüntüden sonraki ilk bit yerine sadece o an bulunan pozisyondan bir sonraki bite yeniden konumlanır ve taramaya devam edilir [180, 184, 189].

7.2.2.9. Maurer'in "evrensel istatistik" testi (Maurer's "universal statistical" test)

Bu test üretilen bit dizisinin veri kaybı olmadan ne kadar sıkıştırılabileceğini inceler [180, 184, 189].

7.2.2.10. Doğrusal karmaşıklık testi (Linear complexity test)

Bu test üretilen bit dizisinin karmaşıklığını inceleyen bir testtir [180, 184, 189].

7.2.2.11. Seri testi (Serial test)

Bu test üretilen bit dizisi içindeki tekrar eden m bitlik 2^m tane bloğun tekrar sayısının dağılımını inceleyen bir testtir [180, 184, 189].

7.2.2.12. Yaklaşık entropi testi (Approximate entropy test)

Bu test üretilen bit dizisi içindeki iki ardışık bloğun (m ve m+1) frekansını (entropisini) inceleyen bir testtir [180, 184, 189].

7.2.2.13. Kümülatif toplamlar testi (Cumulative sums test)

Bu test üretilen bit dizisini ardışık bloklara ayırır ve bu bloklar içindeki "1" ve "0" bitlerinin sayısının oranını (dengesini) belirleyip, bloklar arasındaki bu oranların dengesizliği inceleyen bir testtir [180, 184, 189].

7.2.2.14. Rastgele gezinimler testi (Random excursions test)

Bu test üretilen bit dizisini ardışık bloklara ayırır ve bu bloklar içindeki "1" ve "0" bitlerinin sayısının oranını (dengesini) belirleyip, bloklar arasındaki bu oranların dengesinin dağılımını inceleyen bir testtir [180, 184, 189].

7.2.2.15. Rastgele gezinimler değişken testi (Random excursions variant test)

Bu test ile üretilen bit dizisi ardışık bloklara ayrılır ve bu bloklar içindeki "1" ve "0" bitlerinin sayısının oranı (dengesi) belirlenip ortalama değerden sapma miktarı belirlenir [180, 184, 189].

7.3. Yeni Kaotik Sistem ile Matlab-Simulink Ortamında GRSÜ Tasarımı

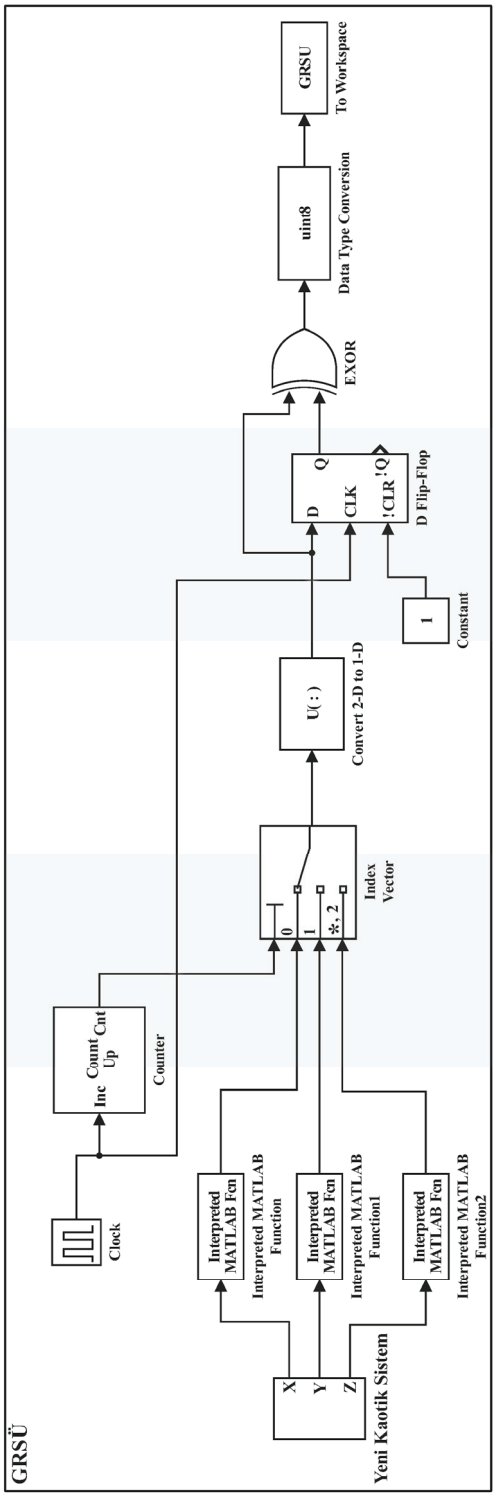
Bu kısımda tez çalışması kapsamında yeni kaotik sistem ile Matlab-Simulink ortamında GRSÜ tasarımı yapılmıştır. Tasarlanan GRSÜ çıkışından elde edilen veriler FIPS 140-1 ile NIST 800-22 testlerine tabi tutulmuştur.

Şekil 7.3.'te yeni kaotik sistem kullanılarak elde edilen GRSÜ tasarımının MATLAB-Simulink blok diyagramı verilmiştir. GRSÜ tasarımında rastgelelik kaynağı olarak tez çalışmasında elde edilen yeni kaotik sistemin x, y ve z durum değişkenleri değeri kullanılmıştır.

Yeni kaotik sistemden elde edilen x, y ve z durum değişkeni değerleri ilk önce 64 bitlik kayan noktalı (floating point) sayı değerine çevrilmiştir. Ardından 64 bitlik kayan noktalı sayı değerinin sadece 32. biti alınmıştır [190]. Bu iki işlemi gerçekleştirmek için Matlab programında bir fonksiyon kullanılmıştır. Bu işlemi sağlamak için GRSÜ Matlab-Simulink bloğunda "*Index Vector*" , "*Clock*" ve iki değerine kadar sayan "*Counter*" elemanları kullanılmıştır. "*Counter*" elemanı girişindeki "*Clock*" sinyali ile çıkışına sırayla "00, 01, 10" değerlerini göndermektedir. "*Index Vector*" yapısı ise "*Counter*" elemanından gelen bu bilgiler ışığında çıkışına sırayla girişindeki 64 bitlik x, y, z değerlerinin 32. bit değerlerini göndermektedir. "*Index Vector*" elemanının çıkışındaki satır matris değerleri "*Convert 2-D to 1-D*" elemanı ile sütun matrisine çevrilmektedir.

GRSÜ'nin son işlem biriminde (post process) EXOR son işlem yöntemi kullanılmıştır. EXOR son işleminde gelen x, y, z durum değişkenlerinin 64 bitlik kayan noktalı sayı değerlerinin 32. bitleri her saat palsinde sırayla (x ve y, sonra z ve x, sonra y ve z gibi) EXOR (Exclusive OR - özel veya) işlemine tabi tutulmuştur. İşlem sonucu elde edilen bitler GRSÜ biriminin çıkışını oluşturmaktadır [190]. Bu işlemi sağlamak için Matlab-Simulink blok şemasında "*D Flip-Flop* (D/FF)" elemanı kullanılmıştır. D/FF ile şimdiki değer ile bir önceki değerin korunması sağlanmıştır. D/FF elemanından gelen şimdiki ve bir önceki değer *EXOR* elemanı ile son işleme tabi tutulmuştur. *EXOR* elemanı çıkışındaki mantıksal bilgileri sayısal işlemde

kullanabilmek için "*Data Type Conversion*" elemanı ile işaretsiz tam sayı tipine dönüştürülmüş ve bu şekilde GRSÜ çıkışı elde edilmiştir.



Şekil 7.3. Yeni kaotik sistem kullanılarak tasarlanan GRSÜ tasarımının Matlab-Simulink blok şeması

7.3.1. GRSÜ Matlab tasarımı sonuçlarının rastgelelik testleri sonuçları

MATLAB-Simulink ortamında tasarlanan GRSÜ çıkışından alınan veriler FIBS 140-1 ve NIST 800-22 testlerine tabi tutulmuştur. Test sonuçları aşağıda verilmiştir.

7.3.1.1. GRSÜ Matlab tasarımının FIPS 140-1 testleri sonuçları

Matlab-Simulink ortamında tasarlanan GRSÜ çıkışından alınan 20000 adet veri FIPS 140-1 testlerine tabi tutulmuş ve tüm testlerden başarılı sonuç alınmıştır. Tablo 7.3.'te bu testler, şartları, test sonuçları ve başarı durumları verilmiştir.

Tablo 7.3. GRSÜ Matlab tasarımının FIPS 140-1 testleri sonuçları

Test	Gerekli Şart	Elde Edilen Değer	Sonuç
Monobit	9654 < X < 10346	9901	Başarılı
Poker	1,03 < X <57,4	51	Başarılı
Runs	Ard arda gelen 1 ve 0 blok uzunluğu 1 bit olanlar. Aralık: 2267-2733	2448	Başarılı
	Ard arda gelen 1 ve 0 blok uzunluğu 2 bit olanlar. Aralık: 1079-1421	1318	Başarılı
	Ard arda gelen 1 ve 0 blok uzunluğu 3 bit olanlar. Aralık: 502-748	601	Başarılı
	Ard arda gelen 1 ve 0 blok uzunluğu 4 bit olanlar. Aralık: 223-402	330	Başarılı
	Ard arda gelen 1 ve 0 blok uzunluğu 5 bit olanlar. Aralık: 90-223	181	Başarılı
	Ard arda gelen 1 ve 0 blok uzunluğu 6 ve daha fazla bit olanlar. Aralık: 90-223	142	Başarılı
Long Runs	"1" ve "0" dan oluşan tüm blokların sayısı 34'ten küçük olmalıdır.	34'ten büyük blok bulunamamıştır.	Başarılı

7.3.1.2. GRSÜ Matlab tasarımının NIST 800-22 testleri sonuçları

Matlab-Simulink ortamında tasarlanan GRSÜ çıkışından alınan 1000000 adet veri NIST 800-22 testlerine tabi tutulmuş ve tüm testlerden başarılı sonuç alınmıştır. NIST 800-22 test sonuçları Tablo 7.4.'te verilmiştir. Testlerden başarılı olmak için *p*

değerinin 0,01'den büyük olması gereklidir. Random-Excursions testi ve Random-Excursions Variant testleri için gerekli döngü sayısı oluşmadığından bu testler gerçekleştirilememiştir.

Tablo 7.4. GRSÜ Matlab tasarımının NIST 800-22 testleri sonuçları

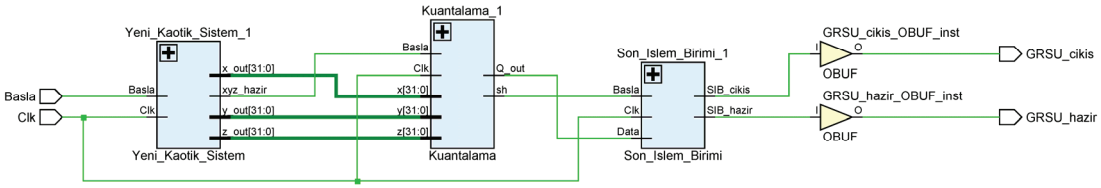
NIST 800-22 Testleri	*p* Değeri	Sonuç
Frekans testi	0,12561	Başarılı
Blok frekans testi (M=128)	0,96166	Başarılı
Yinelemeler testi	0,51418	Başarılı
Blok içinde en uzun bir yinelemesi testi	0,93980	Başarılı
İkili matris rankı testi	0,57019	Başarılı
Ayrık Fourier testi	0,28711	Başarılı
Örtüşmeyen şablon eşleştirme testi (m=7)	0,79326	Başarılı
Örtüşen şablon eşleştirme testi (m=9)	0,19318	Başarılı
Maurer'in "evrensel istatistik" testi	0,43276	Başarılı
Doğrusal karmaşıklık testi (M=500)	0,64950	Başarılı
Seri testi-1 (m=16)	0,38020	Başarılı
Seri testi-2 (m=16)	0,25912	Başarılı
Yaklaşık entropi testi (m=10)	0,09582	Başarılı
Kümülatif toplamlar testi (İleri)	0,18826	Başarılı
Rastgele gezinimler testi	Gerekli döngü sayısı oluşmamıştır.	-
Rastgele gezinimler değişken testi	Gerekli döngü sayısı oluşmamıştır.	-

7.4. Yeni Kaotik Sistem ile FPGA Tabanlı GRSÜ Tasarımı

Bu kısımda Bölüm 7.3.'te Matlab-Simulink ortamında yeni kaotik sistem kullanılarak tasarlanan GRSÜ yapısı VHDL donanım tanımlama dili kullanılarak FPGA tabanlı olarak tasarlanmış ve elde edilen veriler FIPS 140-1 ile NIST 800-22 testlerine tabi tutulmuştur. FPGA tasarımında Xilinx firmasının Artix-7 ailesinden xc7a100tcsg324-1 modeli kullanılmıştır.

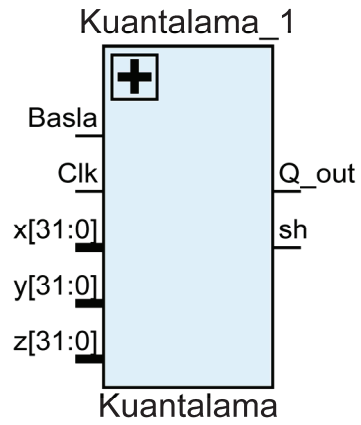
Şekil 7.4.'te yeni kaotik sistem kullanılarak FPGA tabanlı tasarlanan GRSÜ'nin en üst düzey yapısı verilmiştir. Şekil 7.4.'te de görüldüğü gibi tasarımda yeni kaotik sistem birimi, kuantalama birimi ve son işlem birimi bulunmaktadır. Yeni kaotik sistem biriminin FPGA tasarımı Bölüm 5'te ayrıntısı ile verilen tasarımın aynısıdır.

FPGA tabanlı GRSÜ, *Basla* ve *Clk* olmak üzere iki giriş pinleri ile *GRSU_cikis* ve *GRSU_hazir* çıkış pinlerine sahiptir. *Clk* pini sistemin çalışma saat darbesi (Clock Pulse) girişidir. *Basla* pini Lojik-1 olduğunda GRSÜ birimi sayı üretimine başlar. *GRSU_hazir* pini üretilen rastgele bilginin *GRSU_cikis* pininden alınmaya başlanabileceğini belirtir. Üretilen rastgele bilgiler *GRSU_cikis* pininden alınır.

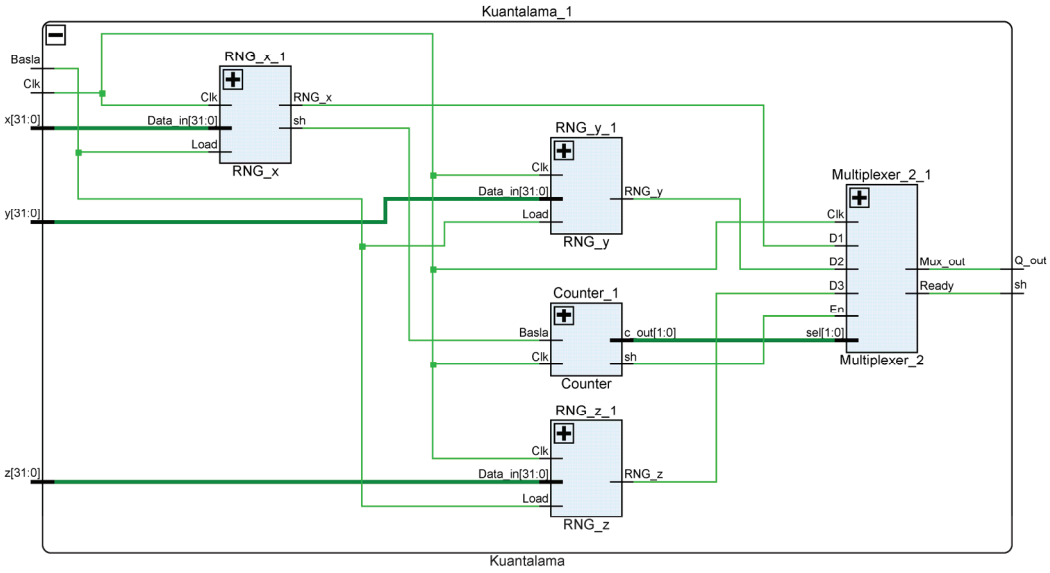


Şekil 7.4. Yeni kaotik sistem kullanılarak FPGA tabanlı tasarlanan GRSÜ

Kuantalama biriminde, yeni kaotik sistemden elde edilen 32 bitlik kayan noktalı formattaki *x*, *y* ve *z* durum değişkenleri değerlerinin LSB bit değerleri alınır ve sırayla *kuantalama* biriminin çıkışına aktarılır. Şekil 7.5.'te *kuantalama* birimi pin diyagramı ve Şekil 7.6.'da ise *kuantalama* biriminin iç yapısı verilmiştir. *Kuantalama* birimi *Basla* pininden Lojik-1 gelmesi ile *x*, *y* ve *z* pinlerinden gelen 32 bitlik verinin LSB değerlerini *RNG_x*, *RNG_y*, *RNG_z* birimleri vasıtasıyla alır. Alınan bu LSB bilgileri iki değerine kadar sayan *Counter* sayıcı birimi ve *Multiplexer* birimi ile sırayla *Q_out* pininden kuantalama çıkışına aktarılır [190]. *sh* pini Lojik-1 olduğunda *kuantalama* biriminin çıkışına veri göndermeye hazır olduğunu belirtir. *Clk* pini sistemin çalışma saat darbesi (Clock Pulse) girişidir.

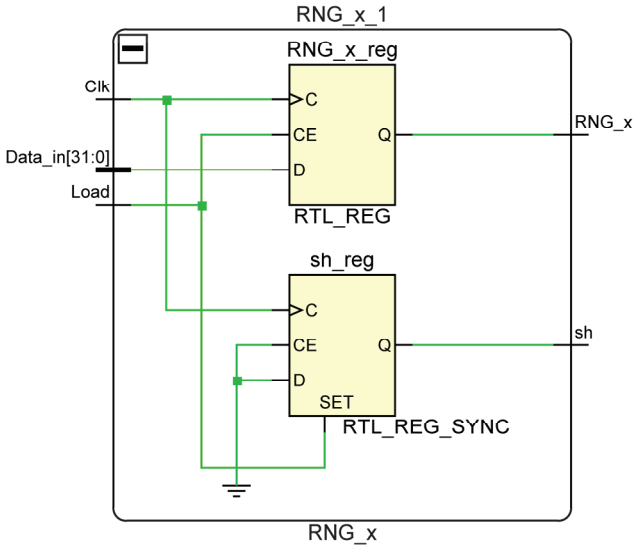


Şekil 7.5. Kuantalama birimi pin diyagramı

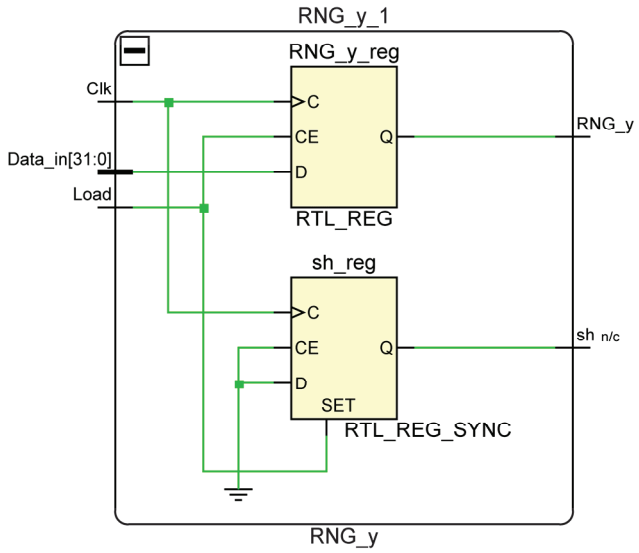


Şekil 7.6. Kuantalama birimi iç yapısı

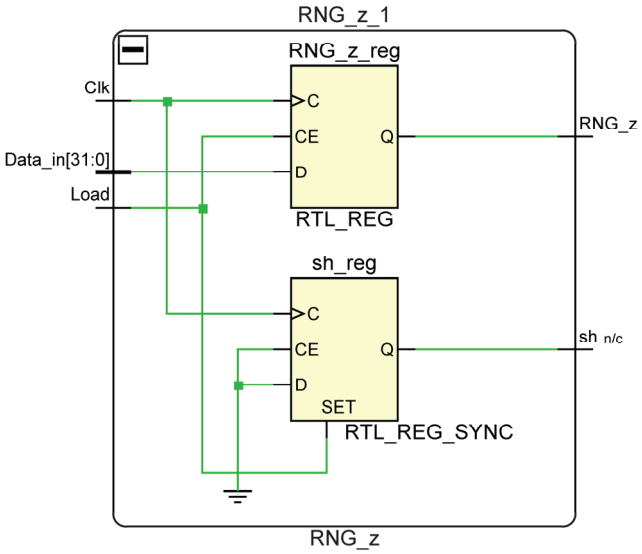
Yeni_Kaotik_Sistem biriminden gelen 32 bitlik *x_out*, *y_ou*t ve *z_out* kaotik sinyal verilerinin LSB değerleri sırayla tasarlanan *RNG_x*, *RNG_y* ve *RNG_z* birimleri vasıtasıyla alınmaktadır. *RNG_x*, *RNG_y* ve *RNG_z* birimlerinin iç yapıları sırayla Şekil 7.7., Şekil 7.8. ve Şekil 7.9.'da verilmiştir.



Şekil 7.7. RNG_x birimi iç yapısı

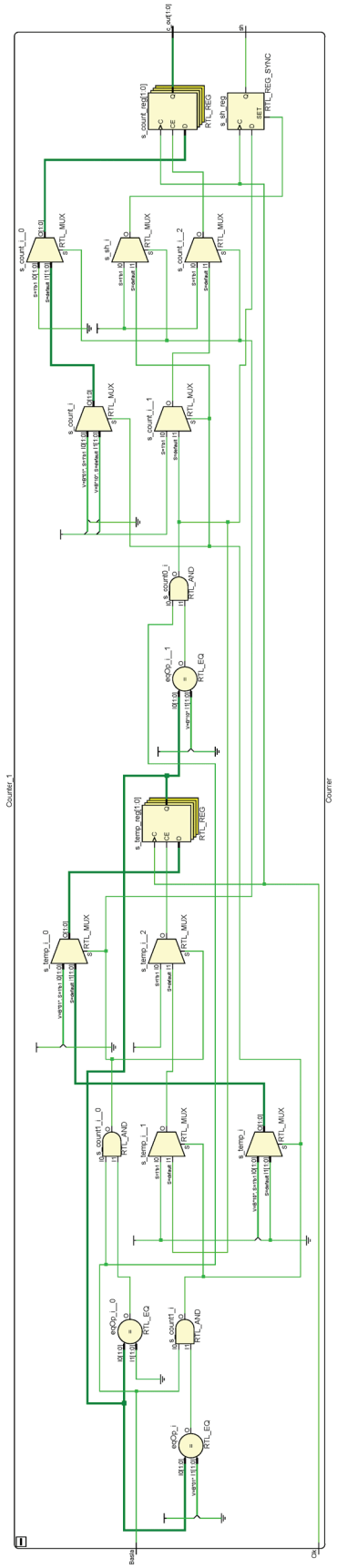


Şekil 7.8. RNG_y birimi iç yapısı

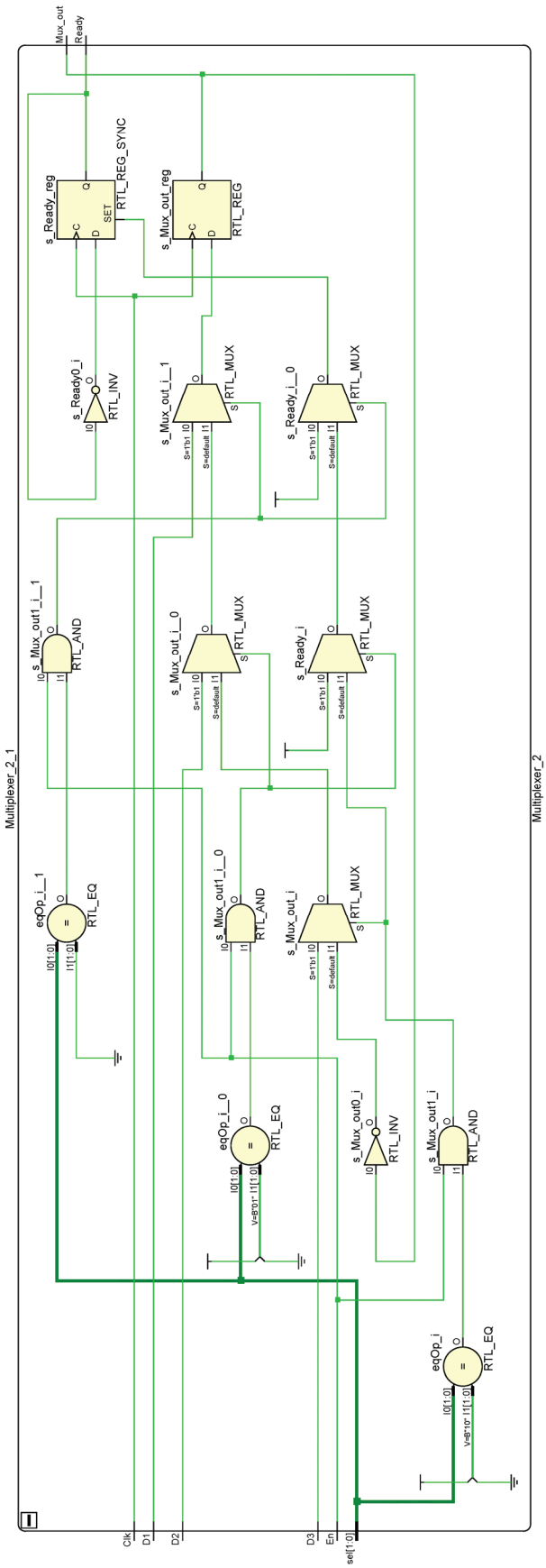


Şekil 7.9. RNG_z birimi iç yapısı

RNG_x, *RNG_y* ve *RNG_z* birimlerinden gelen bilgilerin sırayla çıkışa aktarılması için iki değerine kadar sayan *Counter* birimi ile *Counter* birimden gelen değere göre sırayla *RNG_x*, *RNG_y* ve *RNG_z* birimi çıkış değerlerini çıkışa aktaran *Multiplexer* birimlerinin iç yapısı sırayla Şekil 7.10. ve Şekil 7.11.'de verilmiştir.

