

**BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
TEZLİ YÜKSEK LİSANS PROGRAMI**

**FPGA TABANLI UZUN KISA-SÜRELİ BELLEK YAPAY SİNİR AĞI
İLE DARBESEL SİNYAL TESPİTİ**

HAZIRLAYAN

ERDOĞAN BERKAY TEKİNCAN

YÜKSEK LİSANS TEZİ

ANKARA – 2022

**BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
TEZLİ YÜKSEK LİSANS PROGRAMI**

**FPGA TABANLI UZUN KISA-SÜRELİ BELLEK YAPAY SİNİR AĞI
İLE DARBESEL SİNYAL TESPİTİ**

HAZIRLAYAN

ERDOĞAN BERKAY TEKİNCAN

YÜKSEK LİSANS TEZİ

TEZ DANIŞMANLARI

DR. ÖĞR. ÜYESİ TÜLİN ERÇELEBİ AYYILDIZ

DR. NİZAM AYYILDIZ

ANKARA – 2022

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı çerçevesinde Erdoğan Berkay TEKİNCAN tarafından hazırlanan bu çalışma, aşağıdaki jüri tarafından Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi: 05/01/2022

Tez Adı: FPGA TABANLI UZUN KISA-SÜRELİ BELLEK YAPAY SİNİR AĞI İLE DARBESSEL SİNYAL TESPİTİ

Tez Jüri Üyeleri (Unvanı, Adı- Soyadı, Kurumu)

İmza

Prof. Dr. Hamit ERDEM – Başkent Üniversitesi

.....

Dr. Öğr. Üyesi Tülin ERÇELEBİ AYYILDIZ – Başkent Üniversitesi

.....

Doç. Dr. Mehmet Feyzi AKŞAHİN – Gazi Üniversitesi

.....

ONAY

Prof. Dr. Ömer Faruk ELALDI

Fen Bilimleri Enstitüsü Müdürü

Tarih: ... / ... /

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
YÜKSEK LİSANS TEZ ÇALIŞMASI ORJİNALLİK RAPORU

Tarih: 10 / 01 / 2022

Öğrencinin Adı, Soyadı: Erdoğan Berkay TEKİNCAN

Öğrencinin Numarası: 21820176

Anabilim Dalı: Bilgisayar Mühendisliği Anabilim Dalı

Programı: Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı

Danışmanın Unvanı/Adı, Soyadı: Dr. Öğr. Üyesi Tülin ERÇELEBİ AYYILDIZ, Dr. Nizam AYYILDIZ

Tez Başlığı: FPGA TABANLI UZUN KISA-SÜRELİ BELLEK YAPAY SİNİR AĞI İLE DARBESSEL SİNYAL TESPİTİ

Yukarıda başlığı belirtilen Yüksek Lisans tez çalışmamın; Giriş, Literatür Taraması, Tasarım Çalışmaları, Sonuç ve Öneriler bölümlerinden oluşan, toplam 68 sayfalık kısmına ilişkin, 10 / 01 / 2022 tarihinde tez danışmanım tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 4'tür. Uygulanan filtrelemeler:

1. Kaynakça hariç
2. Alıntılar hariç
3. Beş (5) kelimeden daha az örtüşme içeren metin kısımları hariç

“Başkent Üniversitesi Enstitüleri Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Usul ve Esasları” inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Öğrenci İmzası:

ONAY

Tarih: 10/01/2022

Tez Danışmanı

Dr. Öğr. Üyesi Tülin ERÇELEBİ AYYILDIZ

Bu tezi her zaman ve her koşulda bana destek veren sevgili eşim Gizem BÜGET TEKİNCAN'a ithaf ediyorum.

Erdoğan Berkay TEKİNCAN

Ankara - 2022



TEŞEKKÜR

Yüksek lisans tez çalışmalarında bana her türlü desteği veren ve tecrübelerinden faydalanma imkânı sunan değerli hocalarım Dr. Öğr. Üyesi Tülin ERÇELEBİ AYYILDIZ'a ve Dr. Nizam AYYILDIZ'a en içten şükranlarımı ve teşekkürlerimi sunarım.

Yüksek lisans sürecince ders aşamasından başlayarak tez aşamasının sonuna kadar her konuda olduğu gibi bu süreçte de bana her türlü manevi desteğini eksik etmeyen eşim Gizem BÜGET TEKİNCAN'a, babam Cüneyt TEKİNCAN'a, annem Dilek TEKİNCAN'a ve kardeşim Beray TEKİNCAN'a teşekkürlerimi sunarım.