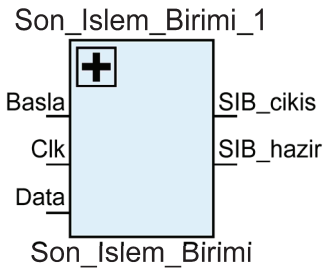


Şekil 7.10. Counter birimi iç yapısı

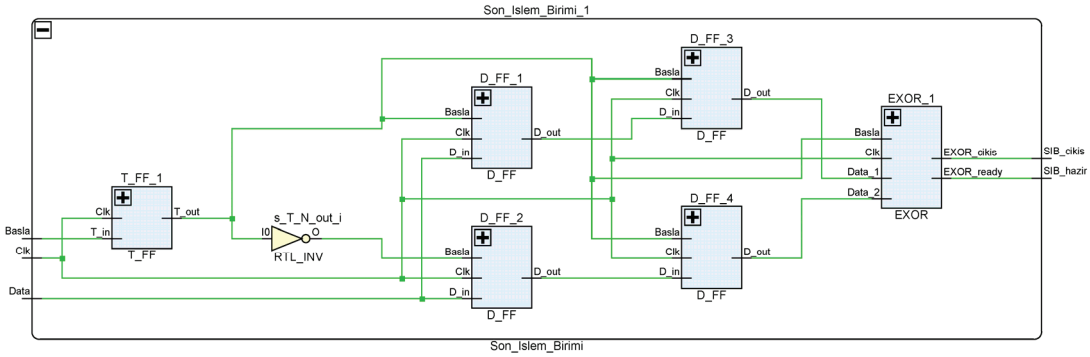


Şekil 7.11. Multiplexer birimi iç yapısı

Kuantalama biriminden gelen veriler *Son İşlem* birimine aktarılır. Şekil 7.12.'de *Son İşlem* birimi pin diyagramı ve Şekil 7.13.'te ise *Son İşlem* biriminin iç yapısı verilmiştir. *Son İşlem* biriminde, *Kuantalama* biriminden gelen verinin hazır olduğunu bildiren *Basla* pini ve *Kuantalama* biriminden gelen verilerin girdiği *Data* giriş pinleri bulunur. *Clk* pini sistemin çalışma saat darbesi girişidir. *Son İşlem* birimi çıkışında verinin çıktığı *SIB_cikis* ve çıkış verisinin hazır olduğunu belirten *SIB_hazir* pinleri bulunur. *Son İşlem* biriminde *EXOR* son işlem yapısı kullanılmıştır. *EXOR* son işleminde *Kuantalama* biriminden gelen *x*, *y*, *z* durum değişkenlerinin 32 bitlik kayan noktalı sayı değerlerinin LSB bitleri her saat palsinde sırayla (*x* ve *y*, sonra *z* ve *x*, sonra *y* ve *z* gibi) *EXOR* (Exclusive OR - özel veya) işlemine tabi tutulmaktadır. Bu işlemi sağlamak için tasarımda D Flip-Flop (D/FF) ve T Flip-Flop (T/FF) elemanları kullanılmıştır. D/FF yapısı ile şimdiki değer ile bir önceki değerin korunması sağlanmıştır. T/FF yapısı ile de *EXOR* yapısına sırayla *x* ve *y*, sonra *z* ve *x*, sonra *y* ve *z* gibi değerlerin gönderilmesi sağlanmıştır [190]. Gelen iki değer *EXOR* yapısı ile EXOR işlemine tabi tutularak *SIB_cikis* pininden çıkışa aktarılır.



Şekil 7.12. Son İşlem birimi pin diyagramı



Şekil 7.13. Son İşlem birimi iç yapısı

FPGA tabanlı GRSÜ, Xilinx Vivado Design Suite v2015.4 programında tasarlanmış ve sentezleme (Synthesis) ile gerçekleme (Implementation) işleminlerine tabi tutulmuştur. Tasarlanan FPGA tabanlı GRSÜ birimi her yeni rastgele çıkış değerini 62 saat periyodu süresinde çıkışa aktarmaktadır.

Tasarımın benzetimi, Xilinx Vivado Design Suite 2015.4 programı simülatöründe yapılmıştır. GRSÜ çıkışından alınan veriler FIBS 140-1 ve NIST 800-22 testlerine tabi tutulmuştur. Sistemde maksimum çalışma frekansı 392,927 MHz ve minimum çalışma periyodu 2,545ns olarak elde edilmiştir. Sistemin maksimum çalışma frekansının 392,927 MHz olduğu düşünülürse ve her 62 saat periyodu süresinde yeni bir rastgele sayı üretildiği hesaba katıldığında tasarlanan FPGA tabanlı GRSÜ biriminin rastgele sayı üretme hızı 392,927 MHz / 62 ≈ 6,338 MHz olmaktadır.

Tablo 7.5.'te GRSÜ tasarımının Xilinx Artix-7 ailesi xc7a100tcsg324-1 modeli üzerinde gerçeklenen FPGA çip istatistikleri verilmiştir. Tablo 7.5.'te verilen kısaltmaların açılımları şu şekildedir: LUT (Look-up Table, Başvuru Tablosu), LUTRAM (Look-up Table RAM), FF (Flip-Flop), DSP (Digital Signal Processor, Sayısal Sinyal İşleyici), IO (Input/Output, Giriş/Çıkış pin sayısı).

Tablo 7.5. FPGA tabanlı GRSÜ tasarımının çip istatistikleri

Kaynaklar	Mevcut	Kullanılan	Kullanım Oranı (%)
LUT	63400	2570	4,05
LUTRAM	19000	50	0,26
FF	126800	2246	1,77
DSP	240	48	20
IO	210	4	1,9

7.4.1. FPGA tabanlı GRSÜ tasarımının rastgelelik testleri

Tasarlanan FPGA tabanlı GRSÜ çıkışından alınan veriler FIBS 140-1 ve NIST 800-22 testlerine tabi tutulmuştur. Test sonuçları aşağıda verilmiştir.

7.4.1.1. FPGA tabanlı GRSÜ tasarımının FIPS 140-1 testleri sonuçları

FPGA tabanlı tasarlanan GRSÜ çıkışından alınan 20000 adet veri FIPS 140-1 testlerine tabi tutulmuş ve tüm testlerden başarılı sonuç alınmıştır. Tablo 7.6.'da bu testler, şartları, test sonuçları ve başarı durumları verilmiştir.

Tablo 7.6. FPGA tabanlı GRSÜ tasarımının FIPS 140-1 testleri sonuçları

Test	Gerekli Şart	Elde Edilen Değer	Sonuç
Monobit	9654 < X < 10346	10143	Başarılı
Poker	1,03 < X <57,4	51	Başarılı
Runs	Art arda gelen 1 ve 0 blok uzunluğu 1 bit olanlar. Aralık: 2267-2733	2551	Başarılı
	Art arda gelen 1 ve 0 blok uzunluğu 2 bit olanlar. Aralık: 1079-1421	1209	Başarılı
	Art arda gelen 1 ve 0 blok uzunluğu 3 bit olanlar. Aralık: 502-748	652	Başarılı
	Art arda gelen 1 ve 0 blok uzunluğu 4 bit olanlar. Aralık: 223-402	305	Başarılı
	Art arda gelen 1 ve 0 blok uzunluğu 5 bit olanlar. Aralık: 90-223	141	Başarılı
	Art arda gelen 1 ve 0 blok uzunluğu 6 ve daha fazla bit olanlar. Aralık: 90-223	146	Başarılı
Long Runs	"1" ve "0" dan oluşan tüm blokların sayısı 34'ten küçük olmalıdır.	34'ten büyük blok bulunamamıştır.	Başarılı

7.4.1.2. FPGA tabanlı GRSÜ tasarımının NIST 800-22 testleri sonuçları

FPGA tabanlı tasarlanan GRSÜ çıkışından alınan 1000000 adet veri NIST 800-22 testlerine tabi tutulmuş ve tüm testlerden başarılı sonuç alınmıştır. NIST 800-22 test sonuçları Tablo 7.7.'de verilmiştir. Testlerden başarılı olmak için p değerinin 0,01'den büyük olması gereklidir. Rastgele gezinimler testi ve Rastgele gezinimler değişken testleri için gerekli döngü sayısı oluşmadığından bu testler gerçekleştirilememiştir.

Tablo 7.7. FPGA tabanlı GRSÜ tasarımının NIST 800-22 testleri sonuçları

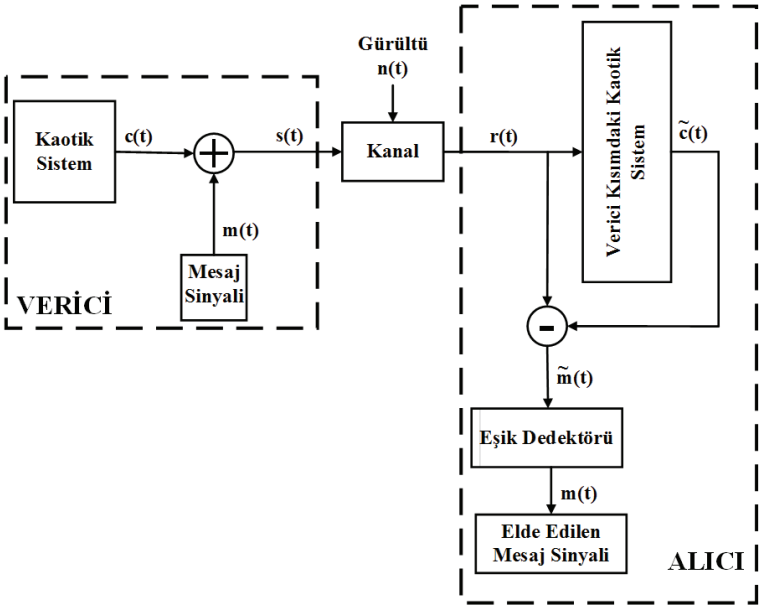
NIST 800-22 Testleri	*p* Değeri	Sonuç
Frekans testi	0,63407	Başarılı
Blok frekans testi (M=128)	0,05181	Başarılı
Yinelemeler testi	0,12596	Başarılı
Blok içinde en uzun bir yinelemesi testi	0,63903	Başarılı
İkili matris rankı testi	0,06250	Başarılı
Ayrık Fourier testi	0,45730	Başarılı
Örtüşmeyen şablon eşleştirme testi (m=7)	0,02530	Başarılı
Örtüşen şablon eşleştirme testi (m=9)	0,21558	Başarılı
Maurer'in "evrensel istatistik" testi	0,28254	Başarılı
Doğrusal karmaşıklık testi (M=500)	0,74121	Başarılı
Seri testi-1 (m=16)	0,65372	Başarılı
Seri testi-2 (m=16)	0,83948	Başarılı
Yaklaşık entropi testi (m=10)	0,58354	Başarılı
Kümülatif toplamlar testi (İleri)	0,81028	Başarılı
Rastgele gezinimler testi	Gerekli döngü sayısı oluşmamıştır.	-
Rastgele gezinimler değişken testi	Gerekli döngü sayısı oluşmamıştır.	-

BÖLÜM 8. ŞİFRELİ KAOTİK HABERLEŞME SİSTEMİ TASARIMININ BENZETİM ÇALIŞMASI

Bu bölümde elde edilen yeni kaotik sistem ve bu sistem kullanılarak tasarlanan GRSÜ birimi ile kaotik maskeleme modülasyon yöntemi kullanılarak şifreli kaotik haberleşme sistemi benzetim çalışması gerçekleştirilmiştir. Benzetim çalışması Matlab-Simulink ortamında yapılmıştır. Tasarlanan şifreli kaotik haberleşme sistemi, rastgele ikilik bilgiler kullanılarak test edilmiştir.

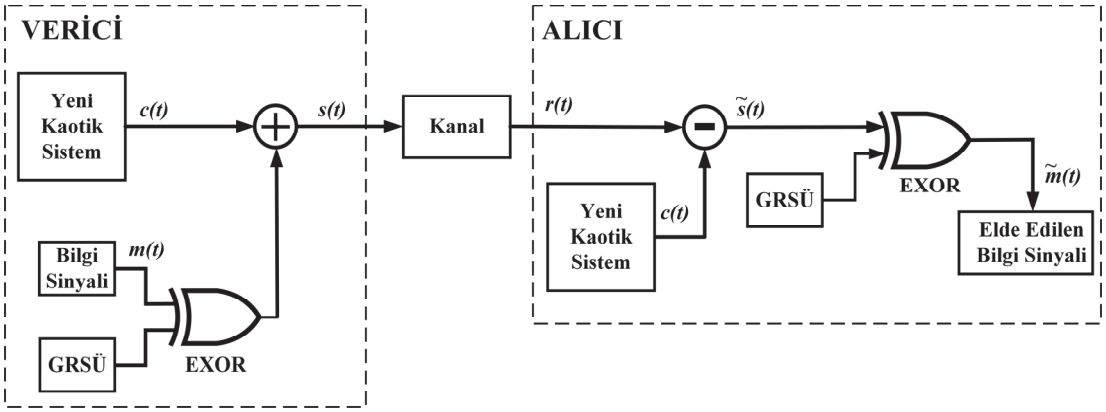
8.1. Şifreli Kaotik Haberleşme Sistemi Tasarımı

Şifreli kaotik haberleşme sistemi tasarımında 3. Bölümde ayrıntısı verilen kaotik maskeleme (KM) (Chaotic masking – CM) modülasyon yöntemi kullanılmıştır. KM yöntemi hem analog hem de sayısal verilerin gönderilmesinde kullanılan bir kaos tabanlı haberleşme yöntemidir. Şekil 8.1.'de KM yönteminin yapısı verilmiştir. Bu yöntemde, verici birimde kaotik sinyale c*(t),* gönderilecek bilgi sinyali *m(t)* eklenerek modüleli sinyal *s(t)* elde edilir. İletim kanalı (ortamı) üzerinden alıcı birime gürültülü modülasyonlu sinyal *r(t)* gelir. Alıcı birimde ilk önce verici kısımdaki kaotik sinyal yeniden aynen elde edilir. Elde edilen kaotik sistem sinyali $\tilde{c}(t)$, gelen gürültülü sinyalden çıkartılarak gürültülü bilgi sinyali $\tilde{m}(t)$ elde edilir. Gürültülü sinyal, eşik dedektöründen geçirilerek gürültüsüz şekilde bilgi sinyali $m(t)$ elde edilir [75-78, 135, 154, 155].



Şekil 8.1. Kaotik maskeleme haberleşme sistemlerinin temel yapısı [155]

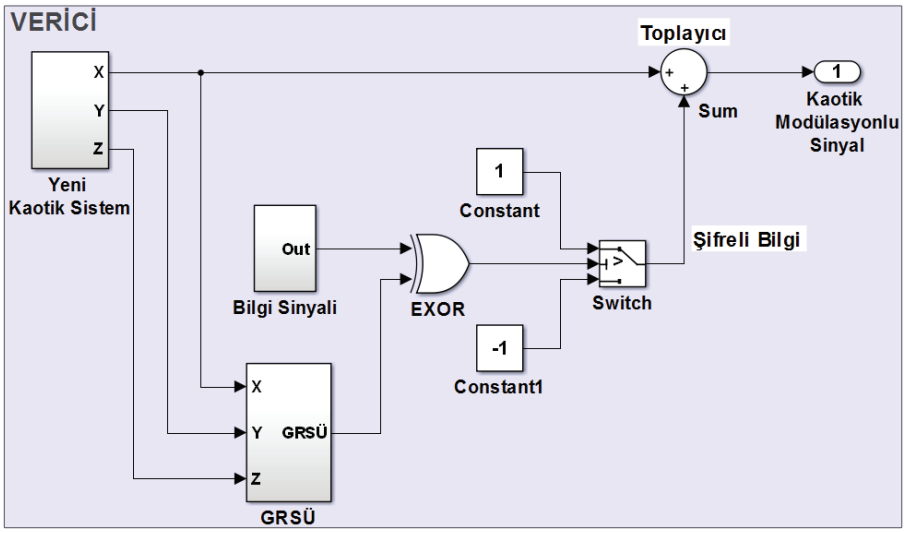
Tasarlanan şifreli kaotik maskeleme modülasyon yöntemli kaotik haberleşme sisteminin blok şeması Şekil 8.2.'de verilmiştir.



Şekil 8.2. Tasarlanan şifreli kaotik maskeleme modülasyon yöntemli kaotik haberleşme sistemi blok şeması

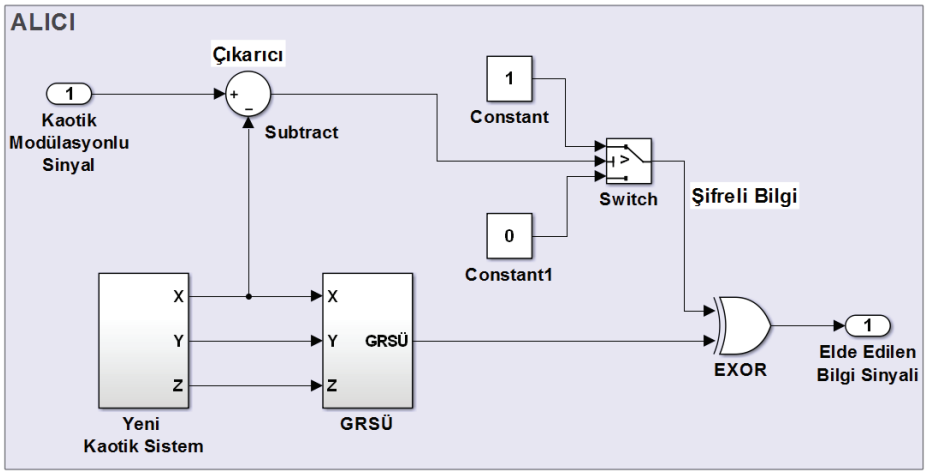
Tasarlanan kaotik maskeleme modülasyonlu kaotik haberleşme sistemi *verici* biriminin Matlab-Simulink blok şeması Şekil 8.3.'te verilmiştir. Verici biriminde şifreleme birimi olarak, elde edilen yeni kaotik sistem kullanılarak tasarlanan GRSÜ birimi (7. Bölümde tasarım çalışmaları verilmişti) kullanılmıştır. 7. Bölümde açıklandığı gibi GRSÜ biriminde: kaotik sistemin x, y ve z çıkışlarından elde edilen 32 bitlik kayan noktalı sayı değerlerinin en düşük değerlikli biti sırayla iki durum değeri (*x-y*, *y-z*, *z-x*) için alınır ve son işlem olarak da bu iki değer EXOR işlemine

tabi tutularak rastgele sayılar elde edilir. GRSÜ biriminden elde edilen rastgele "1, 0" bilgileri ile haberleşme sisteminin verici biriminden gönderilecek bilgi *EXOR* işleminden geçirilerek bilginin şifrelenmesi sağlanmıştır. Gönderilecek bilgide bulunan "0" sayısal değerinin kaotik maskeleme modülasyon biriminde bir etkisi yoktur. İçinde çarpma işlemi bulunan modülasyon tekniklerinde de "0" sayısal bilgisi sinyal değerini tümüyle sıfırladığından güvenlik sorunu oluşturabilmektedir. Bu nedenle verici biriminde şifrelenmiş "1, 0" yapısındaki sayısal bilgi "1, -1" yapısına çevrilmiştir. Bu şekilde elde edilen şifrelenmiş bilgi ile yeni kaotik sistem çıkışından gelen kaotik sinyal toplanarak modülasyonlu sinyal elde edilir.



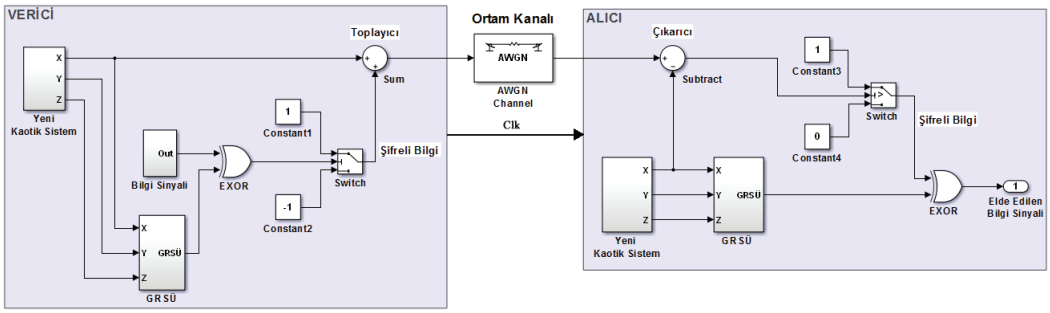
Şekil 8.3. Tasarlanan kaotik maskeleme modülasyonlu kaotik haberleşme sistemi verici biriminin Matlab-Simulink blok şeması

Tasarlanan kaotik maskeleme modülasyonlu kaotik haberleşme sistemi *alıcı* biriminin Matlab-Simulink blok şeması Şekil 8.4.'te verilmiştir. Verici biriminde oluşturulan kaotik sistem ve GRSÜ birimi alıcı birimde tekrar oluşturulmuştur. Verici birimden gelen gürültülü modülasyonlu sinyalden kaotik sistemden elde edilen sinyal çıkartılarak verici birimden gönderilen şifreli bilgi elde edilir. "1, -1" yapısındaki şifreli bilgi alıcı biriminde tekrar "1, 0" yapısına çevrilmektedir. GRSÜ biriminden çıkan bilgi ile elde edilen şifreli bilgi *EXOR* işlemine tabi tutularak şifresi çözülmüş bilgi sinyali elde edilir.



Şekil 8.4. Tasarlanan kaotik maskeleme modülasyonlu kaotik haberleşme sistemi alıcı biriminin Matlab-Simulink blok şeması

Şekil 8.5.'te tasarlanan şifreli kaotik maskeleme modülasyonlu kaotik haberleşme sisteminin verici ve alıcı birimleriyle birlikte Matlab-Simulink blok şeması verilmiştir. Haberleşme sisteminde verici biriminden gönderilen modülasyonlu sinyalin yanında senkronizasyon için saat sinyali *Clk* de alıcı birime gönderilmektedir. Tasarlanan kaotik haberleşme sisteminin tüm testleri eklenebilir AWGN kanal modeli altında analiz edilmiştir.

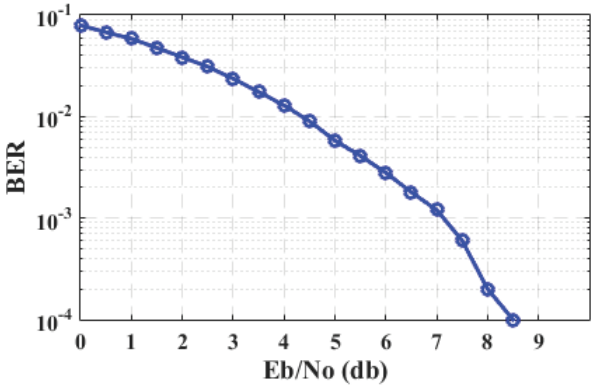


Şekil 8.5. Tasarlanan şifreli kaotik maskeleme modülasyonlu kaotik haberleşme sisteminin Matlab-Simulink blok şeması

8.2. Tasarlanan Şifreli Kaotik Haberleşme Sistemi İle Rastgele İkilik Bilgi İletimi

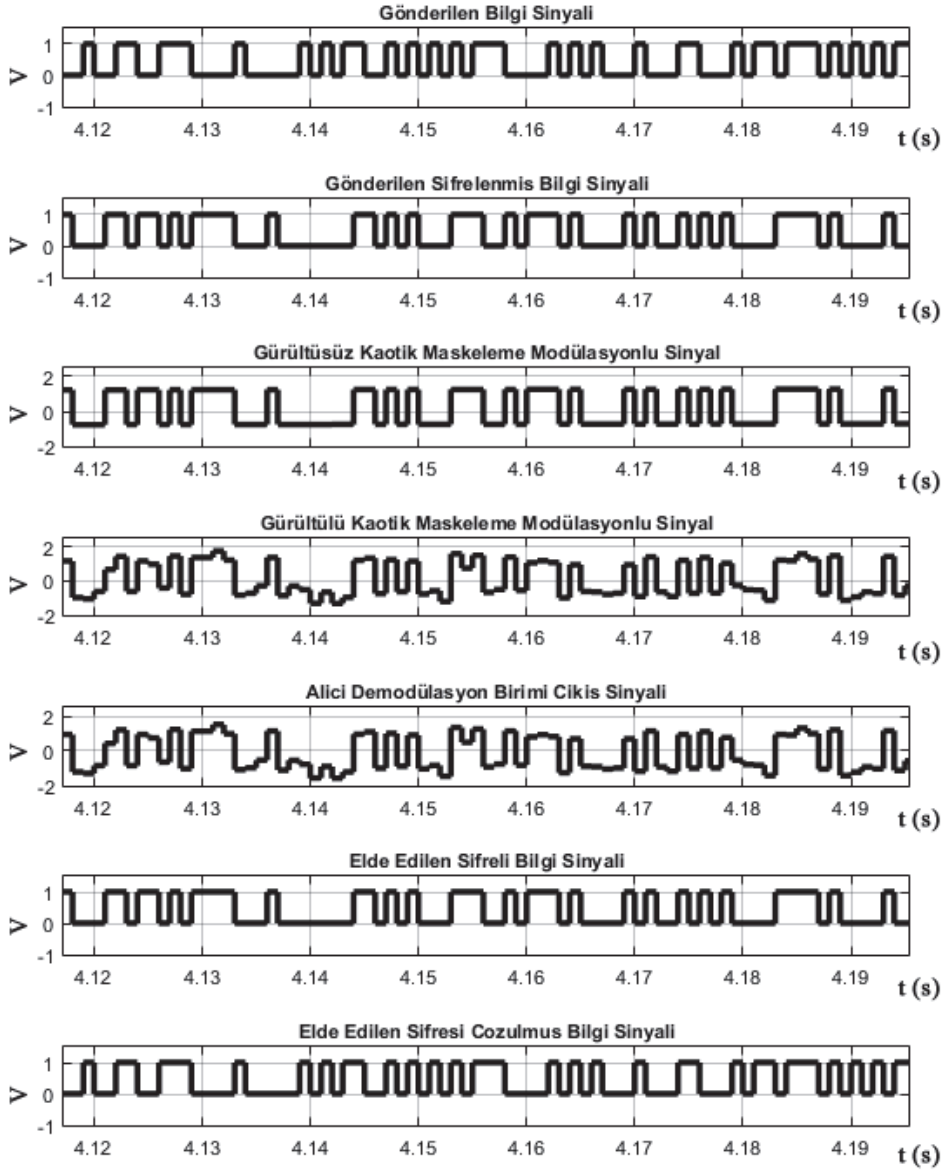
Tasarlanan şifreli kaotik haberleşme sistemi bu kısımda rastgele ikilik veriler gönderilerek test edilmiştir. Test işleminde Şekil 8.5.'te verilen tasarım kullanılmıştır. Haberleşme sisteminin 0-20 dB'lik E_b/N_0 AWGN kanal modeli altında BER analizi gerçekleştirilmiştir. Benzetim 10 saniye süre ile çalıştırılmıştır. Sistemde

bit süresi $T_b=0.001$ saniye alınmıştır. Analiz sonucu Şekil 8.6.'da verilmiştir. Sistemde 9dB E_b/N_0 değerine kadar çok başarılı bir şekilde verici tarafından gönderilen bilgi işareti alıcı taraftan elde edilmiştir.

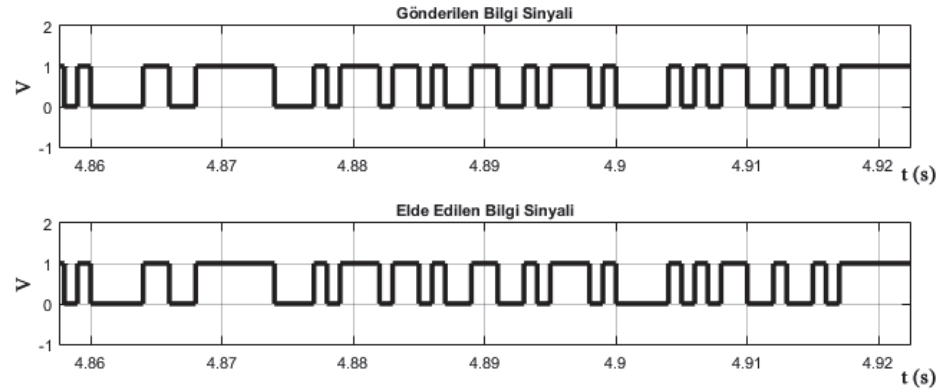


Şekil 8.6. Tasarlanan kaotik haberleşme sisteminin BER analizi grafiği

Şekil 8.7.'de 9dB E_b/N_0 değerindeki AWGN gürültü modeli altında haberleşme sisteminin: verici birim tarafından gönderilen bilgi işareti, verici birim tarafından gönderilen şifrelenmiş bilgi işareti, verici birim tarafından çıkan gürültüsüz kaotik maskeleme modülasyonlu sinyal, alıcı birime gelen gürültülü kaotik maskeleme modülasyonlu sinyal, alıcı demodülasyon birimi çıkış sinyali, alıcı birim tarafından elde edilen şifreli bilgi sinyali ve alıcı birim tarafından elde edilen şifresi çözülmüş bilgi sinyallerinin grafikleri verilmiştir. Şekil 8.8.'de ise verici taraftan gönderilen bilgi sinyali ve alıcı taraftan elde edilen bilgi sinyali birlikte daha net olarak verilmiştir.



Şekil 8.7. Tasarlanan kaotik haberleşme sisteminin sinyal çıkışları



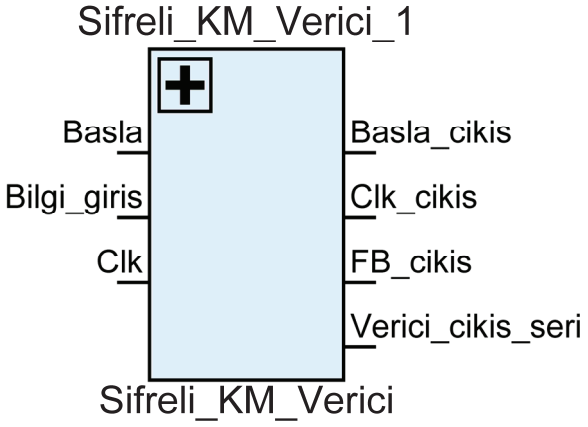
Şekil 8.8. Tasarlanan kaotik haberleşme sisteminde verici taraftan gönderilen bilgi sinyali ve alıcı tarafından elde edilen bilgi sinyali

BÖLÜM 9. FPGA TABANLI ŞİFRELİ KAOTİK HABERLEŞME SİSTEMİ TASARIMI VE GERÇEKLEŞTİRİLMESİ

Bu bölümde elde edilen yeni kaotik sistem ve bu sistem kullanılarak tasarlanan GRSÜ birimi (7. Bölümde tasarım çalışmaları verilmişti) ve kaotik maskeleme modülasyon yöntemiyle VHDL programlama dili kullanılarak FPGA üzerinde şifreli kaotik haberleşme sistemi tasarlanmış ve gerçekleştirilmiştir. Haberleşme sistemi, metin bilgisi, görüntü bilgisi ve ses bilgisi kullanılarak test edilmiştir. Tez çalışmasında gerçekleştirilen şifreli kaotik haberleşme sistemi, literatürdeki rastlanılan benzer iki çalışma ile karşılaştırılmıştır.

9.1. FPGA Tabanlı Şifreli Kaotik Haberleşme Sistemi Verici Birimi Tasarımı

Şifreli kaotik haberleşme sistemi *Verici* birimi tasarımında 3. Bölümde ayrıntısı verilen kaotik maskeleme (KM) (Chaotic masking – CM) yöntemi kullanılmıştır. Kaotik haberleşme sisteminin *Verici* biriminin (*Sifreli_KM_Verici*) pin diyagramı Şekil 9.1.'de verilmiştir. *Verici* birimi, "*Basla*", "*Bilgi_giris*" ve "*Clk*" giriş pinleri ile "*Basla_cikis*", "*Clk_cikis*", "*FB_cikis*" ve "*Verici_cikis_seri*" çıkış pinlerine sahiptir. *Verici* biriminin sahip olduğu pinlerin açıklamaları Tablo 9.1.'de verilmiştir.

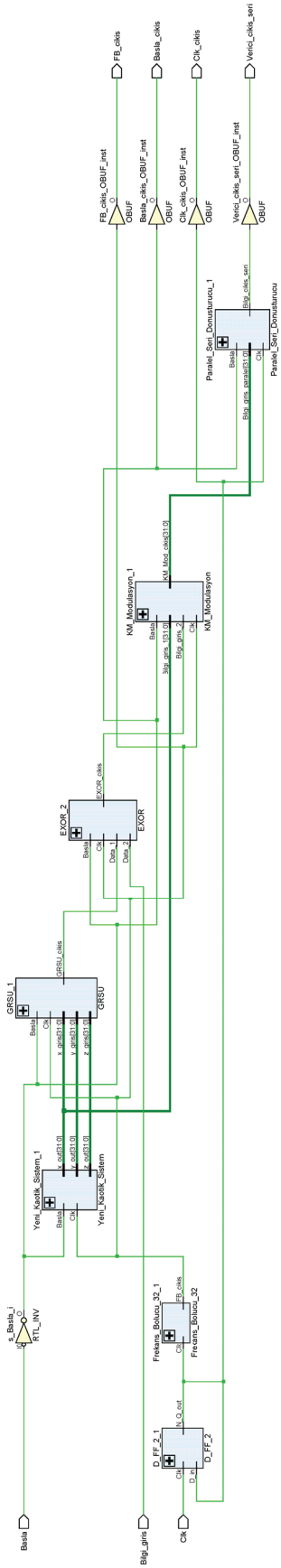


Şekil 9.1. FPGA tabanlı şifreli kaotik haberleşme sistemi verici biriminin pin diyagramı

Tablo 9.1. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi pin açıklamaları

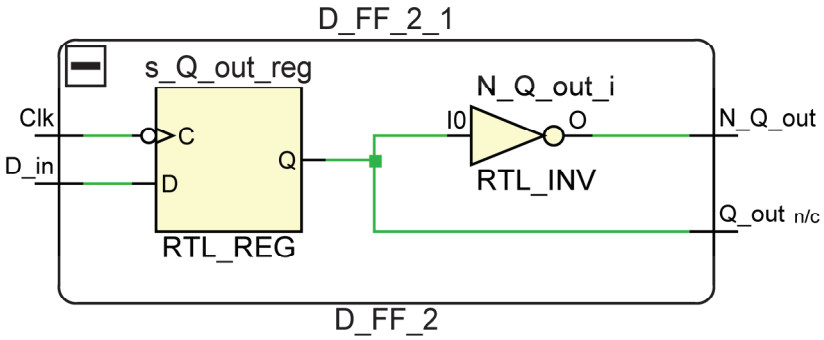
Pin İsmi	Açıklama
Basla	Sistemin aktif veya pasif edilmesini sağlayan giriş pinidir. Bu pin Lojik-1 olduğunda verici birimi aktif olur.
Bilgi_giris	Gönderilecek bilgiler için giriş pinidir.
Clk	Sistemin saat sinyali giriş pinidir.
Basla_cikis	Verici birimin aktif veya pasif olduğunu belirten çıkış pinidir. Bu pin Lojik-1 olduğunda verici birimi aktif anlamındadır.
Clk_cikis	Verici birimin kendi iç birimleri için kullandığı saat sinyali çıkış pinidir.
FB_cikis	Verici birimde bulunan "Frekans Bölücü" biriminin çıkış pinidir.
Verici_cikis_seri	Alıcı birime gönderilecek bilgilerin iletildiği çıkış pinidir.

Verici biriminin iç yapısı Şekil 9.2.'de verilmiştir. *Verici* biriminde bulunan *Yeni_Kaotik_Sistem* birimi Bölüm 5., *GRSU* birimi ise Bölüm 7.'de ayrıntısı verilen yapıların aynısıdır.



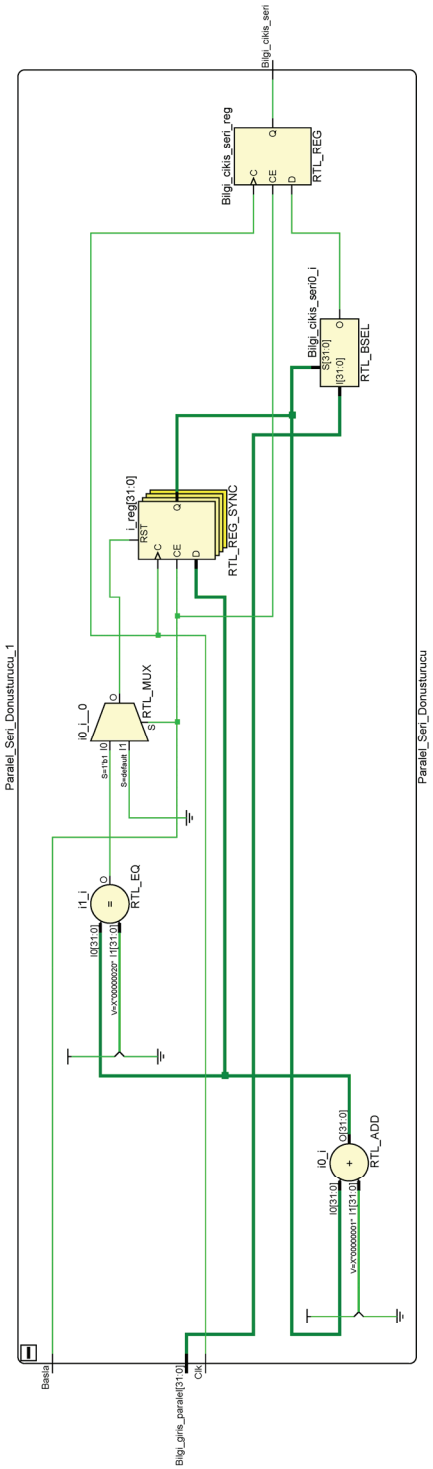
Şekil 9.2. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi iç yapısı

Verici birimindeki saat sinyali (*Clk*) pininden gelen saat sinyali *D/FF_2* biriminden geçirilerek hem verici sistemine hem de *Alıcı* birime gönderilmek üzere çıkışa *Clk_cikis* pinine aktarılmıştır. Bu sayede saat sinyalinde (*Clk*) meydana gelebilecek metastabilite olasılığı durumu azaltılmıştır. Bu amaçla kullanılan *D/FF_2* biriminin iç yapısı Şekil 9.3.'te verilmiştir.

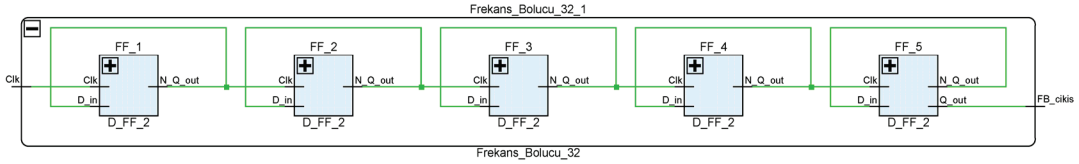


Şekil 9.3. D/FF birimi iç yapısı

Verici birimlerindeki işlemler IEEE 32 bitlik kayan noktalı sayı formatında gerçekleştirilmektedir. Bu nedenle alıcı birime gönderilecek bilgiler 32 bit uzunluğundadır. Bu şekilde veri iletiminin yapılması için 32 adet iletim kanalı gereklidir. Daha verimli iletişim için bilgilerin tek iletim kanalı üzerinden seri olarak gönderilmesi hedeflenmiş ve bunun için sistemde *Paralel_Seri_Donusturucu* birimi tasarlanmıştır. *Paralel_Seri_Donusturucu* biriminde 32 bitlik veri tek tek 32 saat darbesinde gönderilmektedir. *Paralel_Seri_Donusturucu* biriminin iç yapısı Şekil 9.4.'te verilmiştir. Gelen her verinin kaçırılmadan *Alıcı* birime gönderilmesi için *Paralel_Seri_Donusturucu* biriminin diğer tüm birimlerden 32 kat daha hızlı çalışması gerekmektedir. Bunun için ana saat sinyali frekansı tasarlanan *Frekans_Bolucu_32* birimi ile 32'ye bölünmektedir. Bu sayede *Paralel_Seri_Donusturucu* birimi ana saat sinyali frekansı f_{Clk} ile çalışırken, diğer tüm birimler f_{Clk} */32* hızında çalışmaktadır. Şekil 9.5.'te *Frekans_Bolucu_32* biriminin iç yapısı verilmiştir. Şekilden görüleceği üzere *Frekans_Bolucu_32* birimi beş adet D/FF bağlantısı sayesinde girişindeki sinyal frekansını 32'ye bölmektedir ($2^n = 2^5 = 32$).

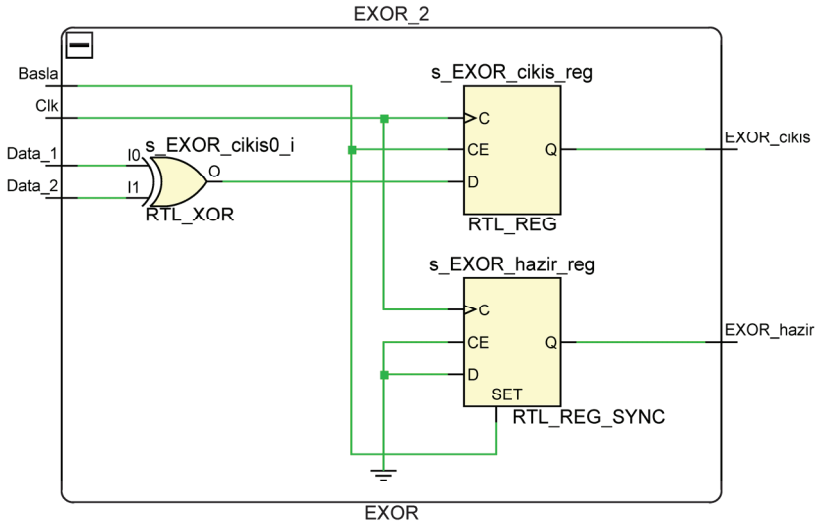


Şekil 9.4. Paralel_Seri_Donusturucu birimi iç yapısı



Şekil 9.5. Frekans_Bolucu_32 birimi iç yapısı

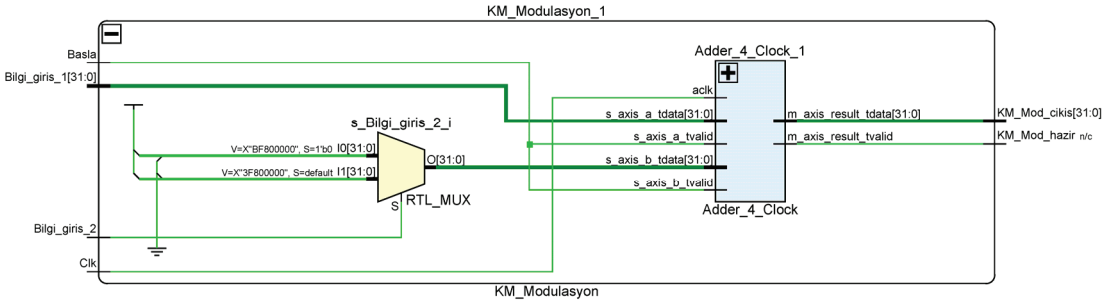
Verici biriminde, gelen bilgi sinyali *GRSU* biriminden üretilen rastgele sayı ile EXOR işlemine tabi tutularak şifrelenir. Bu işlem için *Verici* biriminde iç yapısı Şekil 9.6.'da verilen *EXOR* birimi tasarlanmıştır. Bilgi sinyali ile *GRSU* birimi çıkışından gelen bilgi *EXOR* biriminin girişlerini oluşturur. *EXOR* birimi çıkışından elde edilen şifrelenmiş bilgi sinyali, kaotik maskeleme modülasyonuna tabi tutulmak üzere tasarlanan *KM_Modulasyon* birimine gönderilir.



Şekil 9.6. Verici birimindeki EXOR biriminin iç yapısı

KM_Modulasyon biriminde, kaotik maskeleme modülasyon işlemi gerçekleştirilmektedir. Bu işlem için, *Yeni_Kaotik_Sistem* biriminden gelen kaotik sinyal ile (*x_out*) *EXOR* biriminden gelen sinyal *KM_Modulasyon* biriminde toplanmaktadır. *KM_Modulasyon* biriminin iç yapısı Şekil 9.7.'de verilmiştir. *KM_Modulasyon* biriminde gerçekleştirilen toplama işlemi IEEE 32 bitlik kayan noktalı sayı formatında olmaktadır. *Yeni_Kaotik_Sistem* biriminden gelen kaotik sinyal 32 bitlik veri olarak gelmektedir. Fakat *EXOR* biriminden gelen sinyal tek bitliktir (0 veya 1). Bu nedenle *KM_Modulasyon* biriminde toplama işlemi gerçekleştirilmeden önce *EXOR* biriminden gelen tek bitlik bilginin ilk önce IEEE 32

bitlik kayan noktalı sayı formatı karşılığına çevrilmesi gerekmektedir. Ayrıca *EXOR* biriminden gelen bilgide bulunan "0" sayısal değerinin kaotik maskeleme modülasyon birimindeki toplama işlemine bir etkisi yoktur. İçinde çarpma işlemi bulunan modülasyon tekniklerinde de "0" sayısal bilgisi sinyal değerini tümüyle sıfırladığından güvenlik sorunu oluşturabilmektedir. Bu nedenle *EXOR* biriminden gelen şifrelenmiş "1, 0" yapısındaki sayısal bilgi *KM_Modulasyon* biriminde "1, -1" değerindeki IEEE 32 bitlik kayan noktalı sayı formatına çevrilmektedir. *EXOR* biriminden gelen bilgi Lojik-0 ise 0xBF800000, Lojik-1 ise 0x3F800000 değerlerine çevrilmektedir. Bu aşamadan sonra *KM_Modulasyon* biriminde bulunan *Adder_4_Clock* biriminde toplama işlemi gerçekleştirilmektedir.



Şekil 9.7. KM_Modulasyon birimi iç yapısı

Verici biriminde girişten gelen bilgi *EXOR* biriminde 1 saat periyodu sürede işlem görmekte ve *KM_Modulasyon* birimine gönderilmektedir. *KM_Modulasyon* birimi de toplama işlemini 4 saat periyodu içinde gerçekleştirmektedir. Sonuçta *Verici* birimine gelen bilgi sinyali toplam da 5 saat periyodu süresinde *Verici* birimi çıkışına gelmiş olur.

FPGA tabanlı şifreli kaotik haberleşme sistemi *Verici* birimi, Xilinx Vivado Design Suite v2015.4 programında VHDL ile tasarlanmış ve sentezleme (Synthesis) ile gerçekleme (Implementation) işlemlerine tabi tutulmuştur. Tablo 9.2.'de şifreli kaotik haberleşme sistemi *Verici* biriminin Xilinx Artix-7 ailesi xc7a100tcsg324-1 modeli üzerinde gerçekleştirilen FPGA tasarımının çip istatistikleri verilmiştir. Sistemde maksimum çalışma frekansı 464,037 MHz ve minimum çalışma periyodu 2,155ns olarak elde edilmiştir. Tablo 9.2.'de verilen kısaltmaların açılımları şu şekildedir: LUT (Look-up Table, Başvuru Tablosu), LUTRAM (Look-up Table

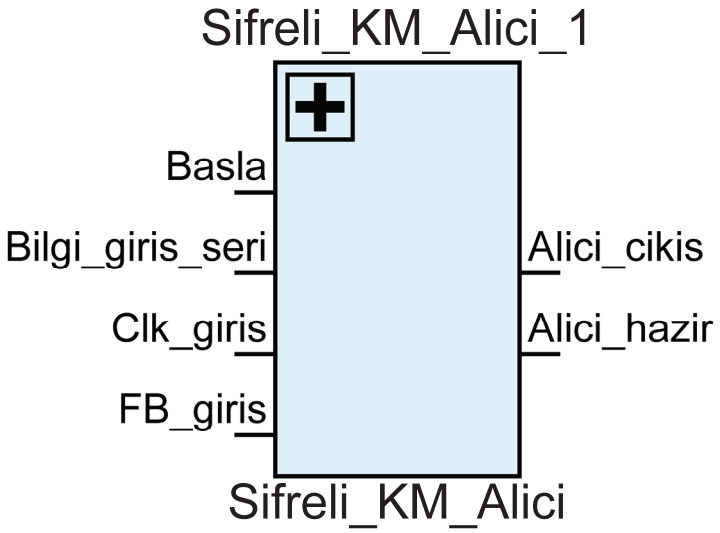
RAM), FF (Flip-Flop), DSP (Digital Signal Processor, Sayısal Sinyal İşleyici), IO (Input/Output, Giriş/Çıkış pin sayısı). Sistemde giriş-çıkış (IO) pinlerinin çalışma gerilim değerleri 3,3 V olarak ayarlanmıştır. Bu şartlar altında sistemin güç harcaması 207 mW'dır.

Tablo 9.2. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tasarımının çip istatistikleri

Kaynaklar	Mevcut	Kullanılan	Kullanım Oranı (%)
LUT	63400	2762	4,36
LUTRAM	19000	50	0,26
FF	126800	2400	1,89
DSP	240	50	20,83
IO	210	7	3,33

9.2. FPGA Tabanlı Şifreli Kaotik Haberleşme Sistemi Alıcı Birimi Tasarımı

Şifreli kaotik haberleşme sistemi *Alıcı* tasarımında 3. Bölümde ayrıntısı verilen kaotik maskeleme (KM) (Chaotic masking – CM) yöntemi kullanılmıştır. Kaotik haberleşme sisteminin *Alıcı* biriminin (*Sifreli_KM_Alici*) pin diyagramı Şekil 9.8.'de verilmiştir. *Alıcı* birimi, "*Basla*", "*Bilgi_giris_seri*", "*Clk_giris*" ve "*FB_giris*" giriş pinleri ile "*Alici_cikis*" ve "*Alici_hazir*" çıkış pinlerine sahiptir. *Alıcı* biriminin sahip olduğu pinlerin açıklamaları Tablo 9.3.'te verilmiştir.

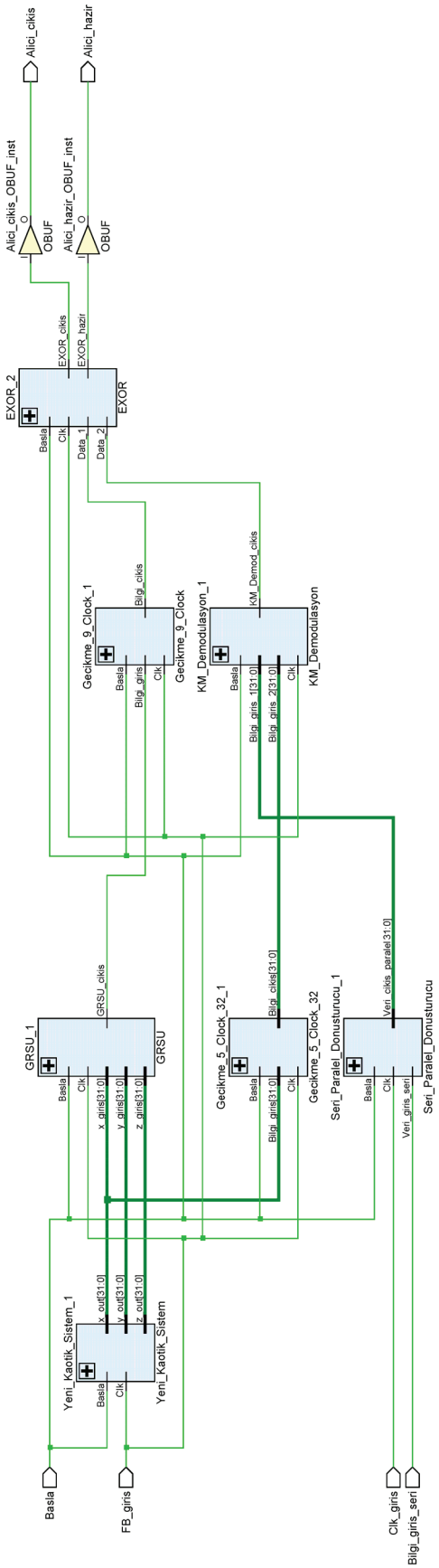


Şekil 9.8. FPGA tabanlı şifreli kaotik haberleşme sistemi alıcı biriminin pin diyagramı

Tablo 9.3. FPGA tabanlı şifreli kaotik haberleşme sistemi alıcı birimi pin açıklamaları

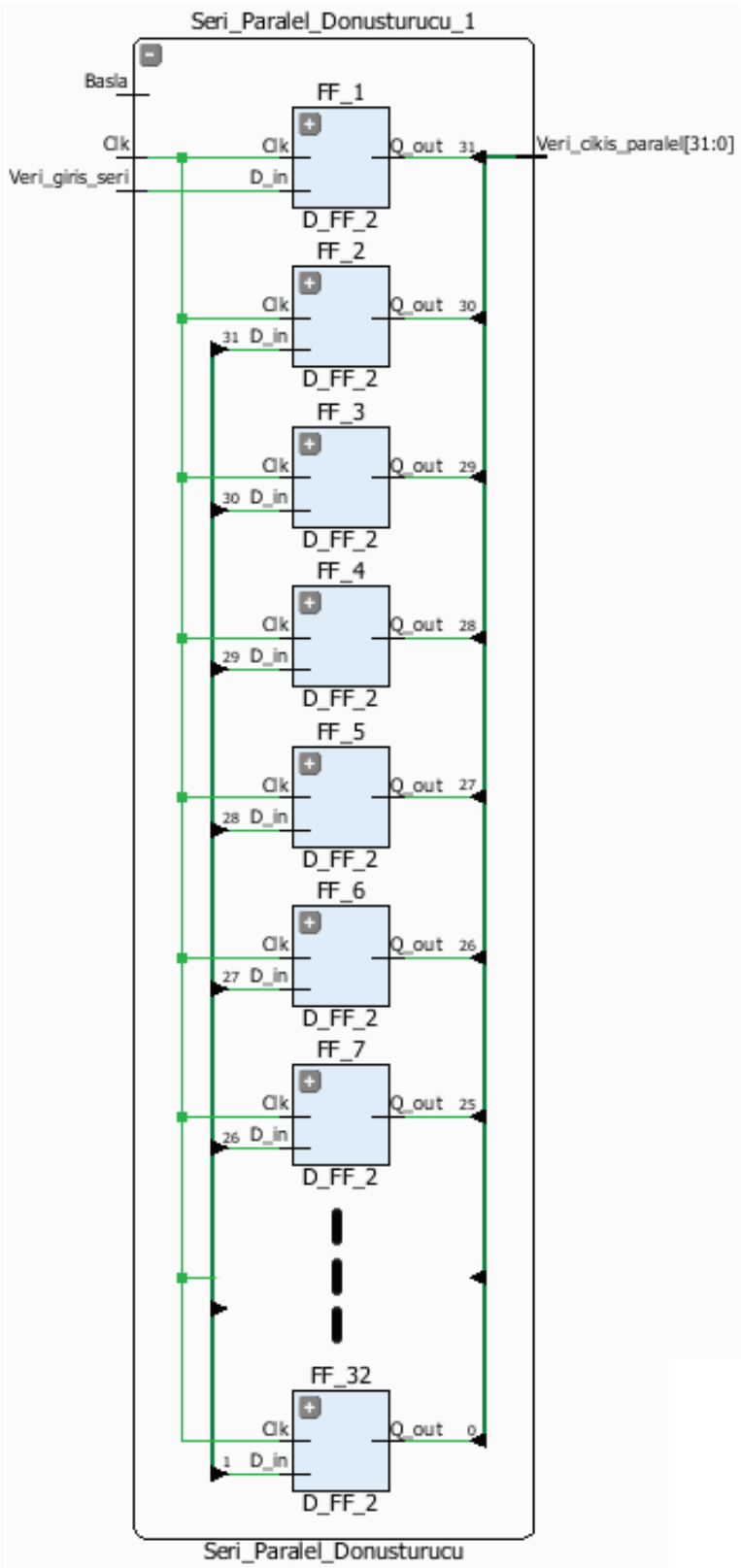
Pin İsmi	Açıklama
Basla	Sistemin aktif veya pasif edilmesini sağlayan giriş pinidir. Bu pin Lojik-1 olduğunda alıcı birimi aktif olur.
Bilgi_giris_seri	Gelen bilgiler için giriş pinidir.
Clk_giris	Sistemin saat sinyali (clock pulse) giriş pinidir.
FB_giris	Verici birimden gelen "Frekans Bölücü" biriminin çıkış pinidir.
Alici_cikis	Verici birimin aktif veya pasif olduğunu belirten çıkış pinidir.
Alici_hazir	Alıcı biriminin veri çıkışına hazır olup olmadığını belirten çıkış pinidir. Bu pin Lojik-1 olduğunda alıcı birimi veri çıkışına hazır demektir.

Alici biriminin iç yapısı Şekil 9.9.'da verilmiştir. *Sifreli_KM_Alici* biriminde bulunan *Yeni_Kaotik_Sistem* birimi ile *GRSU* birimi *Verici* birimindeki yapıların aynısıdır.



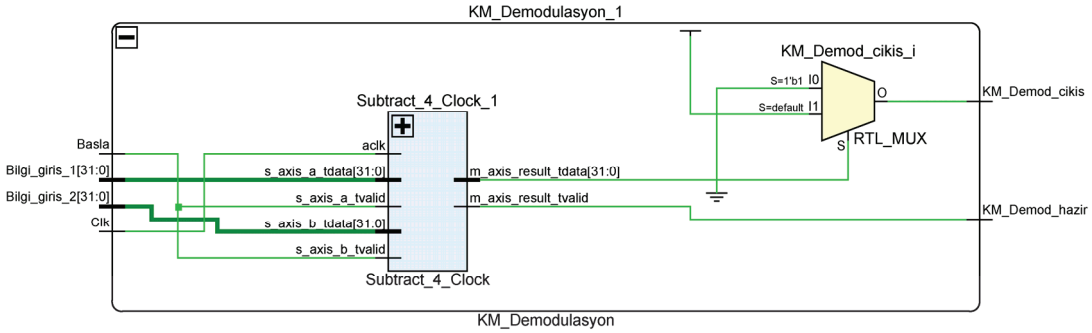
Şekil 9.9. FPGA tabanlı şifreli kaotik haberleşme sistemi alıcı birimi iç yapısı

Alıcı birimine seri olarak gelen IEEE 32 bitlik kayan noktalı sayı formatındaki bilgi tasarlanan *Seri_Paralel_Donusturucu* birimi ile 32 bitlik paralel bilgiye çevrilir. Seri bilginin paralel bilgiye çevirme işleminde 32 adet D/FF kullanılmıştır. *Seri_Paralel_Donusturucu* biriminin iç yapısı Şekil 9.10.'da verilmiştir.



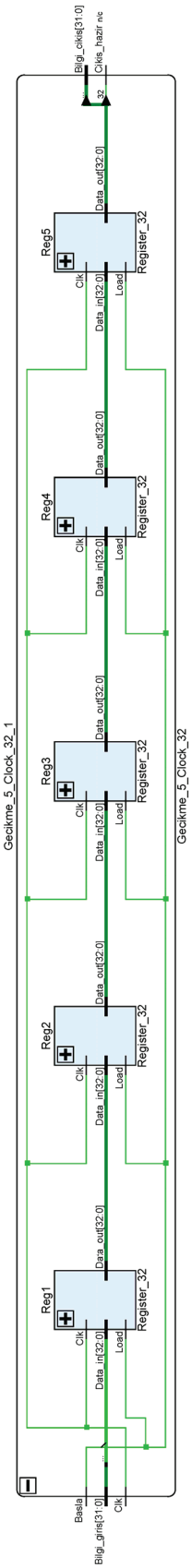
Şekil 9.10. Seri_Paralel_Donusturucu birimi iç yapısı

Verici biriminden gelen modüleli bilgi sinyalinin *Alıcı* biriminde demodülasyon işlemine tabi tutulması gerekmektedir. Demodülasyon işleminde *Verici* biriminden gelen bilgiden, *Yeni_Kaotik_Sistem* biriminden gelen kaotik sinyal *(x_out*) çıkartılır. Alıcı biriminde demodülasyon işlemi için *KM_Demodulasyon* birimi tasarlanmıştır. *KM_Demodulasyon* biriminin iç yapısı Şekil 9.11.'de verilmiştir. *KM_Demodulasyon* işlemi 4 saat periyodu süresinde işlemini icra etmektedir.



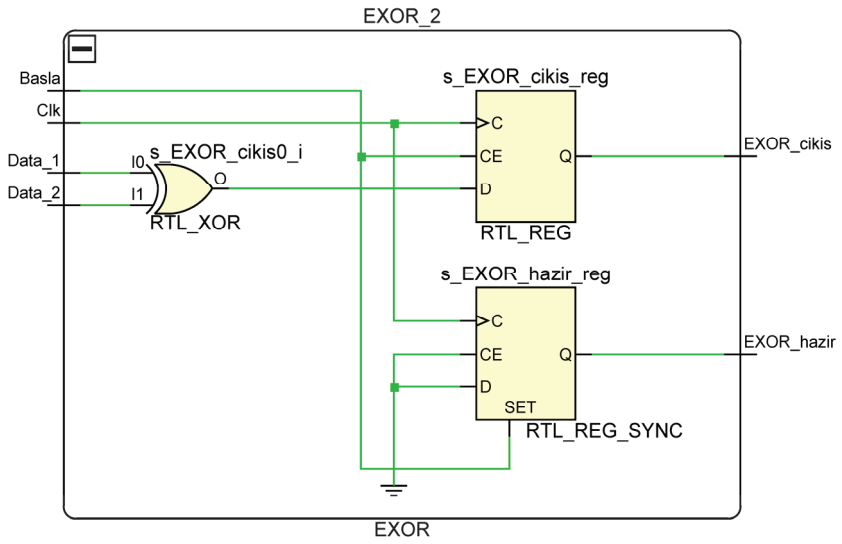
Şekil 9.11. KM_Demodulasyon birimi iç yapısı

Verici biriminde gönderilecek bilgi ilk olarak 5 saat periyodu sonrası alıcı birimine gelmekteydi. Bu nedenle *Alıcı* birimindeki *Yeni_Kaotik_Sistem* birimi çıkışı olan kaotik sinyalin (*x_out*) *KM_Demodulasyon* birimine gelmeden önce 5 saat periyodu süresince geciktirilmesi gerekmektedir. *Yeni_Kaotik_Sistem* biriminden çıkan 32 bitlik *x_out* kaotik sinyalinin 5 saat periyodu gecikme işlemi için *Gecikme_5_Clock_32* birimi tasarlanmıştır. *Gecikme_5_Clock_32* biriminin iç yapısı Şekil 9.12.'de verilmiştir. Gecikme işleminde 5 adet 32 bitlik girişlere sahip D/FF kullanılmıştır.



Şekil 9.12. Gecikme_5_Clock_32 birimi iç yapısı

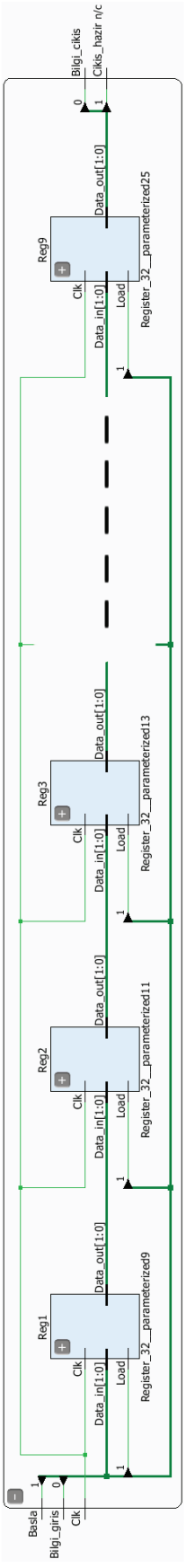
KM_Demodulasyon biriminden çıkan bilgi, *Verici* birimindeki şifreli bilgidir. Bu şifreli bilgiden asıl bilgi sinyalinin elde edilmesi için, şifreli bilginin *Alıcı* birimindeki *GRSU* biriminden gelen bilgi ile EXOR işlemine tabi tutulması gerekmektedir. EXOR işlemi için tasarlanan *EXOR* biriminin iç yapısı Şekil 9.13.'te verilmiştir.



Şekil 9.13. Alıcı birimindeki EXOR birimi iç yapısı

Fakat *KM_Demodulasyon* biriminden çıkan ilk doğru bilgi 9 saat periyodu sonrasında çıkmaktadır. Bu nedenle *GRSU* biriminden çıkan bilginin 9 saat periyodu kadar geciktirilmesi gerekmektedir. Bu gecikme işlemi için *Gecikme_9_Clock* birimi tasarlanmıştır. *Gecikme_9_Clock* biriminin iç yapısı Şekil 9.14.'te verilmiştir. 9 saat periyodu gecikme işlemi için *Gecikme_9_Clock* biriminde 9 adet D/FF kullanılmıştır.

Gecikme_9_Clock biriminden gelen bilgi ile *GRSU* biriminden gelen bilgi *EXOR* biriminde EXOR işlemine tabi tutulmaktadır. Bu bilgiler *EXOR* birimine 9 saat periyodu sonunda gelmektedirler. *EXOR* birimi de 1 saat periyodu süresinde işlemini gerçekleştirmektedir. Sonuçta *Verici* birimden gönderilen ilk bilgi *Alıcı* birimden toplamda 10 saat periyodu sonunda alınmaktadır.



Şekil 9.14. Gecikme_9_Clock birimi iç yapısı

FPGA tabanlı şifreli kaotik haberleşme sistemi *Alıcı* birimi, Xilinx Vivado Design Suite v2015.4 programında VHDL ile tasarlanmış ve sentezleme (Synthesis) ile gerçekleme (Implementation) işlemlerine tabi tutulmuştur. Tablo 9.4.'te şifreli kaotik haberleşme sistemi alıcı biriminin Xilinx Artix-7 ailesi xc7a100tcsg324-1 modeli üzerinde gerçekleştirilen FPGA tasarımının çip istatistikleri verilmiştir. Sistemde maksimum çalışma frekansı 392,927 MHz ve minimum çalışma periyodu 2,545 ns olarak elde edilmiştir. Tablo 9.4.'te verilen kısaltmaların açılımları şu şekildedir: LUT (Look-up Table, Başvuru Tablosu), LUTRAM (Look-up Table RAM), FF (Flip-Flop), DSP (Digital Signal Processor, Sayısal Sinyal İşleyici), IO (Input/Output, Giriş/Çıkış pin sayısı). Sistemde giriş-çıkış (IO) pinlerinin çalışma gerilim değerleri 3,3 V. olarak ayarlanmıştır. Bu şartlar altında sistemin güç harcaması 127 mW'dır.

Tablo 9.4. FPGA tabanlı şifreli kaotik haberleşme sistemi alıcı birimi tasarımının çip istatistikleri

Kaynaklar	Mevcut	Kullanılan	Kullanım Oranı (%)
LUT	63400	2718	4,29
LUTRAM	19000	68	0,36
FF	126800	2385	1,88
DSP	240	49	20,42
IO	210	6	2,86

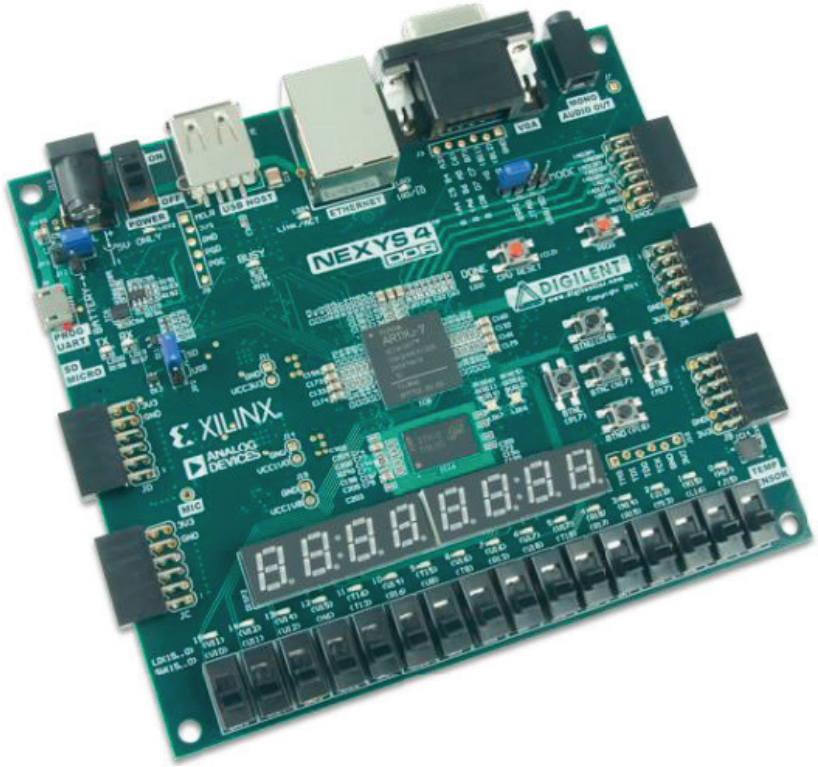
9.3. FPGA Tabanlı Şifreli Kaotik Haberleşme Sisteminin Gerçek Ortam Test Düzeneği

FPGA tasarımı yapılan şifreli kaotik haberleşme sistemi gerçek ortamda test edilmiştir. Gerçek ortam test işlemlerinde FPGA donanımı için, içinde Xilinx firmasının Artix-7 xc7a100tcsg324-1 modeli FPGA entegresi bulunan Digilent firmasının Nexys4 DDR FPGA geliştirme kartı kullanılmıştır. Tablo 9.5.'te Artix-7 xc7a100tcsg324-1 FPGA modelinin kaynakları verilmiştir

Tablo 9.5. Artix-7 xc7a100tcsg324-1 FPGA modeli kaynakları [177-179]

Mantık Hücresi	Slice	SLICEL	SLICEM	LUT	Dağıtılmış RAM (Kb)	Kaymalı Kaydedici (Kb)	Flip-Flop	DSP-48E1	Kullanıcı I/O Pin Sayısı
101440	15850	11100	4750	63400	1188	594	126800	240	210

Nexys4 DDR FPGA geliştirme kartı Şekil 9.15.'te verilmiştir. Nexys4 DDR FPGA geliştirme kartı içinde: 1 adet xc7a100tcsg324-1 FPGA entegresi, saat sinyali için 1 adet 100 Mhz kristal, 16 adet anahtar, 16 adet LED, 2 adet üç renkli LED, 8 adet 7-segment display, 1 adet seri Flash hafıza entegresi, 1 adet USB-UART entegresi, 1 adet 12 bitlik VGA çıkışı, 1 adet üç eksenli ivmeölçer, 1 adet sıcaklık sensörü, 1 adet mikrofon, 1 adet 128MB DDR2 hafıza birimi, 1 adet USB konnektörü, 1 adet JTAG konnektör, 1 adet mikro SD kart konnektörü, 1 adet mono ses çıkış konnektörü, 1 adet Ethernet konnektörü, 1 adet XADC sinyalleri çıkışı için PMOD konnektör, her biri 8 çıkışlı 4 adet genel giriş/çıkış (I/O) PMOD konnektörü bulunmaktadır [191].

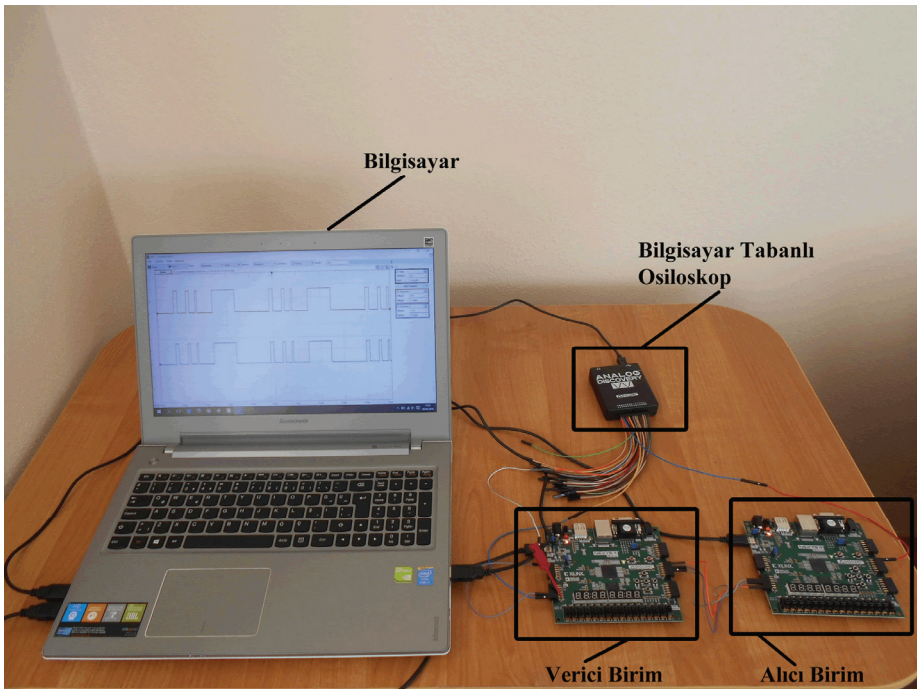


Şekil 9.15. Digilent Nexys4 DDR FPGA geliştirme kartı [191]

Tasarlanan haberleşme sisteminin FPGA üzerinde gerçek ortam testleri ölçümleri için Digilent firmasının Analog Discovery adlı bilgisayar tabanlı osiloskop cihazı kullanılmıştır. Analog Discovery bilgisayar tabanlı osiloskop cihazının görünümü Şekil 9.16.'da verilmiştir. Analog Discovery bilgisayar tabanlı osiloskop cihazı şu özelliklere sahiptir: 1 MΩ empedanslı 5 MHz bant genişlikli saniyede 100 Mega-örnek/saniye (Msample/sec) kapasitesine sahip ±25V'luk 14 bitlik 2 kanal osiloskop, 5 MHz bant genişlikli saniyede 100 Mega-örnek/saniye kapasitesine sahip 14 bit 2 kanal sinyal üreteci, 100 Mega-örnek/saniye kapasiteli 3,3V 16 kanal lojik analizör, 100 Mega-örnek/saniye kapasiteli 3,3V 16 kanal patern üreteci, spektrum analizörü, 1 Hz – 10 MHz arası network analizör, çeşitli dijital protokol analizörü (SPI, I^2C, UART, Paralel), 2 adet ±5V 50mA güç kaynağı, çoklu ölçüm cihazlarının senkronizasyonu için 2 adet dijital tetikleme sinyali, tek kanal AC-DC ±25V voltmetre [192].

Şekil 9.16. Digilent Analog Discovery bilgisayar tabanlı osiloskop cihazı [192]

Tasarlanan haberleşme sistemi test düzeneği için: 1 adet *Verici* birimi ve 1 adet de *Alıcı* birimi olmak üzere 2 adet Nexys4 DDR FPGA geliştirme kartı, ölçümler için 1 adet Analog Discovery bilgisayar tabanlı osiloskop ve bilgisayar kullanılmıştır. Şekil 9.17.'de gerçek ortam test düzeneği verilmiştir.



Şekil 9.17. FPGA tabanlı şifreli kaotik haberleşme sistemi test düzeneği

Nexys4 DDR FPGA geliştirme kartında saat sinyali için 100 MHz'lik kristal bulunmaktadır. Tasarlanan haberleşme sisteminde ana çalışma saat sinyali frekansı (f_{Clk}) 100 MHz'dir. Ana saat sinyali haberleşme sistemi *Verici* biriminde bulunan *D/FF* birimi nedeniyle 2'ye bölünmektedir ($f_{D/FF}$). Denklem 9.1'de *D/FF* birimi çıkışından elde edilen saat sinyali frekansı verilmiştir. Ayrıca paralel bilgilerin seriye dönüştürülmesi işlemleri için *Frekans_Bolucu_32* birimi girişine gelen $f_{D/FF}$ sinyali de 32'ye bölünmektedir (f_{FB}). Denklem 9.2'de *Frekans_Bolucu_32* birimi çıkışından elde edilen saat sinyali frekansı verilmiştir. Denklem 9.3'te verildiği üzere haberleşme sisteminin çalışma hızı f_S = 1,5625 MHz ve çalışma hızı periyodu ise T_S = 0,64µs.'dir. Sonuç olarak haberleşme sistemi en fazla 1,5625 Mbps (bps - bit per second – Saniyedeki bit sayısı) hızındaki verileri doğru bir şekilde işleyebilmektedir.

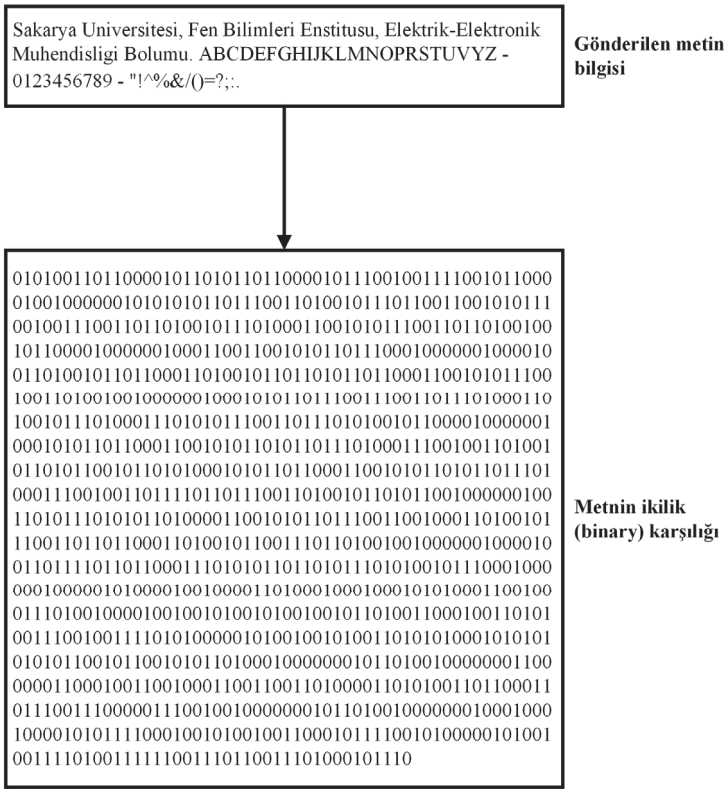
$$f_{D/FF} = \frac{f_{Clk}}{2} = \frac{100MHz}{2} = 50MHz \tag{9.1}$$

$$f_{FB} = \frac{f_{D/FF}}{32} = \frac{50MHz}{32} = 1,5625MHz \tag{9.2}$$

$$f_S = 1,5625MHz, \quad T_S = \frac{1}{f_S} = \frac{1}{1,5625MHz} = 0,64\mu s. \tag{9.3}$$

9.4. Tasarlanan FPGA Tabanlı Şifreli Kaotik Haberleşme Sistemi İle Metin Bilgisi İletimi

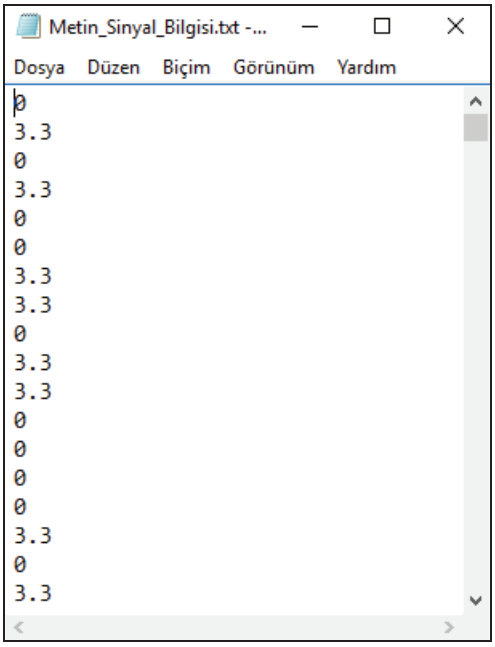
Bu kısımda tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi *Verici* biriminden bir metin bilgisi gönderilmiş ve *Alıcı* birimden alınmıştır. Örnek olarak alınan metin bilgisi toplam 140 adet karakterden oluşmaktadır. Metin bilgisinde bulunan karakterler ilk önce sekiz bitlik olarak ikilik (binary) Genişletilmiş (Extended) ASCII (American Standart Code for Information Interchange – Bilgi Değişimi İçin Amerikan Standart Kodlama Sistemi) kod karşılıklarına çevrilmiş ve elde edilen veri, haberleşme sistemi *Verici* birimi tarafından gönderilmiştir. Örnek metnin ikilik karşılığı toplam 140x8= 1120 adet bitten (0 ve 1) oluşmaktadır. Şekil 9.18.'de örnek olarak gönderilen metin verisi ve bu verinin ikilik Genişletilmiş ASCII kod karşılığı verilmiştir.



Şekil 9.18. FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden gönderilen metin bilgisi ve ikilik (binary) karşılığı

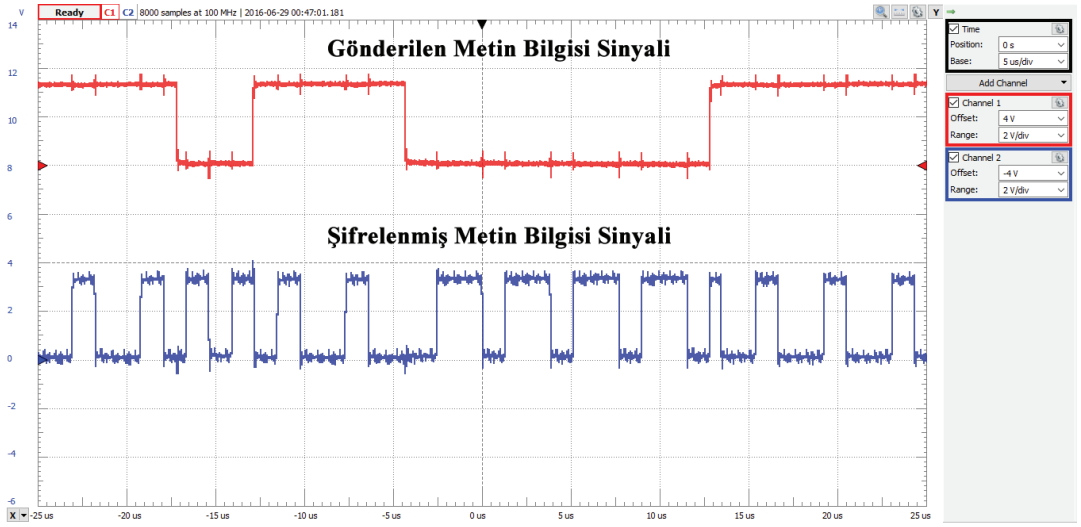
Metin bilgisinin gönderimi için ilgili metin bilgisinin ikilik karşılığı gerilim karşılıkları olarak dosyaya yazılmış ve Analog Discovery bilgisayar tabanlı osiloskop cihazı ile bu bilgiler 1,5 Mbps hızında sinyal olarak ürettirilmiştir. Lojik-0 değeri 0V ve Lojik-1 değeri ise 3,3V ile temsil edilmiştir. Bu işlem için oluşturulan sinyal bilgisi dosyasından örnek bir görüntü Şekil 9.19.'da verilmiştir. Gönderilen verinin bit periyodu süresi Denklem 9.4'te verildiği gibi $T_{bit} \approx 0{,}67$ µs.'dir.

$$T_{bit} = \frac{1}{f_{bit}} = \frac{1}{1{,}5MHz} \cong 0{,}67\mu s. \tag{9.4}$$



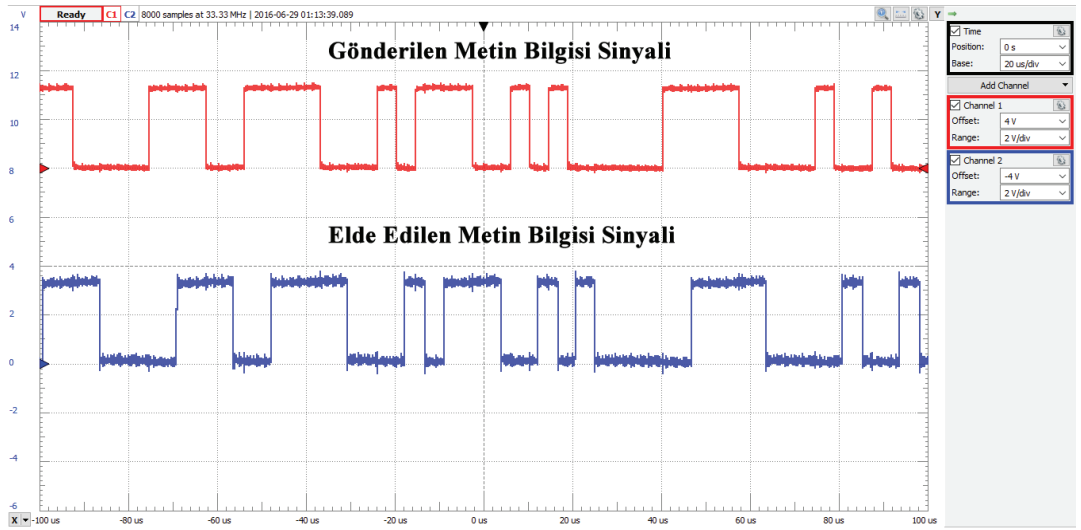
Şekil 9.19. FPGA tabanlı şifreli haberleşme sistemi üzerinden gönderilen metin bilgisi dosyasından örnek bir görüntü

Şekil 9.20.'de 1,5 Mbps hızında *Verici* birimi girişine uygulanan metin bilgisi sinyali ile *Verici* birim tarafından şifrelenmiş metin bilgisi sinyalinin Analog Discovery bilgisayar tabanlı osiloskop cihazı ile yapılan ölçüm görüntüsü verilmiştir (Volt/Div: 2V, Time/Div: 5µs). Şekil 9.20.'de verilen şifrelenmiş bilgi sinyali ayrıca *Verici* birimdeki *KM_Modulasyon* biriminde kaotik sinyal ile toplanarak iletim ortamına verilmektedir. Böylece gönderilen bilgi sinyali ile iletim ortamındaki bilgi sinyali birbirinden çok farklı olmaktadır.



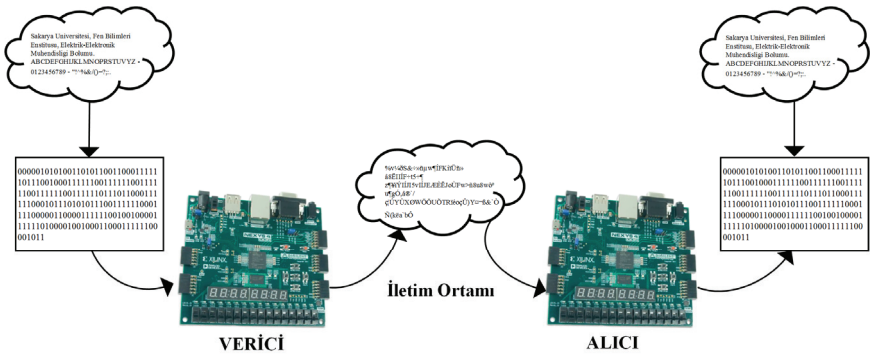
Şekil 9.20. Gönderilen metin bilgisi sinyali ile şifrelenmiş metin bilgisi sinyali osiloskop görüntüsü

Şekil 9.21.'de *Verici* birim tarafından gönderilen metin bilgisi sinyali ile *Alıcı* birimi çıkışından elde edilen sinyal bilgilerinin Analog Discovery bilgisayar tabanlı osiloskop cihazı ile yapılan ölçüm görüntüsü verilmiştir (Volt/Div: 2V, Time/Div: 5µs). Şekilden görüleceği üzere *Verici* biriminden gönderilen metin bilgisi *Alıcı* biriminde başarılı bir şekilde elde edilmiştir. Haberleşme sisteminin işlem süresi daha önce belirtildiği gibi 10 saat periyodu olduğu için *Alıcı* biriminden alınan bilgi verici birimden gönderilen bilginin 10xTs = 10x0,64 = 6,4µs. daha gecikmiş halidir.

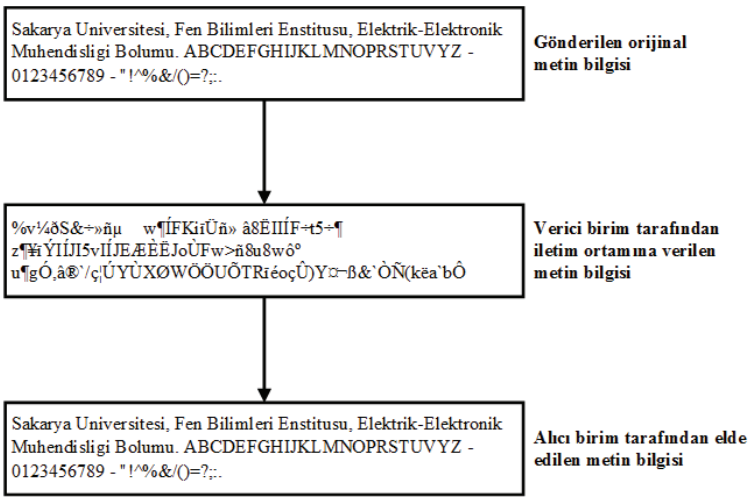


Şekil 9.21. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tarafından gönderilen metin bilgisi sinyali ile alıcı birimi tarafından elde edilen metin bilgisi sinyali

Tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden metin bilgisi gönderimi ve alımı test işleminde *Alıcı* birimden elde edilen bilgilerin dosyaya yazdırılması işlemi için Xilinx Vivado Design Suite 2015.4 programının kendi simülatörü kullanılmıştır. Şekil 9.18.'de verilen metin bilgisinin ikilik karşılığı *Verici* birimine bilgi sinyali olarak girilmiştir. *Verici* birimi ile *Alıcı* birimi çıkışlarından alınan ikilik bilgiler tekrar Genişletilmiş ASCII tablosu kullanılarak karakter karşılıklarına çevrilmiştir. Şekil 9.22.'de metin bilgisinin gönderimi ve alımı işlemi verilmiştir. Şekil 9.23.'te gönderilen metin bilgisi, *Verici* birim tarafından iletim ortamına aktarılan şifreli ve kaotik maskeleme modülasyonlu metin bilgisi ile *Alıcı* birim tarafından elde edilen metin bilgisi verilmiştir. Şekil 9.23.'ten görüleceği üzere *Verici* biriminden gönderilen metin bilgisi başarılı bir şekilde *Alıcı* birim tarafından tekrar elde edilmiştir.



Şekil 9.22. FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden metin bilgisi gönderimi ve alımı



Şekil 9.23. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tarafından gönderilen orijinal metin bilgisi sinyali, verici birim tarafından iletim ortamına verilen bilgi ile alıcı birim tarafından elde edilen metin bilgisi sinyali

İki değişken arasındaki ilişkinin ölçüsü bu iki değişkenin *koveryansı (Cov)* ile belirlenebilir. İki değişken arasındaki ilişki için, ölçüm birimindeki değişimden etkilenmeyen standart bir ölçü olarak da *korelasyon katsayısı (ρ)* kullanılmaktadır. Böylece korelasyon katsayısı ile iki değişkenin birbirinden ne kadar farklı olduğu belirlenebilir. Korelasyon katsayısı $-1 \leq \rho(x,y) \leq 1$ arasında değişir. Korelasyon katsayısı 0 ise iki değişken arasında doğrusal bir ilişki olmadığını belirtir. Korelasyon katsayısı -1 ise iki değişken arasında ters yönlü tam doğrusal bir ilişki olduğunu belirtir. Korelasyon katsayısı 1 ise iki değişken arasında aynı yönlü tam doğrusal bir ilişki olduğunu belirtir. Korelasyon katsayısı sıfır (0) değerine ne kadar yakınsa iki değişken birbiri ile o derece ilişkisizdir. Korelasyon katsayısı ρ, Denklem 9.5 ile hesaplanır. Denklem 9.5'teki *Var* sembolü varyansı ifade etmektedir [193-195].

$$\rho(\mathrm{x},\mathrm{y}) = \frac{Cov(x,y)}{\sqrt{Var(x)Var(y)}} \tag{9.5}$$

Gönderilen metin bilgisi ile *Verici* birim tarafından iletim ortamına aktarılan şifreli ve kaotik maskeleme modülasyonlu metin bilgisi arasındaki korelasyon katsayısı Matlab programında hesaplatılmış ve 0,0202 olarak elde edilmiştir. Buradan görüldüğü gibi korelasyon katsayısı değeri sıfıra çok yakın olduğundan gönderilen metin bilgisi ile *Verici* birim tarafından iletim ortamına gönderilen metin bilgisi arasındaki ilişki çok çok düşüktür. Bu sayede istenmeyen girişimler ile gönderilen metin bilgisinin elde edilmesi çok zordur.

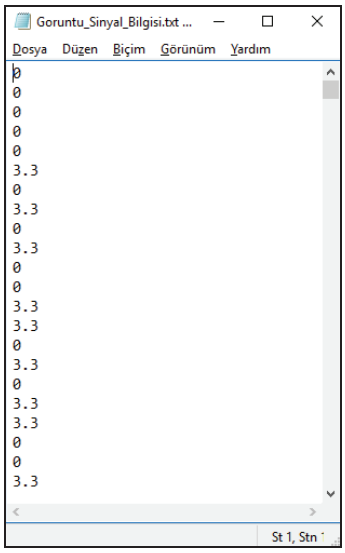
9.5. Tasarlanan FPGA Tabanlı Şifreli Kaotik Haberleşme Sistemi İle Görüntü Bilgisi İletimi

Bu kısımda tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi *Verici* biriminden bir görüntü bilgisi gönderilmiş ve *Alıcı* birimden alınmıştır. Görüntü olarak Matlab programının veri tabanında bulunan "camaraman.tif" adlı 256x256 pikselden oluşan gri tonlamalı görüntü dosyası kullanılmıştır. İlgili görüntü Şekil 9.24.'te verilmiştir. İlk olarak görüntünün piksel değerleri 256x256 işaretsiz

tam sayı türünde matris olarak elde edilmiştir. Ardından işaretsiz tam sayı değerleri 8 bitlik ikilik (binary) karşılıklarına çevrilmiştir. Elde edilen 8 bitlik ikilik değerler daha sonra sütun matrisine çevrilmiştir. Böylece görüntü bilgisi olarak tek tek gönderilecek bit değerleri 0 ve 1 şeklinde elde edilmiştir. Örnek görüntü toplam 256x256x8= 524288 adet bitten (0 ve 1) oluşmaktadır.

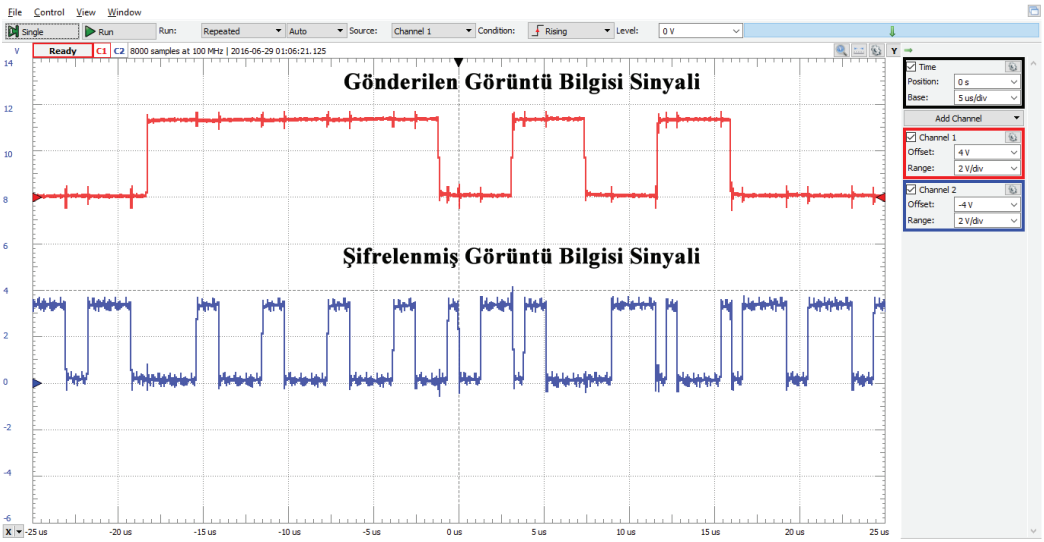
Şekil 9.24. FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden gönderilen görüntü

Görüntü bilgisinin gönderimi için görüntü bilgisinin ikilik karşılığı gerilim karşılıkları olarak dosyaya yazılmış ve Analog Discovery bilgisayar tabanlı osiloskop cihazı ile bu bilgiler 1,5 Mbps hızında sinyal olarak ürettirilmiştir. Lojik-0 değeri 0V ve Lojik-1 değeri ise 3,3V ile temsil edilmiştir. Bu işlem için oluşturulan sinyal bilgisi dosyasından örnek bir görüntü Şekil 9.25.'te verilmiştir. Gönderilen verinin bit periyodu süresi daha önce Denklem 9.4'te açıklandığı gibi $T_{bit} \approx 0{,}67\mu s$.'dir.



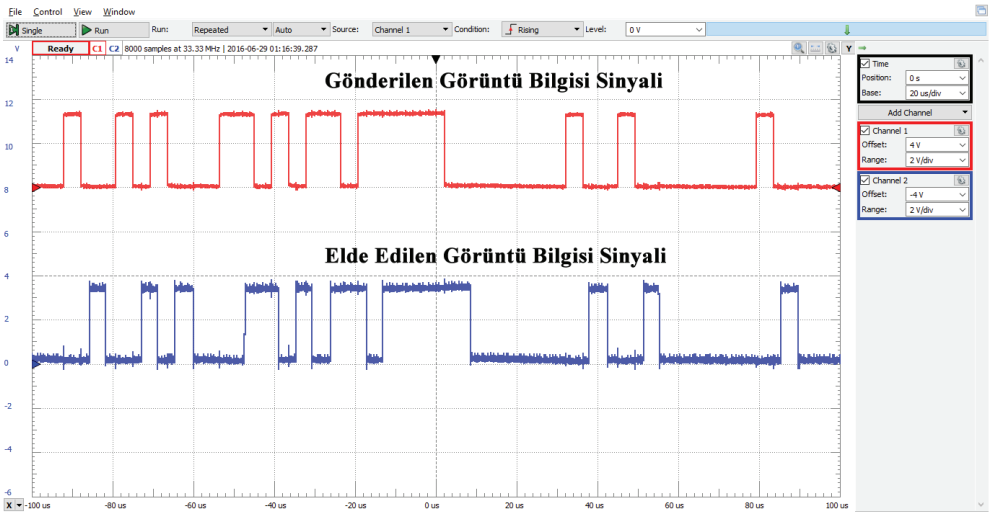
Şekil 9.25. FPGA tabanlı şifreli haberleşme sistemi üzerinden gönderilen görüntü bilgisi dosyasından örnek bir görüntü

Şekil 9.26.'da 1,5 Mbps hızında *Verici* birimi girişine uygulanan görüntü bilgisi sinyali ile *Verici* birim tarafından şifrelenmiş görüntü bilgisi sinyalinin Analog Discovery bilgisayar tabanlı osiloskop cihazı ile yapılan ölçüm görüntüsü verilmiştir (Volt/Div: 2V, Time/Div: 5µs). Şekil 9.26.'da verilen şifrelenmiş bilgi sinyali ayrıca *Verici* birimdeki *KM_Modulasyon* biriminde kaotik sinyal ile toplanarak iletim ortamına verilmektedir. Böylece gönderilen bilgi sinyali ile iletim ortamındaki bilgi sinyali birbirinden çok farklı olmaktadır.



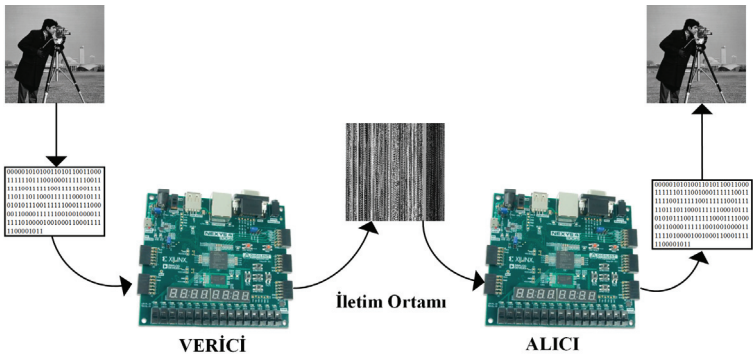
Şekil 9.26. Gönderilen görüntü bilgisi sinyali ile şifrelenmiş görüntü bilgisi sinyali osiloskop görüntüsü

Şekil 9.27.'de *Verici* birim tarafından gönderilen görüntü bilgisi sinyali ile *Alıcı* birimi çıkışından elde edilen sinyal bilgilerinin Analog Discovery bilgisayar tabanlı osiloskop cihazı ile yapılan ölçüm görüntüsü verilmiştir (Volt/Div: 2V, Time/Div: 5µs). Şekilden görüleceği üzere *Verici* biriminden gönderilen görüntü bilgisi *Alıcı* biriminde başarılı bir şekilde elde edilmiştir. Haberleşme sisteminin işlem süresi daha önce belirtildiği gibi 10 saat periyodu olduğu için *Alıcı* biriminden alınan bilgi verici birimden gönderilen bilginin 10xTs = 10x0,64 = 6,4µs. daha gecikmiş halidir.



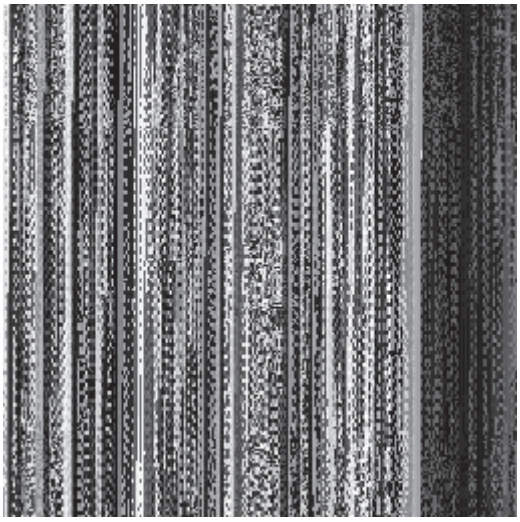
Şekil 9.27. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tarafından gönderilen görüntü bilgisi sinyali ile alıcı birimi tarafından elde edilen görüntü bilgisi sinyali

Tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden görüntü bilgisi gönderimi ve alımı test işleminde *Alıcı* birimden elde edilen bilgilerin dosyaya yazdırılması işlemi için Xilinx Vivado Design Suite 2015.4 programının kendi simülatörü kullanılmıştır. Görüntü bilgisinin ikilik karşılığı *Verici* birimine bilgi sinyali olarak girilmiştir. Şekil 9.28.'de görüntü bilgisinin gönderimi ve alımı işlemi verilmiştir. Şekil 9.29a.'da gönderilen görüntü, Şekil 9.29b.'de *Verici* birim tarafından iletim ortamına aktarılan şifreli ve kaotik maskeleme modülasyonlu görüntü bilgisi, Şekil 9.29c.'de *Alıcı* birim tarafından elde edilen görüntü ve Şekil 9.29d.'de ise gönderilen ve alınan görüntü bilgilerinin birbirlerine göre çizdirilmiş grafiği görülmektedir. Şekil 9.29.'dan görüleceği üzere *Verici* biriminden gönderilen görüntü bilgisi başarılı bir şekilde *Alıcı* birim tarafından tekrar elde edilmiştir.



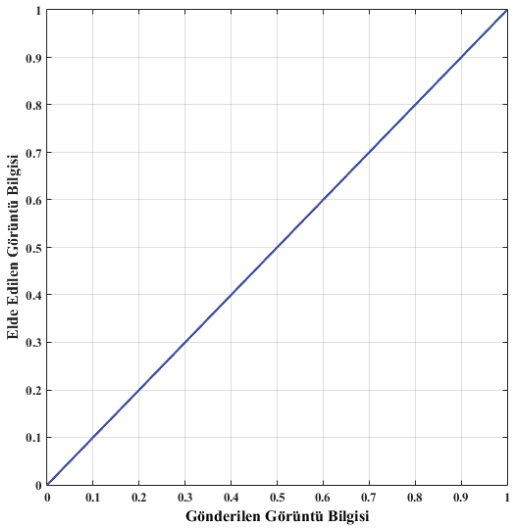
Şekil 9.28. FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden görüntü bilgisi gönderimi ve alımı

(a)



(b)

(c)



(d)

Şekil 9.29. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tarafından gönderilen (a) orijinal görüntü (b) verici birim tarafından iletim ortamına gönderilen görüntü (c) alıcı birim tarafından elde edilen görüntü bilgisi (d) gönderilen ve elde edilen görüntü bilgilerinin birbirlerine göre çizdirilmiş grafiği

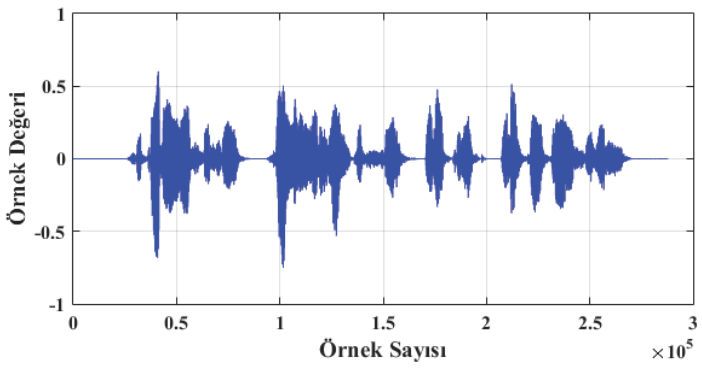
Gönderilen görüntü bilgisi ile *Verici* birim tarafından iletim ortamına aktarılan şifreli ve kaotik maskeleme modülasyonlu görüntü bilgisi arasındaki korelasyon katsayısı Matlab programında hesaplatılmış ve -0,0050 olarak elde edilmiştir. Buradan görüldüğü gibi korelasyon katsayısı değeri sıfıra çok yakın olduğundan gönderilen görüntü bilgisi ile *Verici* birim tarafından iletim ortamına gönderilen görüntü bilgisi arasındaki ilişki çok çok düşüktür. Bu sayede istenmeyen girişimler ile gönderilen görüntü bilgisinin elde edilmesi çok zordur.

9.6. Tasarlanan FPGA Tabanlı Şifreli Kaotik Haberleşme Sistemi İle Ses Bilgisi İletimi

Bu kısımda tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi *Verici* biriminden bir ses bilgisi gönderilmiş ve *Alıcı* birimden alınmıştır. Ses bilgisi olarak aşağıda verilen ifadenin seslendirilmiş halinin mikrofon kaydı kullanılmıştır.

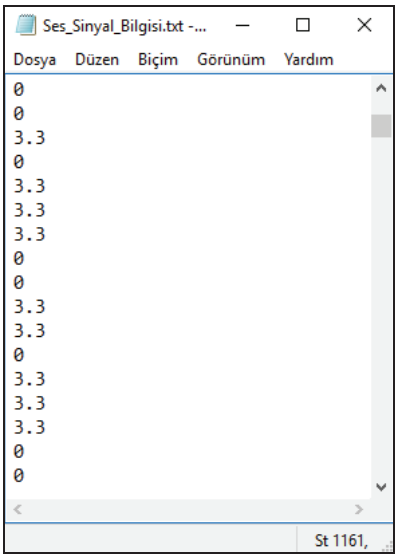
> "Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Bölümü"

Ses kaydı 48000 bps örnekleme oranı ile 6 saniye süre de Matlab programında mikrofon ile kaydedilmiştir. Elde edilen ses kaydı daha sonra 32 bitlik ikilik bilgi formatına çevrilmiştir. Ardından bu 32 bitlik bilgiler tek bitlik veri haline getirilmiştir. Böylece ses bilgisi olarak tek tek gönderilecek bit değerleri 0 ve 1 şeklinde elde edilmiştir. Ses kaydı toplam 48000x6x32= 9216000 adet bitten (0 ve 1) oluşmaktadır. İlgili ses kaydının grafiksel gösterimi Şekil 9.30.'da verilmiştir.



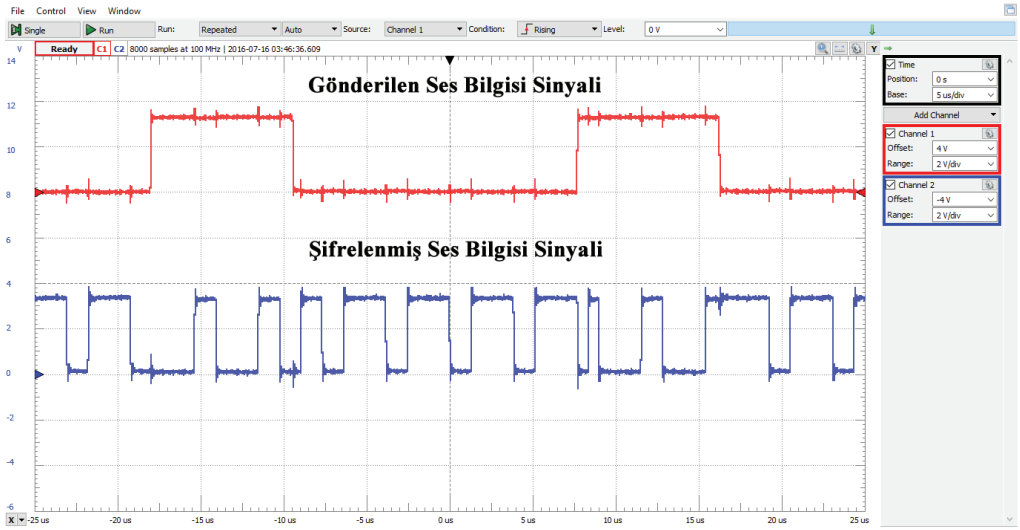
Şekil 9.30. FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden gönderilen ses bilgisi

Ses kaydı bilgisinin gönderimi için ses bilgisinin ikilik karşılığı gerilim karşılıkları olarak dosyaya yazılmış ve Analog Discovery bilgisayar tabanlı osiloskop cihazı ile bu bilgiler 1,5 Mbps hızında sinyal olarak ürettirilmiştir. Lojik-0 değeri 0V ve Lojik-1 değeri ise 3,3V ile temsil edilmiştir. Bu işlem için oluşturulan sinyal bilgisi dosyasından örnek bir görüntü Şekil 9.31.'de verilmiştir. Gönderilen verinin bit periyodu süresi daha önce Denklem 9.4'te açıklandığı gibi $T_{bit} \approx 0{,}67\mu s$.'dir.



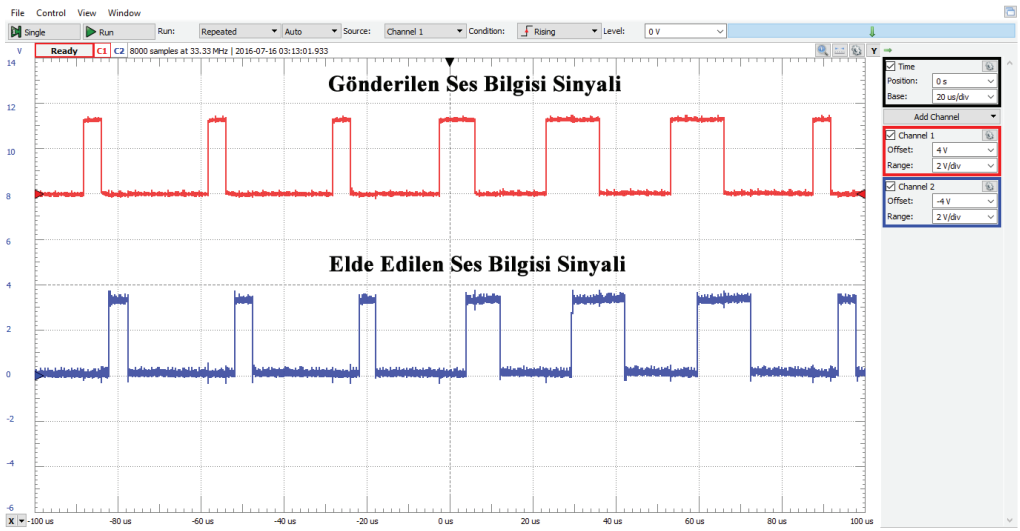
Şekil 9.31. FPGA tabanlı şifreli haberleşme sistemi üzerinden gönderilen ses bilgisi dosyasından örnek bir görüntü

Şekil 9.32.'de 1,5 Mbps hızında *Verici* birimi girişine uygulanan ses bilgisi sinyali ile *Verici* birim tarafından şifrelenmiş ses bilgisi sinyalinin Analog Discovery bilgisayar tabanlı osiloskop cihazı ile yapılan ölçüm görüntüsü verilmiştir (Volt/Div: 2V, Time/Div: 5µs). Şekil 9.32.'de verilen şifrelenmiş bilgi sinyali ayrıca *Verici* birimdeki *KM_Modulasyon* biriminde kaotik sinyal ile toplanarak iletim ortamına verilmektedir. Böylece gönderilen bilgi sinyali ile iletim ortamındaki bilgi sinyali birbirinden çok farklı olmaktadır.



Şekil 9.32. Gönderilen ses bilgisi sinyali ile şifrelenmiş ses bilgisi sinyali osiloskop görüntüsü

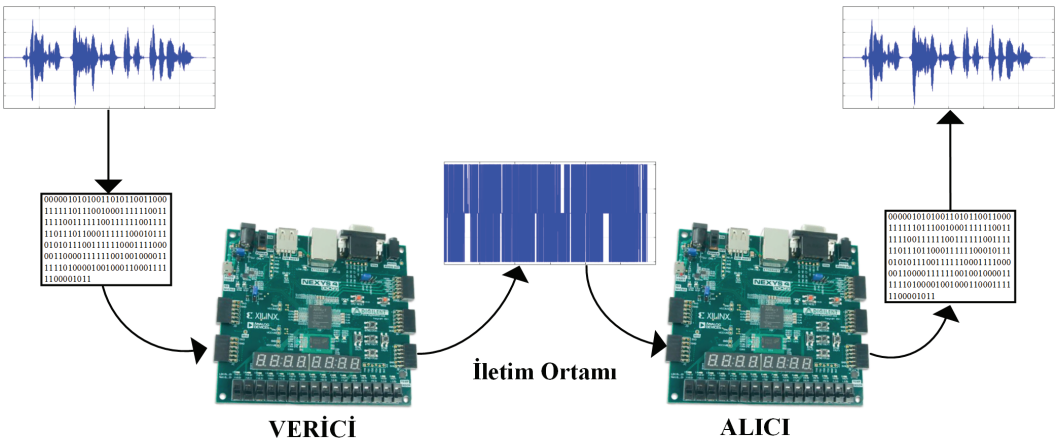
Şekil 9.33.'te *Verici* birim tarafından gönderilen ses bilgisi sinyali ile *Alıcı* birimi çıkışından elde edilen sinyal bilgilerinin Analog Discovery bilgisayar tabanlı osiloskop cihazı ile yapılan ölçüm görüntüsü verilmiştir (Volt/Div: 2V, Time/Div: 20µs). Şekilden görüleceği üzere *Verici* biriminden gönderilen ses bilgisi *Alıcı* biriminde başarılı bir şekilde elde edilmiştir. Haberleşme sisteminin işlem süresi daha önce belirtildiği gibi 10 saat periyodu olduğu için *Alıcı* biriminden alınan bilgi verici birimden gönderilen bilginin 10xTs = 10x0,64 = 6,4µs. daha gecikmiş halidir.



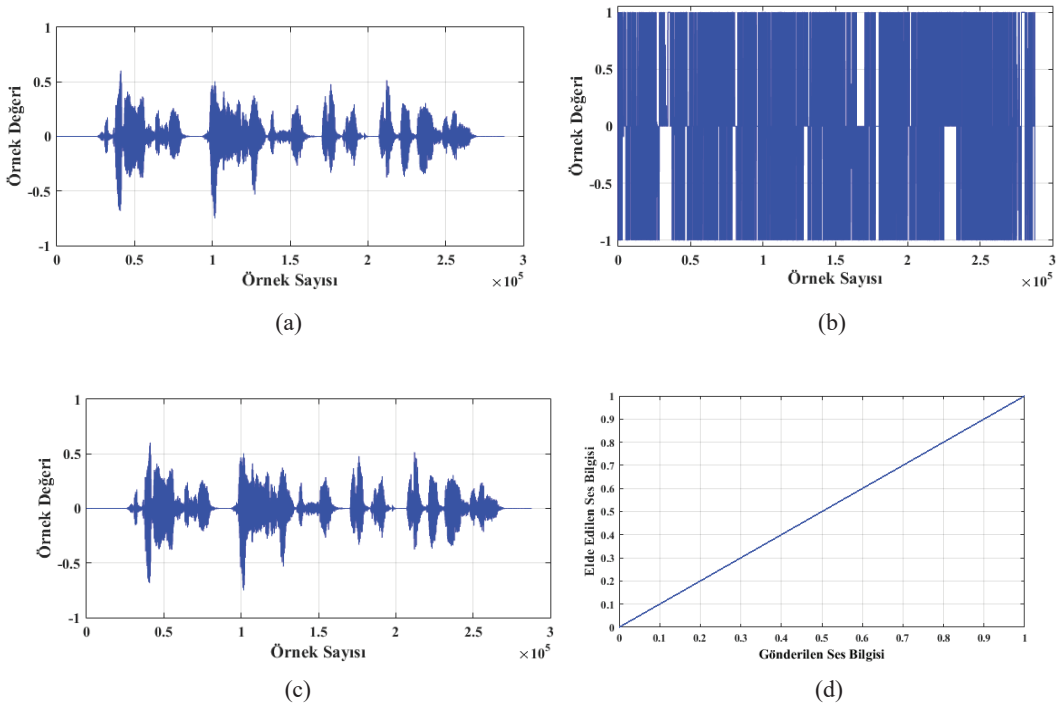
Şekil 9.33. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tarafından gönderilen ses bilgisi sinyali ile alıcı birimi tarafından elde edilen ses bilgisi sinyali

Tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden ses bilgisi gönderimi ve alımı test işleminde *Alıcı* birimden elde edilen bilgilerin dosyaya yazdırılması işlemi için Xilinx Vivado Design Suite 2015.4 programının kendi simülatörü kullanılmıştır. Ses bilgisinin ikilik karşılığı *Verici* birimine bilgi sinyali olarak girilmiştir.

Şekil 9.34.'te ses bilgisinin gönderimi ve alımı işlemi verilmiştir. Şekil 9.35a.'da gönderilen ses bilgisi, Şekil 9.35b.'de *Verici* birim tarafından iletim ortamına aktarılan şifreli ve kaotik maskeleme modülasyonlu ses bilgisi, Şekil 9.35c.'de *Alıcı* birim tarafından elde edilen ses bilgisi ve Şekil 9.35d.'de ise gönderilen ve alınan ses bilgilerinin birbirlerine göre çizdirilmiş grafiği görülmektedir. Şekil 9.35.'ten görüleceği üzere *Verici* biriminden gönderilen ses bilgisi başarılı bir şekilde *Alıcı* birim tarafından tekrar elde edilmiştir.



Şekil 9.34. FPGA tabanlı şifreli kaotik haberleşme sistemi üzerinden ses bilgisi gönderimi ve alımı



Şekil 9.35. FPGA tabanlı şifreli kaotik haberleşme sistemi verici birimi tarafından gönderilen (a) orijinal ses bilgisi (b) verici birim tarafından iletim ortamına gönderilen ses bilgisi (c) alıcı birim tarafından elde edilen ses bilgisi (d) gönderilen ve elde edilen ses bilgilerinin birbirlerine göre çizdirilmiş grafiği

Gönderilen ses bilgisi ile *Verici* birim tarafından iletim ortamına aktarılan şifreli ve kaotik maskeleme modülasyonlu ses bilgisi arasındaki korelasyon katsayısı Matlab programında hesaplatılmış ve -0,0022 olarak elde edilmiştir. Buradan görüldüğü gibi korelasyon katsayısı değeri sıfıra çok yakın olduğundan gönderilen ses bilgisi ile *Verici* birim tarafından iletim ortamına gönderilen ses bilgisi arasındaki ilişki çok çok düşüktür. Bu sayede istenmeyen girişimler ile gönderilen ses bilgisinin elde edilmesi çok zordur.

BÖLÜM 10. SONUÇ VE ÖNERİLER

Bu tez çalışmasında yeni bir üç boyutlu kaotik sistem elde edilerek bu yeni kaotik sistem ile FPGA tabanlı şifreli kaotik haberleşme sisteminin tasarımı yapılmış ve gerçekleştirilmiştir. Bu kapsamda tasarlanan FPGA tabanlı şifreli kaotik haberleşme sistemi ile metin, görüntü ve ses bilgisi gönderimi ile alımı işlemleri gerçekleştirilmiştir. Yapılan çalışmalar, sonuçlar ve öneriler aşağıda verilmiştir.

Tezde ilk olarak konu ile ilgili literatür taramasına yer verilmiş, literatür taramasından sonra ise dinamik sistemler, kaotik sistemler ve kaos analizi hakkında bilgiler verilmiştir. Ardından literatürde kaos tabanlı haberleşme sistemleri için önerilen yöntemler incelenmiştir.

Tezin konu alanı ile ilgili genel bilgilerin verilmesinden sonra, elde edilen yeni üç boyutlu kaotik sistem ile ilgili dinamik analizler (Kararlılık ve denge noktaları, Lyapunov üstelleri, Lyapunov boyutu, Çatallaşma diyagramları), nümerik benzetim ve elektronik devre tasarımı çalışmaları yapılmıştır. Yapılan bu çalışmalarda elde edilen yeni üç boyutlu kaotik sistemin kaos özelliği gösterdiği dinamik analizler ve nümerik benzetim çalışması sonucu ispatlanmıştır. Bu şekilde literatüre kazandırılan yeni üç boyutlu kaotik sistem, kaotik haberleşme uygulamalarının yanında kriptoloji, rastgele sayı üreteçleri, bilgisayar oyunları, resim-grafik üretimi, kontrol sistemleri gibi çeşitli alanlar için kullanılabilir.

Yeni kaotik sistemin işlemsel yükselteç (OPAMP), analog çarpıcı entegresi, direnç ve kondansatör elemanları ile analog elektronik devre tasarımı yapılmış ve elektronik devre tasarımı sonucu ile nümerik analiz sonucu birbirini doğrulamıştır. Böylece yeni kaotik sistemin gerçek ortamlarda çeşitli analog elektronik tasarımlar için kullanabileceği gösterilmiştir.

Yeni kaotik sistem için yapılan analog elektronik devre tasarımı çalışmasından sonra yeni kaotik sistemin Euler algoritması ile nümerik hesaplanması işlemleri Matlab programında yapılmıştır. Ardından yeni kaotik sistem VHDL programlama dili kullanılarak FPGA üzerinde tasarlanmıştır. Tasarımdaki işlem sonuçlarının daha hassas olması için işlemlerdeki sayılarda 32 bit IEEE 754-1985 kayan noktalı sayı formatı (floating point) tercih edilmiştir. Nümerik hesaplama sonucu ile FPGA tasarımının çıkışlarından elde edilen sonuçlar birbirini doğrulamıştır. Böylece yeni kaotik sistemin çeşitli analog ve sayısal gerçek ortam uygulamalarında kullanılabilmesi sağlanmıştır.

Elde edilen yeni kaotik sistemin kaotik haberleşme uygulamalarındaki başarımının testi için Matlab-Simulink programında yeni kaotik sistem kullanılarak, kaotik maskeleme (CM), kaos kaydırmalı anahtarlama (CSK), kaotik açma-kapama anahtarlama (COOK), korelasyon gecikmeli kaydırmalı anahtarlama (CDSK), simetrik kaos kaydırmalı anahtarlama (SCSK) kaos tabanlı sayısal haberleşme yöntemleri ile benzetim çalışmaları yapılmıştır. Yapılan benzetim çalışmalarının 0dB ile 20dB arasındaki E_b/N_0 değerlerinde AWGN kanal modeli altındaki BER performansları karşılaştırılmıştır. Yapılan karşılaştırmada aynı şartlar altında kaotik maskeleme yönteminin BER performansının diğer yöntemlere göre daha iyi olduğu görülmüştür. Bu nedenle tez kapsamında tasarlanan kaotik haberleşme sisteminde de kaotik maskeleme kaotik haberleşme yöntemi tercih edilmiştir.

Çeşitli kaotik haberleşme yöntemlerinin yeni kaotik sistem ile benzetim ortamında tasarımı ve incelemelerinden sonra tasarlanan kaotik haberleşme sisteminin şifreleme biriminde kullanılmak üzere, elde edilen yeni kaotik sistem kullanılarak FPGA tabanlı gerçek rastgele sayı üreteci (GRSÜ) tasarlanmıştır. GRSÜ tasarımda yeni kaotik sistemden elde edilen x, y ve z durum değişkeni değerleri ilk önce 32 bitlik kayan noktalı sayı değerine çevrilmiştir. Ardından 32 bitlik kayan noktalı sayı değerinin sadece LSB biti değeri alınmıştır. GRSÜ'nin son işlem biriminde (post process) EXOR son işlem yöntemi kullanılmıştır. EXOR son işleminde gelen x, y, z durum değişkenlerinin 32 bitlik kayan noktalı sayı değerlerinin LSB bitleri her saat palsinde sırayla (x ve y, sonra z ve x, sonra y ve z gibi) EXOR işlemine tabi

tutulmuştur. GRSÜ ilk önce Matlab-Simulink programında tasarlanarak elde edilen çıkış verilerinin (0 ve 1) rastgelelik analizi yapılmıştır. Rastgelelik analizi için GRSÜ tasarımı çıkışından elde edilen veriler FIBS 140-1 ve NIST 800-22 testlerine tabi tutulmuştur. Matlab-Simulink programında tasarlanan GRSÜ birimi FIBS 140-1 ve NIST 800-22 testlerinden başarılı bir şekilde geçmiştir. Matlab-Simulink programında tasarım çalışması yapılan GRSÜ, daha sonra VHDL programlama dili kullanılarak FPGA üzerinde tasarlanmıştır. FPGA tabanlı GRSÜ birimi çıkışları da FIBS 140-1 ve NIST 800-22 rastgelelik testlerine tabi tutulmuştur. FPGA tabanlı GRSÜ birimi çıkışları FIBS 140-1 ve NIST 800-22 rastgelelik testlerinden başarılı bir şekilde geçmiştir. FPGA tabanlı GRSÜ biriminin maksimum rastgele sayı üretme hızı 6,338MHz olarak elde edilmiştir. Sonuçta şifreleme birimi için gerekli olan rastgele sayılar (0 ve 1) FPGA üzerinde başarılı bir şekilde üretilebilmiştir.

Şifreleme amacıyla GRSÜ birimi tasarımından sonra kaotik maskeleme modülasyon yöntemi kullanılarak şifreli kaotik haberleşme sistemi benzetim çalışması gerçekleştirilmiştir. Benzetim çalışması Matlab-Simulink programında yapılmıştır. Tasarlanan şifreli kaotik haberleşme sistemi, rastgele ikilik (binary) bilgiler kullanılarak AWGN kanal modeli altında 0dB ile 20dB arasındaki E_b/N_0 değerleri altında test edilmiştir. Tasarlanan şifreli kaotik haberleşme sistemi, 9dB E_b/N_0 değerine kadar çok başarılı bir şekilde verici tarafından gönderilen bilgiyi alıcı taraftan elde etmiştir.

Haberleşme sistemi benzetim çalışmasından sonra kaotik maskeleme modülasyon yöntemiyle şifreli kaotik haberleşme sistemi VHDL programlama dili kullanılarak FPGA üzerinde tasarlanmış ve gerçekleştirilmiştir. Tasarımdaki işlemler IEEE 32 bitlik kayan noktalı sayı formatında yapılmıştır. Bu sayede işlem sonuçlarının çok hassas olması sağlanmıştır. Haberleşme sistemi, 100 MHz kristal bağlı iki ayrı FPGA kartı üzerinde metin bilgisi, görüntü bilgisi ve ses bilgisi kullanılarak test edilmiştir. Haberleşme sisteminin test çalışmalarında maksimum veri iletim hızı 1,5 Mbps, veri iletim gecikmesi ise 6,4 μs ve gönderilen bilgi ile iletim ortamına aktarılan bilgi arasındaki korelasyon katsayı değerleri metin, görüntü ve ses bilgisi için sırayla 0,0202, -0,0050, -0,0022 olarak tespit edilmiştir. Test işlemlerinde iletim ortamına

iletilen şifreli ve kaotik modülasyonlu bilginin asıl gönderilen bilgiden çok farklı olduğu ve bu bilgiden asıl bilginin elde edilmesinin çok zor olduğu korelasyon katsayılarından görülmüştür. Sonuç olarak tez çalışmasında gerçekleştirilen FPGA tabanlı şifreli kaotik haberleşme sistemi ile güvenli bir şekilde sayısal veri iletimi sağlanabilmektedir.

Literatürde FPGA tabanlı kaotik haberleşme sistemi üzerine biri 2013 yılında Sadoudi ve arkadaşları tarafından [196] ve diğeri 2015 yılında Tlelo-Cuautle ve arkadaşları tarafından [197] olmak üzere iki adet çalışmaya rastlanılmıştır. Bu iki çalışmada da sadece görüntü bilgisi iletimi yer almaktadır. Tez çalışmasında ise görüntü bilgisinin yanında, metin ve ses bilgileri ile de haberleşme sistemi test edilmiştir.

2013 yılında yapılan çalışmada [196], FPGA işlemleri için 32 bitlik sabit noktalı (fixed-point) sayı formatı kullanılmıştır. 2015 yılında yapılan çalışmada [197] ise 19 bitlik sabit noktalı sayı formatı kullanılmıştır. Tez çalışmasında ise 32 bitlik kayan noktalı (floating-point) sayı formatı kullanılmıştır. Bu sayede tez çalışmasında gerçekleştirilen FPGA tabanlı şifreli kaotik haberleşme sistemindeki işlemler diğer iki çalışmaya göre daha hassas sonuç vermektedir.

Tez çalışmasında kullanılan kaotik sistem daha önce literatürde olmayan yeni elde edilen üç boyutlu bir sistemdir. Diğer iki çalışmada [196, 197] ise literatürde daha önce var olan kaotik sistemler kullanılmıştır. 2013 yılında yapılan çalışmada [196] dört boyutlu hiper-kaotik Lorenz ve 2015 yılında yapılan çalışmada ise [197] üç boyutlu iki ve altı çekerli kaotik sistem kullanılmıştır.

Tez çalışmasında kaotik haberleşme yöntemi olarak kaotik maskeleme modülasyon yöntemi seçilmiştir. Diğer iki çalışmada da [196, 197] kaotik maskeleme ile modülasyon yöntemi kullanılmıştır. 2013 yılındaki çalışma [196] incelendiğinde, bilgi sinyalinin kaotik sinyale eklenerek modülasyonlu tek bilgi hattının alıcı birime gönderildiği ve verici ile alıcı arasında bu modülasyonlu sinyal kullanılarak dinamik geri besleme yöntemiyle senkronizasyon sağlandığı görülmektedir. Bu şekilde bilgi

işaretinin kaotik sinyale eklenerek şifreli bir şekilde gönderilmesi planlanmıştır. Yine 2015 yılındaki çalışma [197] incelendiğinde aynı şekilde bilgi sinyali kaotik sinyale eklenmiş ve alıcı birime gönderilmiştir. Fakat bu çalışmada kullanılan senkronizasyon yöntemi için alıcı birime aynı zamanda üç adet kaotik sistem durum değişkeni çıkışı da gönderilmektedir. Bu durum iletim kanalı sayısını fazlasıyla artırarak haberleşme sistemi için bir dezavantaj oluşturmaktadır.

Tez kapsamında gerçekleştirilen kaotik haberleşme sisteminde iletişim kablolu olarak senkron bir şekilde seri olarak çıkışa aktarılmaktadır. Haberleşme sistemi, iletişim için üç adet çıkışa sahiptir. 2013 yılında yapılan çalışmada [196] ise, bilgi asenkron bir şekilde RS232 iletişim formatında seri olarak gönderilmektedir. Bu çalışmada iletişim kablosuz bir modül ile sağlanmakta ve iletişim için tek çıkış kanalı kullanılmaktadır. Kablosuz modül kullanılması ve iletişim için tek iletişim kanalına ihtiyaç duyması bu özellikler yönünden tez çalışmasındaki sisteme göre avantaj sağlamaktadır. 2015 yılında yapılan çalışmada [197] ise paralel ve kablolu iletişim kullanılmıştır. Ayrıca bu çalışmada iletişim için 19 bitlik dört kanal kullanılmıştır. Paralel iletişim uygulandığından iletişim için toplamda 76 ayrı kanala ihtiyaç vardır. Bu da gerçek ortamda haberleşmenin gerçekleştirilmesini çok zorlaştıran bir dezavantajdır.

Kaotik maskeleme modülasyon yöntemi ile gönderilecek bilgi kaotik sinyal içine gizlenmektedir. Bu durum bilgi sinyalini bir yönüyle şifrelemiş olmaktadır. 2013 [196] ve 2015 [197] yılında yapılan benzer çalışmalarda bilginin şifrelenmesi bu şekilde sağlanmıştır. Tez çalışmasında ise haberleşme sisteminde ayrıca gerçek rastgele sayı üreteci (GRSÜ) birimi tasarlanmış ve GRSÜ çıkışı ile gönderilecek bilgi şifrelendikten sonra kaotik maskeleme modülasyonuna tabi tutulmuştur. Böylece tez çalışmasında gerçekleştirilen FPGA tabanlı şifreli kaotik haberleşme sistemi diğer iki çalışmaya göre iki seviyeli bir şifreleme sistemine sahiptir. Bu durum da haberleşmede bilgi güvenliği için bir avantaj sağlamaktadır.

Tez çalışmasında tasarlanan kaotik haberleşme sistemi VHDL programlama dili ile tasarlanarak gerçek ortamda FPGA üzerinde test edilmiştir. Haberleşme sistemi,

Xilinx Artix-7 ailesi xc7a100tcsg324-1 modeli üzerinde tasarlanmış ve gerçek ortam uygulaması gerçekleştirilmiştir. Tasarımın 100MHz saat sinyali frekansında (f_{Clk}) gerçek ortam testlerinde bilgi iletişim hızı 1,5 Mbps olarak elde edilmiştir. 2013 yılında yapılan çalışma [196], VHDL programlama dili ile tasarlanarak gerçek ortamda FPGA üzerinde test edilmiştir. Haberleşme sistemi, Xilinx Virtex-II Pro ailesi xcv2pff896-7 modeli üzerinde gerçekleştirilmiştir. Sistemin Xbee (Zigbee) kablosuz modül ile gerçek ortam denemesinde 100MHz saat sinyali frekansında (f_{Clk}) bilgi iletişim hızı 250 Kbps olarak elde edilmiştir. Çalışmada Wi-Fi (WLAN) ile bilgi iletişim hızının 54 Mbps olacağı belirtilmiştir. Fakat ilgili çalışmada bu hızda iletişim için bir tasarım yapılarak gerçek ortam uygulamasında denenmemiştir. Sadece teorik olarak bir iletişim hızı değeri verilmiştir. Çalışma incelendiğinde görüntü bilgisinin anlaşılmayacak düzeyde iyi bir şekilde şifrelendiği görülmektedir. 2015 yılında yapılan çalışma [197], Matlab-Simulink programında HDL blokları ile Altera Stratix IV EP4SGX230KF40C2 FPGA modeli ile tasarlanmış ve benzetimi yapılmıştır. Çalışmanın gerçek ortam uygulaması yapılmamıştır. Çalışmada sistemin bilgi iletişim hızı belirtilmemiştir. Çalışmada 16 ve 19. şekiller [197, Fig.16 ve Fig.19) incelendiğinde şifrelenmiş görüntü de orijinal görüntüde olan bazı bölgeler anlaşılabilmektedir. Bu durum da sistemde şifrelemenin çok iyi olmadığını göstermektedir. Ayrıca çalışmada iki ve altı çekerli kaotik sistemler kullanıldığında gönderilen görüntü bilgisi ile verici tarafından iletim ortamına gönderilen bilgi arasındaki korelasyon değerleri sırayla -0,3334 ve -0,0314 olarak belirtilmiştir. Tez çalışmasında ise görüntü testinde korelasyon katsayısı -0,0050 olarak elde edilmiştir. Bu durum da tez çalışmasında sunulan kaotik haberleşme sisteminin daha güvenli bir haberleşme sağladığını göstermektedir. Sonuç olarak tez çalışmasında sunulan kaotik haberleşme sisteminin gerçek ortamdaki bilgi iletim hızı diğer iki çalışmanın gerçek ortamdaki bilgi iletim hızlarından daha büyüktür. Ayrıca tez çalışmasında gerçekleştirilen şifreleme 2015 yılında yapılan çalışmaya [197] göre daha iyi bir şifreleme sağlamaktadır.

Tez çalışmasında tasarlanan ve gerçekleştirilen FPGA tabanlı şifreli kaotik haberleşme sisteminin özelliklerinin literatürde mevcut olan diğer iki çalışmaya [196, 197] göre avantaj ve dezavantajları Tablo 10.1.'de özetlenmiştir.

Tablo 10.1. Tez çalışması ile mevcut olan diğer iki çalışmanın özelliklerinin karşılaştırılması

Özellikler	Tez Çalışması	Kaynak [196]	Kaynak [197]
Sayı formatı	32 bit kayan noktalı	32 bit sabit noktalı	19 bit sabit noktalı
FPGA tasarımı yöntemi	VHDL	VHDL	Matlab-Simulink HDL Blokları
Gerçek ortam uygulaması	Var	Var	Yok
Kullanılan kaotik sistem	Yeni elde edilen üç boyutlu kaotik sistem	Dört boyutlu hiper-kaotik Lorenz sistem	İki ve altı çekerli kaotik sistem
Kaotik modülasyon yöntemi	Kaotik maskeleme	Kaotik maskeleme	Kaotik maskeleme
Bilgi gönderme formatı	Senkron Seri, Kablolu	Asenkron Seri, Kablosuz	Paralel
İletim hattı sayısı	3	1	76
İkinci bir şifreleme	Var	Yok	Yok
Gönderilen ve iletim ortamına aktarılan bilgi arasındaki korelasyon katsayısı değeri	0,0202 (Metin bilgisi) -0,0050 (Görüntü bilgisi) -0,0022 (Ses bilgisi)	Belirtilmemiş	-0,0314 (Görüntü bilgisi)
Şifrelenmiş bilgi kalitesi (İletim hattındaki görüntü resmine göre)	İyi	İyi	Kötü
Bilgi iletim hızı	1,5 Mbps	250 Kbps (Xbee ile)	Belirtilmemiş

Tez çalışmasında kaotik haberleşme yöntemi olarak kaotik maskeleme yöntemi kullanılmıştır. Diğer kaotik haberleşme yöntemleri de benzer şekilde denenebilir.

Kaotik haberleşme sistemi 100 MHz kristal bağlı iki adet FPGA kartı üzerinde test edilmiştir. Daha yüksek bilgi iletim hızlarına ulaşmak için daha yüksek frekans değerine sahip kristaller kullanılabilir.

Kaotik haberleşme sisteminin verici biriminden bilgiler seri olarak çıkmaktadır. Verici ve alıcı birim arasındaki bu seri bilgi transferi için RS232, SPI, I^2C gibi istenen çeşitli iletişim protokolleri kullanılabilir. Benzer şekilde çeşitli kablosuz iletişim modülleri gerçekleştirilen haberleşme sistemine eklenerek veri iletişimi kablosuz olarak sağlanabilir.

KAYNAKLAR

[1] Kurt, E., Kasap, R. Karmaşanın bilimi kaos. Nobel Akademik Yayıncılık Eğitim Danışmanlık Tic. Ltd. Şti., Ankara, 1-10, 2011.

[2] Pehlivan, İ. Yeni kaotik sistemler: elektronik devre gerçeklemeleri, senkronizasyon ve güvenli haberleşme uygulamaları. Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Bölümü, Doktora Tezi, 2010.

[3] Holmes, P. Poincaré, celestial, mechanics, dynamical-systems theory and "chaos". Physics Reports, 193(3):137-163, 1990.

[4] Pol, B. V. D., Mark, J. V. D. Frequency Demultiplication. Nature, 120(3019):363-364, 1927.

[5] Chua, L. O., Kennedy, M. P. Van Der Pol and chaos. IEEE Trans. Circuits Syst., 33(10):974-980, 1986.

[6] Kennedy, M. P. Experimental chaos from autonomous electronic circuit. Phil. Trans. R. Soc. Lond. A, 353(1701):13-32, 1995.

[7] Gleick, J. Kaos. TÜBİTAK, Ankara, 1-91, 1997.

[8] Wyk, M. A., Steeb, W. H. Chaos in electronics. Springer, 1-20, 1997.

[9] Lorenz, E. D. Deterministic nonperiodic flow. J. Atmos. Sci., 20:130-141, 1963.

[10] Li, T. Y., Yorke, J. A. Period three implies chaos. The American Mathematical Monthly, 82(10):985-992, 1975.

[11] Ruelle, D. Rastlantı ve Kaos, TÜBİTAK, Ankara, 64-70, 2001.

[12] Güven, P. Otonom olmayan kaotik sistemlerde rasgele sayı üretiminin incelenmesi. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği, Yüksek Lisans Tezi, 2006.

[13] Ueda, Y., Akamatsu, N. Chaotically transitional phenomena in the forced negative-resistance oscillator. IEEE Transactions on Circuit and Systems, 28(3):217-224, 1981.

[14] Linsay, P. S. Period doubling and chaotic behavior in a driven anharmonic oscillator. Physical Review Letters, 47(19):1349-1352, 1981.

[15] Testa, J., Perez, J., Jeffries, C. Evidence for universal chaotic behavior of a nonlinear oscillator. Physical Review Letters, 48(11):714-717, 1982.

[16] Chua, L. O., Wu, C. W., Huang, A., Zhong, G. Q. A universal circuit for studying and generating chaos-part I: routes to chaos. IEEE Transaction on Circuit and Systems-I: Funamental Theory and Applications, 40(10):732-744, 1993.

[17] Rössler, O. E. An equation for contionus chaos. Phys. Lett. A., 57(5):397-398, 1976.

[18] Chen, G., Ueta, T. Yet another chaotic attractor. Int. J. Bifurcat. Chaos., 9(7):1465-1466, 1999.

[19] Sprott, J. C. Some simple chaotic flows. Phys. Rev. E., 50(2):647-650, 1994.

[20] Kılıç, R., Alçı, M., Tokmakçı, M. Mixed-mode chaotic circuit. Electronics Letters, 36(2):103-104, 2000.

[21] Matsumoto, T., Chua, L. O., Tanaka, S. Simplest chaotic nonautonomous circuit. Physical Review A, 30(2):1155-1158, 1984.

[22] Matsumoto, T., Chua, L. O., Tokunaga, R. Chaos via torus breakdown. IEEE Transactions on Circuit and Systems, 34(3):240-253, 1987.

[23] Ogorzalek, M. J. Order and chaos in a third-order RC ladder network with nonlinear feedback. IEEE Transaction on Circuit and Systems, 36(9):1221-1230, 1989.

[24] Nakagawa, S., Saito, T. An RC OTA hysteresis chaos generator. IEEE Transaction on Circuits and Systems-I: Fundamental Theory and Applications, 43(12):1019-1021, 1996.

[25] Rodriguez-Vazquez, A. B., Huertas, J. L., Chua, L. O. Chaos switched-capacitor circuit. IEEE Transaction on Circuit and Systems, 32(10):1083-1085, 1985.

[26] Horio, Y., Suyama, K. Experimental verification of signal transmission using synchronized SC chaotic neural networks. IEEE Transaction on Circuits and Systems-I: Fundamental Theory and Applications, 42(7):393-395, 1995.

[27] Wang, P. Y. Chaos in phase locked loop. IEEE Transaction on Circuits and Systems, 35(8):987-1003, 1988.

[28] Bradley, E., Straub, D. E. Using chaos to broaden the capture range of a phase-locked loop: experimental verification. IEEE Transaction on Circuits and Systems-I: Fundamental Theory and Applications, 43(11):914-922, 1996.

[29] Chua, L. O., Lin, T. Chaos in digital filters. IEEE Transaction on Circuits and Systems, 35(6):648-658, 1988.

[30] Chua, L. O., Lin, T. Chaos and fractals from 3rd-order digital filters. International Journal of Circuit Theory and Applications, 18(3):241-255, 1990.

[31] Macchi, O., Jaidane-Saidane, M. Adaptive IIR filtering and chaotic dynamics: application to audiofrequency coding. IEEE Transaction on Circuits and Systems, 36(4):591-599, 1989.

[32] Poddar, G., Chakrabarty, K., Banarjee, S. Control of chaos in the boost converter. Electronics Letters, 31(11):841-842, 1995.

[33] Tse, C. K., Fung, S. C., Kwan, M. W. Experimental confirmation of chaos in a current-programmed cuk converter. IEEE Transaction on Circuits and Systems-I: Fundamental Theory and Applications, 43(7):605-608, 1996.

[34] Han, F., Wang, Y., Yu, X., Feng, Y. Experimental confirmation of a new chaotic attractor. Chaos, Solitions and Fractals, 21(1):69-74, 2004.

[35] Liu, Y., Yang, Q. Dynamics of a new Lorenz-like chaotic system. Nonlinear Analysis: Real World Applications, 11(4):2563-2572, 2010.

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

[37] Wei, Z., Yang, Q. Dynamical analysis of a new autonomous 3-D chaotic system only with stable equilibria. Nonlinear Analysis: Real World Applications, 12(1):106-118, 2011.

[38] Pehlivan, İ., Uyaroğlu, Y. A new 3D chaotic system with golden proportion equilibria: analysis and electronic circuit realization. Computers and Electrical Engineering, 38(6): 1777-1784, 2012.

[39] Wang, X., Chen, G. A chaotic system with only one stable equilibrium. Commun Nonlinear Sci Numer Simulat, 17(3):1264-1272, 2012.

[40] Liu, J., Zhang, W. A new three-dimensional chaotic system with wide range of parameters. Optik, 124(22):5528-5532, 2013.

[41] Qi, G., Chen, G. Analysis and circuit implementation of a new 4D chaotic system. Physics Letters A, 352(4-5):386-397, 2004.

[42] Li, C., Tang, Z., Yu, S. A new 4D four-wing hyperchaotic smooth autonomous system and its improved form. Fourth International Workshop on Chaos-Fractals Theories and Applications, Hangzhou, 18-21, 2011.

[43] Dang, H. G. Parameter idenfication of a new hyper-chaotic system. Fifth Conference on Measuring Technology and Mechatronics Automation, Hong Kong, 785-787, 2013.

[44] Zhou, P., Huang, K. A new 4-D non-equilibrium fractional-order chaotic system and its circuit implementation. Commun Nonlinear Sci Numer Simulat, 19(6):2005-2011, 2014.

[45] Ditto, W., Munakata, T. Principles and applications of chaotic systems. Commun. ACM., 38(11):96-102, 1995.

[46] Otte, E., Grebogi, C., Yorke, J. A. Controlling chaos. Phys. Rev. Lett., 64(11):1196-1199, 1990.

[47] Chen, M., Han, Z. Controlling and synchronizing chaotic genesio system via nonlinear feedback control. Chaos, Solitions&Fractals, 17(4):709-716, 2003.

[48] Park, J. H., Kwon, O. M. A novel criterion for delayed feedback control of time-delay chaotic systems. Chaos, Solitions&Fractals, 23(2):495-501, 2005.

[49] Xing-Yuan, W., Ming-Jun, W. A chaotic secure communication scheme based on observer. Commun Nonlinear Sci Numer Simulat, 14(4):1502-1508, 2009.

[50] Xing-Yuan, W., Yong-Feng, G. A switch-modulated method for chaos digital secure communication based on user-defined protocol. Commun Nonlinear Sci Numer Simulat, 15(1):99-104, 2010.

[51] Liu, H., Wang, X., Quanlong, Z. Asynchronous anti-noise hyper chaotic secure communication system based on dynamic delay and state variables switching. Phys. Lett. A, 375(30-31):2828-2835, 2011.

[52] Xing-Yuan, W., Bing, X., Huaguang, Z. A multi-ary number communication system based on hyperchaotic system of 6th-order cellular neural network. Commun Nonlinear Sci Numer Simulat. 15(1):124-133, 2010.

[53] Fujisaka, H., Yamada, T. Stability theory of synchronized motion in coupled-oscillator systems. Progr. Theor. Phys., 69(1):32-37, 1983.

[54] Pecora, L. M., Carroll, T. L. Synchronization in chaotic systems. Phys. Rev. Lett., 64(8):821-825, 1990.

[55] Pecora, L. M., Carroll, T. L. Driving systems with chaotic signals. Phys. Rev. A, 44(4):2374-2383, 1991.

[56] Pecora, L. M., Carroll, T. L., Johnson, G. A., Mar, D. J., Heagy, J. F. Fundamentals of synchronization in chaotic systems: concept and application. Chaos, 7(4):520-543, 1997.

[57] Murali, K., Lakshmanan, M. Drive-response scenario of chaos synchronization in identical nonlinear systems. Phys. Rev. E, 49(6):4882-4885, 1994.

[58] Ott, E., Grebogi, C., Yorke, J. A. Controlling chaos. Phys. Rev. Lett., 64(11):1196-1199, 1990.

[59] Lio, T. L., Adaptive synchronization of two Lorenz systems. Chaos, Solitons&Fractals, 9(9):1555-1561, 1998.

[60] Xu, D. Chaos synchronization between two different Sprott system. Adv. Theor. Appl. Mech., 3(4):195-201, 2010.

[61] Rivera, M., Mekler, G. M., Parmananda, P. Synchronization phenomena for a pair of locally coupled chaotic electrochemical oscillators: a survey. Chaos, 16:37105-1 – 37105-8, 2006.

[62] Blakely, J. N., Pruitt, M. W., Corron, N. J. Time shifts and correlations in synchronized chaos. Chaos, 18:013117-1 – 013117-6, 2008.

[63] Juan, M., Xingyuan, W. Generalized synchronization via nonlinear control. Chaos, 18:023108-1 – 023108-5, 2008.

[64] Galias, Z., Ogorzalek, M. J. Synchronization and cluster formation phenomena in cnn-like structures of coupled nonlinear circuits. J. Circuits Syst. Comput. 12(4):389-397, 2003.

[65] Bonnin, M., Corinto, F., Gilli, M. Phase model reduction and synhronization of periodically forced nonlinear oscillators. J. Circuits Syst. Comput., 19(4):749-762, 2010.

[66] Volos, C. K., Kyprianidis, I. M., Stouboulos, I. N. Various synchronization phenomena in bidirectionally coupled double scroll circuits. Commun Nonlinear Sci Numer Simulat, 16(8):3356-3366, 2011.

[67] Zhang, Q., Lu, J. Chaos synchronization of a new chaotic system via nonlinear control. Chaos, Solitions&Fractals, 37(1):175-179, 2008.

[68] Li, W., Liu, Z., Miao, J. Adaptive synchronization for a unified chaotic system with uncertaintly. Commun Nonlinear Sci Numer Simulat, 15(10):3015-3021, 2010.

[69] Pourmahmood, M., Khanmohammadi, S., Alizadeh, G. Synchronization of two different uncertain chaotic systems with unknown parameters using a robust adaptive sliding mode controller. Commun Nonlinear Sci Numer Simulat, 16(7):2853-2868, 2011.

[70] Yu, W. Synchronization of three dimensional chaotic systems via a single state feedback. Commun Nonlinear Sci Numer Simulat, 16(7):2880-2886, 2011.

[71] Motallebzadeh, F., Motlagh, M., Cherati, Z. Synchronization of different-order chaotic systems: adaptive active vs. optimal control. Commun Nonlinear Sci Numer Simulat, 17(9):3643-3657, 2012.

[72] Qing-Qing, W., Tao, L. Analysis, circuit implementation and synchronization of a new chaotic system. Proceedings of the 33rd Chinese Control Conference, Nanjing-China, 6070-6073, 2014.

[73] Fu-hong, M., Shu-yi, S., Deng-hui, L., Si-cong, W. A new mixed-order chaotic system and its synchronization control. Proceedings of the 33rd Chinese Control Conference, Nanjing-China, 1915-1919, 2014.

[74] Deng, K., Li, J., Yu, S. Dynamics analysis and synchronization of a new chaotic attractor. Optik, 125(13):3071-3075, 2014.

[75] Oppenheim, A. V., Wornell, G. W., Isabella, S. H., Cuomo, K. M. Signal processing in the context of chaotic signals. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), San Francisco, 117-120, 1992.

[76] Cuomo, K. M., Oppenheim, A. V., Strogatz, S. H. Synchronized of Lorenz-based chaotic circuits with applications to communications. IEEE Trans. Circuit Syst. II, Analog Digit Signal Process, 40(10):626-633, 1993.

[77] Cuomo, K. M., Oppenheim, A. V. Circuit implementation of synchronized chaos with applications to communications. Phys. Rev. Lett., 71(1):65-68, 1993.

[78] Kocarev, L., Halle, K. S., Eckert, K., Chuo, L. O., Parlitz, U. Experimental demonstration of secure communications via chaotic synchronization. International Journal of Bifurcation and Chaos, 2(3):709-713, 1992.

[79] Kolumban, G., Kennedy, M. P., Chua, L. O. The role of synchronization in digital communications using chaos-Part II: chaotic modulation and chaotic synchronization. IEEE Transactions on Circuit and Systems-I: Fundemental Theory and Applications, 45(11):1129-1140, 1998.

[80] Parlitz, U., Chua, L. O., Kocarev, L., Halle, K. S., Shang, A. Transmission of digital signals by chaotic synchronization. International Journal of Bifurcation and Chaos, 2(4):973-977, 1992.

[81] Dedieu, H., Kennedy, M. P., Hasler, M. Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits. IEEE Transactions on Circuit and Systems-II: Analog and Digital Signal Processing, 40(10):634-642, 1993.

[82] Kolumban, G., Kennedy, M. P., Kis, G., Jako, Z. FM-DCSK: A novel method for chaotic communications. Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Monterey – CA, 477-480, 1998.

[83] Sushchik, M., Tsimring, L. S., Volkovskii, A. R. Performance analysis of correlation-based communication schemes utilizing chaos. IEEE Transactions on Circuit and Systems-I: Fundemental Theory and Applications, 47(12):1684-1691, 2000.

[84] Tam, W. M., Lau, F. C. M., Tse, C. K. Digital communications with chaos: multiple Access techniques and performance, Elsevier, 1-31, 2006.

[85] Ding, Q., Wang, J. N. Design of frequency-modulated correlation delay shift keying chaotic communication systems. IET Communications, 5(7):901-905, 2011.

[86] Lau, F. C. M., Cheong, K. Y., Tse, C. K. Permutation-based DCSK and multiple-access DCSK systems. IEEE Transactions on Circuit and Systems-I: Fundamental Theory and Applications, 50(6):733-742, 2003.

[87] Tam, W. M., Lau, F. C. M., Tse, C. K. Generalized correlation-delay-shift-keying scheme for noncoherent chaos-based communication systems. IEEE Transactions on Circuit and Systems-I: Regular Papers, 53(3):712-721, 2006.

[88] Yang, H., Jiang, G. P. High-efficiency differential-chaos-shift-keying scheme for chaos-based noncoherent communication. IEEE Transactions on Circuits and Systems II: Express Briefs, 59(5):312-316, 2012.

[89] Duan, J. Y., Jiang, G. P., Yang, H. A new chaotic communications scheme: differential correlation delay shift keying. International Conference on Communications, Circuits and Systems (ICCCAS), Chengdu, 446-449, 2013.

[90] Yang, H., Jiang, G. P. Reference-modulated DCSK: a novel chaotic communication scheme. IEEE Transactions on Circuits and Systems II: Express Briefs, 60(4):232-236, 2013.

[91] Albassam, N. N. A new hybrid DCSK-CDSK scheme for chaos based communications. 5th International Conference on Information and Communication (ICICS), Irbid, 1-5, 2014.

[92] Albassam, N. N., Sumesh, E. P. Enhancing of chaotic on-off keying scheme. IEEE 8th GCC Conference and Exhibition (GCCCE), Muscat, 1-6, 2015.

[93] Duan, J. Y., Jiang, G. P., Yang, H. Reference-adaptive CDSK: an enhanced version of correlation delay shift keying. IEEE Transactions on Circuits and Systems II: Express Briefs, 62(1):90-94, 2015.

[94] Hayes, S., Grebogi, C., Ott, E. Communicating with chaos. Physical Review Letters, 70(20): 3031-3034, 1993.

[95] Yeh, J. P., Wu, K. L. A simple method to synchronize chaotic systems and its application to secure communications. Math. Comput. Model., 47(9-10):894-902, 2008.

[96] Pehlivan, İ., Uyaroğlu, Y. Rikitake attractor and its synchronization application for secure communication systems. J. Appl., Sci., 7(2):232-236, 2007.

[97] Uyaroğlu, Y., Pehlivan, İ. Nonlinear Sprott 94 case H chaotic equation: synchronization and masking communication applications. Comput. Electr. Eng., 36(6):1093-1100, 2010.

[98] Pehlivan, İ., Uyaroğlu, Y. Chaotic oscillator design and realizations of the Rucklidge attractor and its synchronization and masking simulations. Sci. Res. Essays, 5(16):2210-2219, 2010.

[99] Pehlivan, İ., Uyaroğlu, Y. Simplified chaotic diffusionless Lorenz attractor and its applications to secure communication systems. IET Commun., 1(5):1015-1022, 2007.

[100] Sundarapandıan, V., Pehlivan, İ. Analysis, control, synchronization and circuit design of a novel chaotic system. Math. Comput. Model., 55(7-8):1904-1915, 2012.

[101] Acho, L. Expanded Lorenz systems and chaotic secure communication system design. J. Circuits. Syst. Comput., 15(4):607-614, 2006.

[102] Yujun, N., Xingyuan, W., Mingjun, W., Huaguang, Z. A new hyperchaotic system and its circuit implementation. Commun. Nonlinear Sci. Numer. Simulat, 15(11):3518-3524, 2010.

[103] Wang, X., Nian, F., Guo, G. High precision fast projective synchronization in chaotic (hyperchaotic) systems. Phys. Lett. A. 73(20):1754-1761, 2009.

[104] Trejo-Guerra, R., Tlelo-Cuautle, E., Cruz-Hernandez, C., Sanchez-Lopez, C. Chaotic communication system using Chua's oscillators realized with CCII+s. Int. J. Bifurcat. Chaos. 19(12):4217-4226, 2009.

[105] Gamez-Guzman, L., Cruz-Hernandez, C., Lopez-Gutıerrez, R. M., Garcia-Guerrero, E. E. Synchronization of multi-scroll chaos generators: application to private communication. Rev. Mex. Fis. 54(4):299-305, 2008.

[106] Trejo-Guerra, R., Tlelo-Cuautle, E., Jimenez-Fuentes, J. M., Sanchez-Lopez, C., Munoz-Pacheco, J. M., Espinosa-Flores-Verdad, G., Rocha-Perez, J. M. Integrated circuit generating 3- and 5-scroll attractors. Commun Nonlinear Sci Numer Simulat, 17(11):4328-4335, 2012.

[107] Trejo-Guerra, R., Tlelo-Cuautle, E., Sanchez-Lopez, C., Munoz-Pacheco, J. M., Cruz-Hernandez, C. Realization of multiscroll chaotic attractors by using current-feedback operational amplifiers. Rev. Mex. Fis., 56(4):268-274, 2010.

[108] Sanchez-Lopez, C., Trejo-Guerra, R., Munoz-Pacheco, J. M., Tlelo-Cuautle, E. N-scroll chaotic attractors from saturated function series employing CCII+s. Nonlinear Dynam., 61(1):331-341, 2010.

[109] Munoz-Pacheco, J. M., Tlelo-Cuautle, E. Automatic synthesis of 2D-n-scrolls chaotic systems by behavioral modeling. J. Appl. Res. Technol. 7(1):5-14, 2009.

[110] Cong, L., Xiaofu, W. Design and realization of an FPGA-based generator for chaotic frequency hopping sequences. IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, 48(5):521-532, 2001.

[111] Guang-Yi, W., Xu-Lei, B., Zhong-Lin, W. Design and FPGA implementation of a new hyperchaotic system. Chinese Physics B, 17(10):3596-3602, 2008.

[112] Sadoudi, S., Azzaz, M. S., Djeddou, M., Benssalah, M. An FPGA real-time implementation of the Chen's chaotic system for securing chaotic communications. International Journal of Nonlinear Science, 7(4):467-474, 2009.

[113] Xue, H., Fan, X. Design and implementation of a new chaotic system. International Conference on Computer, Mechatronics, Control and Electronic Engineering(CMCE), Changchun, 552-555, 2010.

[114] Aseeri, M. A. S. Chaotic model (Rössler) using filed programmble gate array (FPGA). 4th International Design and Test Workshop (IDT), Riyadh, 1-4, 2009.

[115] Micco, L., Larrondo, H. A. FPGA implementation of a chaotic oscillator using RK4 method. VII Southern Conference on Programmable Logic (SPL), Cordoba, 185-190, 2011.

[116] Bereber, S. M., Wang, C., Wei, K. K. Design of a CDMA system in FPGA technology. IEEE 65th Vehicular Technology Conference, Dublin, 3061-3065, 2007.

[117] Azzaz, M. S., Tanougast, C., Sadoudi, S., Bouridane, A., Dandache, A. An FPGA implementation of a feed-back chaotic synchronization for secure communications. 7th International Communication Systems Networks and Digital Signal Processing (CSNDSP), Newcastle, 239-243, 2010.

[118] Premalatha, L., Vanaja, R. Implementation of FPGA based digital controller for controlling chaos in DC/DC converters. Journal of Computer Science and Control Systems, 3(1):109-114, 2010.

[119] Yau, H. T., Pu, Y. C., Li, S. C. An FPGA-based PID controller design for chaos synchronization by evolutionary programming. Discrete Dynamics in Nature and Society, 2011:1-11, 2011.

[120] Xue, H., Li, W. Implementation of chaos synchronization on FPGA. Advances in Information Sciences and Service Sciences, 4(6):42-51, 2012.

[121] Caponetto, R., Mauro, A., Fortuna, L., Frasca, M. Field programmable analog array to implement a programmable Chua's circuit. International Journal of Bifurcation and Chaos, 15(5):1829-1836, 2005.

[122] Kılıç, R., Dalkıran, F. Y. Reconfigurable implementations of Chua's circuit. International Journal of Bifurcation and Chaos, 19(4):1339-1350, 2009.

[123] Kılıç, R., Dalkıran, F. Y. Programmable design and implementation of a chaotic system utilizing multiple nonlinear functions. Turk J Elec Eng&Comp Sci, 18(4):647-655, 2010.

[124] Rahma, F., Ali, R. S., Fortuna, L. Analog programmable electronic circuit-based chaotic Lorenz system. Basrah Journal for Enginnering Sciences, 14(1):39-47, 2014.

[125] Ott, E. Chaos in dynamical systems. Cambridge University Press, Canada, 6-20, 1993.

[126] Togur, C. Otonom ve otonom olmayan iki kaotik osilatör tasarımı. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Bölümü, Yüksek Lisans Tezi, 2007.

[127] Kara, R. Nonlineer dinamik sistemlerde kaos, dallanma ve fraktaller. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Bölümü, Yüksek Lisans Tezi, 2006.

[128] Kılıç, R. Elektronik devrelerdeki kaos olayının bilgisayar simülasyonları ile incelenmesi. Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik Bölümü, Yüksek Lisans Tezi, 1996.

[129] Jost, J. Dynamical systems: examples of complex behaviour. Springer, New York, 1-6, 2005.

[130] Kia, B. Chaos computing from theory to application. Arizona State University, Doktora Tezi, 2011.

[131] Fraga, L. G., Tlelo-Cuautle, E., Carbajal-Gomez, V. H., Munoz-Pacheco, J. M. On maximizing positive Lyapunov exponents in chaotic oscillator with heuristics. Rev. Mex. Fis., 58(3):274-281, 2012.

[132] Giannakopoulos, K., Deliyannis, T., Hadjidemetriou, J. Means for detecting chaos and hyperchaos in nonlinear electronic circuits. 14th International Conference on Digital Signal Processing (DSP), Greece, 951-954, 2002.

[133] Özer, A. B. Elektriksel sürücü sistemlerinde doğrusal olmayan olguların kaotik analizi ve yumuşak hesaplama yöntemleri ile denetimi. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Bölümü, Doktora Tezi, 2005.

[134] Özkaynak, F. Doğrusal olmayan sistemlerde Lyapunov üstellerini hesaplayan yazılımının gerçekleştirilmesi. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Bölümü, Yüksek Lisans Tezi, 2007.

[135] Stavroulakis, P. Chaos applications in telecommunications. CRC Press, USA, 125-169, 2006.

[136] Chau, K. T., Wang, Z. Chaos in electric drive systems: analysis, control and application. John Wiley & Sons (Asia) Pte Ltd, Singapore, 3-44, 2011.

[137] Kinsner, W. Characterizing chaos through Lyapunov metrics. IEEE Transactions on Systems, MAN, And Cybernetics-Part C: Applications and Reviews, 36(2):141-151, 2006.

[138] Wolf, A., Swift, J. B., Swinney, H. L., Vastano, J. A. Determining Lyapunov exponents from a time series. Physica, D, 16:285-317, 1985.

[139] Bolotin, Y., Tur, A., Yanovsky, V. Chaos: concepts, control and construvtive use. Springer, Berlin, 19-34, 2009.

[140] Sandri, M. Numerical calculations of Lyapunov exponents. The Mathematica Journal, 6(3):78-84, 1996.

[141] http://staff.www.ltu.se/~larserik/applmath/chap9en/part7.html, Erişim Tarihi: 18.01.2016.

[142] https://en.wikipedia.org/wiki/Limit_cycle, Erişim Tarihi: 18.01.2016.

[143] https://en.wikipedia.org/wiki/Attractor, Erişim Tarihi: 18.01.2016.

[144] Çiçek, S., Ferikoğlu, A., Pehlivan, İ. Simulation and circuit implementation of Sprott case h chaotic system and its synchronization application for secure communication systems. Journal of Circuits, Systems and Computers, 22(4):1350022-1 – 1350022-15, 2013.

[145] Sardar, Z., Abrams, I. Kaos (Guliyeva, D., Çev.). NTV Yayınları, İstanbul, 28-39, 2011.

[146] Kındıkoğlu, S. Çok-boyutlu kaotik sistemlerin senkronizasyonu ve haberleşmede kullanılması. Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Fizik Bölümü, Doktora Tezi, 1999.

[147] Koçak, K. Kaotik bir davranış kriteri olarak fraktal boyut değişimi ve dinamik sistemlere uygulanması. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 1996.

[148] Strogatz, S. H. Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering. Perseus Book Publishing, Canada, 398-416, 1994.

[149] http://www.wahl.org/fe/HTML_version/link/FE3W/c3.htm, Erişim Tarihi: 19.01.2016.

[150] http://math.stackexchange.com/questions/279267/properties-of-the mandelbrot-set-accessible-without-knowledge-of-topology, Erişim Tarihi: 19.01.2016.

[151] http://www.wahl.org/fe/HTML_version/link/FE2W/c2.htm, Erişim Tarihi: 19.01.2016.

[152] Kaplan, J. L, Yorke, J. A. Functional differantial equations and approximation of fixed points. Peitgen, H.O., Walther H.O. (Eds.). Springer-Verlag, Bonn, 204-227, 1979.

[153] Liu, C., Liu, T., Ling, L., Liu, K. A new chaotic attractor. Chaos, Solitons and Fractals, 22(5):1031-1038, 2004.

[154] Yang, T. A survey of chaotic secure communication systems. International Journal of Computational Cognition, 2(2):81-130, 2004.

[155] Li, S., Alvarez, G., Li, Z., Halang, W. A. Analog chaos-based secure communications and cryptanalysis: a brief survey. 3rd International Conference "Physics and Control" (PhysCon2007), Potsdam-Germany, 2007.

[156] Abel, A., Schwarz, W. Chaos communications-principles, schemes and system analysis. Proceedings of the IEEE, 90(5):691-710, 2002.

[157] Illing, L. Digital communication using chaos and nonlinear dynamics. Nonlinear Analysis: Theory, Methods&Applications, 71(12):e2958-e2964, 2009.

[158] Şevik, S. Kaotik haberleşme sistemlerinden diferansiyel kaos kaydırmalı anahtarlama metodu ve performans analizi. Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği, Yüksek Lisans Tezi, 2003.

[159] Kolumban, G., Kennedy, M. P., Chua, L. O. The role of synchronization in digital communications using chaos-Part I: fundamentals of digital communications. IEEE Transactions on Circuit and Systems-I: Fundamental Theory and Applications, 44(10):927-936, 1997.

[160] Banerjee, S., Mitra, M., Rondoni, L. (Eds.). Applications of chaos and nonlinear Dynamics in engineering-vol.1. Springer, Berlin-Germany, 203-225, 2011.

[161] Kennedy, M. P., Kolumban, G. Digital communications using chaos. Signal Processing, 80(7):1307-1320, 2000.

[162] Yardım, F. E., Afacan, E. Lorenz-tabanlı diferansiyel kaos kaydırmalı anahtarlama (dcsk) modeli kullanılarak kaotik bir haberleşme sisteminin simülasyonu. Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 25(1):101-110, 2010.

[163] Oğraş, H. Kaos Tabanlı Sayısal Haberleşme Sistemlerinin Benzetimi İçin Bir Grafik Kullanıcı Arabirim Tasarımı. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği, Yüksek Lisans Tezi, 2010.

[164] Lynnyk, V. Chaos-based communication systems. Czech Technical University, Elektrik Mühendisliği Fakültesi, Kontrol Mühendisliği Bölümü, Elektrik Mühendisliği ve Bilgi Teknolojisi Programı, Doktora Tezi, 2010.

[165] Kolumban, G., Vizvari, B., Schwarz, W., Abel, A. Differential chaos shift keying: a robust coding for chaos communication. Proc. 4th International Workshop on Nonlinear Dynamics of Electronics Systems (NDES'98), Budapest-Hungary, 41-51, 1996.

[166] Karaboğa, N. Sayısal yöntemler ve matlab uygulamaları. Nobel Akademik Yayıncılık, Ankara, 348-349, 2012.

[167] Karagöz, İ. Sayısal analiz ve mühendislik uygulamaları. Nobel Akademik Yayıncılık, Ankara, 348-351, 2011.

[168] Çağal, B. Sayısal analiz. Birsen Yayınevi, İstanbul, 437-439, 1989.

[169] Aydın, İ. Gerçek zamanlı durum izleme ve arıza teşhisi için bağışık akıllı hesaplama tekniklerinin geliştirilmesi. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği, Doktora Tezi, 2011.

[170] Sarıtaş, E., Karataş, S. Her yönüyle fpga ve vhdl. Palme Yayıncılık, Ankara, 3-26, 2013.

[171] Dikmeşe, Ş. Kablosuz haberleşme sistemlerinde fpga uygulaması. Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği, Yüksek Lisans Tezi, 2007.

[172] Güneroğlu, A. Fotovoltaik sistemlerde fpga kullanımı. Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik Eğitimi, Yüksek Lisans Tezi, 2008.

[173] Özmen, A. Hızlı hesaplama için yüksek performanslı paralel önek toplayıcıların tasarımı. İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği, Yüksek Lisans Tezi, 2011.

[174] Doğan, A. Y. AES algoritmasının fpga üzerinde düşük güçlü tasarımı. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Bölümü, Elektronik Mühendisliği Programı, Yüksek Lisans Tezi, 2008.

[175] Kleitz, W. Digital electronics a practical approach with vhdl. Pearson Education Inc., USA, 112-121, 2012.

[176] IEEE. IEEE standart for binary floating-point arithmetic. The Institute of Electrical and Electronics Engineer, New York-USA, 1-8, 1985.

[177] Xilinx. UG474 User Guide:7 series FPGAs configurable logic block user guide. Xilinx, 9-23, 2014.

[178] Xilinx. DS180: 7 series FPGAs overview user guide. Xilinx, 1-6, 2015.

[179] Xilinx. UG479: 7 series DSP48E1 slice user guide. Xilinx, 9-11, 2014.

[180] Demirkol, A. Ş. Kaotik osilatör girişli ADC tabanlı rastgele sayı üreteci. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Bölümü, Elektronik Mühendisliği Programı, Yüksek Lisans Tezi, 2007.

[181] Özdemir, K. Sürekli-zamanlı kaos ile rastgele sayı üreteci tasarımı. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Bölümü, Elektronik Mühendisliği Programı, Yüksek Lisans Tezi, 2008.

[182] Tavas, V. Tümleştirmeye uygun rastgele sayı üreteçleri. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Bölümü, Elektronik Mühendisliği Programı, Doktora Tezi, 2011.

[183] Avaroğlu, E. Donanım tabanlı rasgele sayı üretecinin gerçekleştirilmesi. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Bölümü, Telekomünikasyon Programı, Doktora Tezi, 2014.

[184] Erat, M. FPGA tabanlı, PCI arayüzlü, gerçek zamanlı rasgele sayı üreteci test sistemi tasarımı ve uygulamaları. Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Bölümü, Doktora Tezi, 2008.

[185] Yıldırım, S. A true random number generator in FPGA for cryptographic applications. Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik ve Elektronik Mühendisliği, Yüksek Lisans Tezi, 2012.

[186] Ateş, E. Ö. Chaotic oscillator based random bit generator. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 2005.

[187] Bayam, F. Chaotic oscillator based random number generator. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 2005.

[188] NIST. Federal information processing standard (FIPS) 140-1, NIST, USA, 43-45, 2015.

[189] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., Vo, S. A statistical test suite for random and pseudorandom number generators for cryptographic applications, NIST, USA, 2.1 – 2.40, 2010.

[190] Koyuncu, İ. Kriptolojik uygulamalar için FPGA tabanlı yeni kaotik osilatörlerin ve gerçek rasgele sayı üreteçlerinin tasarımı ve gerçeklenmesi. Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Bölümü, Doktora Tezi, 2014.

[191] Digilent. Nexys4 DDR FPGA board reference. Digilent Inc., 1-2, 2014.

[192] Digilent. Analog Discovery technical reference manual. Digilent Inc., 1-2, 2015.

[193] Spiegel M. R., Schiller, J., Srinivasan, R. A. Schaum's outline series: Probability and statistics. McGraw-Hill, USA, 81-82, 2009.

[194] Menzel, K. 14:30 Introduction to statistical methods in economics-Lecture notes 13. MIT OpenCourseWare, USA, 1-3, 2009.

[195] Özdemir, T. İstatistiksel kalite kontrol. A.Ü.F.F. Döner Sermaye İşletmesi Yayınları, Ankara, 78-79, 2000.

[196] Sadoudi, S., Tanougast, C., Azzaz, M. S., Dandache, A. Design and FPGA implementation of a wireless hyperchaotic communication system for secure real-time image transmission. EURASIP Journal on Image and Video Processing, 2013:43, 2013.

[197] Tlelo-Cuautle, E., Carbajal-Gomez, V. H., Obeso-Rodelo, P. J., Rangel-Magdaleno, J. J., Núñez-Pérez, J. C. FPGA realization of a chaotic communication system applied to image processing. Nonlinear Dynamics, 82(4):1879-1892, 2015.

ÖZGEÇMİŞ

Serdar ÇİÇEK, 17.11.1981'de Mersin'de doğdu. İlk, orta ve lise eğitimini Mersin'de tamamladı. 1998 yılında Mersin Teknik Lisesi Elektronik bölümünden mezun oldu. 1999 yılında başladığı Gazi Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü Elektronik Öğretmenliği programını 2003 yılında bitirdi. 2003 yılında MEB Kocaeli Körfez Mesleki ve Teknik Anadolu lisesinde Elektronik/Telekomünikasyon öğretmeni olarak çalışmaya başladı. Aynı yıl başladığı Gazi Üniversitesi Fen Bilimleri Enstitüsü Elektronik-Bilgisayar Eğitimi bölümündeki yüksek lisans eğitimini 2006 yılında bitirdi. 2010 yılında Nevşehir Üniversitesinde öğretim görevlisi olarak çalışmaya başladı. Aynı yıl Sakarya Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği bölümünde doktora eğitimine başladı. Doktora eğitimi esnasında 2012 yılında başladığı Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar ve Bilişim Mühendisliği bölümdeki ikinci yüksek lisans eğitimini 2014 yılında bitirdi. Halen Nevşehir Hacı Bektaş Veli Üniversitesinde çalışmaya devam etmektedir.