ÖZET

Erdoğan Berkay TEKİNCAN

FPGA TABANLI UZUN KISA-SÜRELİ BELLEK YAPAY SİNİR AĞI İLE DARBESSEL SİNYAL TESPİTİ

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

2022

Bu tez çalışmasında, elektronik harp destek sistemlerinde de önemli bir rolü olan darbesel sinyal tespit uygulamaları için FPGA tabanlı bir Uzun Kısa-Sürelî Bellek yapay sinir ağı önerilmiştir. Sinyal alması tarafından toplanan darbesel sinyal örnekleri üzerinde herhangi bir işlem yapmadan sadece ham veriler kullanılarak düşük sinyal gürültü oranı (SGO) seviyelerinde yaygın sinyal tespit yöntemi olan Sabit Yanlış Alarm Oranı (SYAO) (Constant False Alarm Rate) yöntemi ile tespit edilemeyen darbesel sinyallerin önerilen bu yöntem ile tespit edilmesi hedeflenmiştir. Daha önce gerçekleştirilen benzer çalışmalar Uzun Kısa-Sürelî Bellek sinir ağlarının düşük SGO seviyelerinde yüksek başarı oranı ile sinyal tespiti yapabildiğini göstermektedir. Benzer çalışmalara ek olarak bu tez çalışmasında FPGA tabanlı bir Uzun Kısa-Sürelî Bellek sinir ağı çalışması yapılmıştır ve -5 dB SGO seviyesinde %94 doğruluk değeri ile darbesel sinyal tespiti yapılabildiği FPGA benzetim sonuçları ile gözlemlenmiştir. Bu tez çalışmasında, FPGA tabanlı Uzun Kısa-Sürelî Bellek sinir ağının eğitilmesi ve test edilmesi için gerekli veri setlerinin oluşturulması amacıyla MATLAB/Simulink modelleri tasarlanmıştır. Bu modeller ile ayrıca SYAO yönteminin test veri setleri de elde edilmiştir. Bu tez çalışmasında, FPGA tabanlı Uzun Kısa-Sürelî Bellek sinir ağının sentez işlemi sonrası elde edilen performans ölçütleri, SYAO yönteminin FPGA’de uygulanması sonrası elde edilen performans ölçütleri ile karşılaştırılmıştır. Elde edilen sonuçlar, bu tez çalışmasında önerilen FPGA tabanlı Uzun Kısa-Sürelî Bellek sinir ağının bir donanım (FPGA) üzerinde gerçekleştirilerek klasik eşik değeri yönteminin (SYAO yöntemi) başarılı sonuçlar elde edemediği düşük SGO seviyelerinde sinyal tespiti gerçekleştirebildiğini ancak gerçek zamanlı bir elektronik harp sistemi radar alması gereksinimlerini karşılayamadığını göstermektedir.

ANAHTAR KELİMELELER: FPGA, Uzun Kısa-Sürelî Bellek, Yapay Sinir Ağı, Düşük SGO Seviyelerinde Darbesel Sinyal Tespiti, Sabit Yanlış Alarm Oranı

ABSTRACT

Erdoğan Berkay TEKİNCAN

FPGA-BASED LONG SHORT-TERM MEMORY ARTIFICIAL NEURAL NETWORK FOR PULSE SIGNAL DETECTION

Başkent University Institute of Science

Department of Computer Engineering

2022

In this thesis, an FPGA-based Long Short-Term Memory (LSTM) Artificial Neural Network (ANN) has been proposed for pulsed signal detection applications, which also has an important role in Electronic Warfare Support Systems. It is aimed to detect pulsed signals collected by the signal receiver at low signal-to-noise ratio (SNR) levels with this proposed FPGA-based LSTM, using only raw data without any processing on the pulsed signal samples that cannot be detected by the widely used Constant False Alarm Rate (CFAR) method. Previous studies show that LSTM neural networks can detect pulsed signals with a high success rate at low SNR levels. In addition to these studies, an FPGA-based LSTM neural network is implemented in this thesis. According to simulation results, it is observed that pulsed signal detection could be performed with 94% accuracy at -5 dB SNR level. In this thesis, MATLAB/Simulink models were designed to generate the datasets required for training and testing the FPGA-based Long Short-Term Memory neural network. With these models, test datasets of the CFAR method were also generated. Also, in this thesis, the performance criteria obtained from the synthesis process of the FPGA-based LSTM neural network and the synthesis results of the CFAR method on FPGA is compared. The obtained results show that the FPGA-based LSTM neural network, which proposed in this thesis, can be implemented on a hardware (FPGA) and can detect pulsed signal at low SNR levels where the traditional threshold method (CFAR) cannot obtain successful results, however, it shows that this proposed architecture cannot meet the real-time radar receiver requirements of an Electronic Warfare System.

KEYWORDS: FPGA, LSTM, Long Short-Term Memory, Artificial Neural Network, Low SNR Level, Pulse Signal Detection, Constant False Alarm Rate, CFAR

İÇİNDEKİLER

TEŞEKKÜR	i
ÖZET	ii
ABSTRACT	iii
İÇİNDEKİLER	iv
ŞEKİLLER LİSTESİ	v
TABLolar LİSTESİ	ix
SİMGELER VE KISALTMALAR LİSTESİ	x
1. GİRİŞ	1
1.1. Tezin Amacı	10
1.2. Tezin Yapısı	10
2. LİTERATÜR TARAMASI	11
2.1. Yapay Sinir Ağları ile Radar Sinyali Tespit Çalışmaları	11
2.2. FPGA’de Yapay Sinir Ağı Uygulamaları	19
3. TASARIM ÇALIŞMALARI	26
3.1. MATLAB/Simulink Modeli İle Eğitim ve Test Veri Setlerinin Elde Edilmesi	26
3.2. MATLAB ile SYAO Darbesel Sinyal Tespit Yöntemi Çalışmaları	35
3.3. SYAO Darbesel Sinyal Tespit Yöntemi FPGA Tasarım Çalışması	38
3.4. SYAO Darbesel Sinyal Tespit Yöntemi FPGA Benzetim Çalışması	39
3.5. Uzun Kısa-Sürelili Bellek Sinir Ağı MATLAB Çalışmaları	42
3.6. Uzun Kısa-Sürelili Bellek Sinir Ağı FPGA Tasarım Çalışmaları	45
3.7. Uzun Kısa-Sürelili Bellek Sinir Ağı FPGA Benzetim Çalışmaları	61
4. SONUÇ VE ÖNERİLER	65
KAYNAKLAR	69

ŞEKİLLER LİSTESİ

Sayfa

Şekil 1.1	Elektronik harpte bulunan örnek elektromanyetik dalgalar [1].....	2
Şekil 1.2	Radar sistemlerinde gönderilen ve geri alınan elektromanyetik darbeler [4]	3
Şekil 1.3	Darbeli bir radar vericisindeki sinyal dalga biçimleri [5]	4
Şekil 1.4	Darbeli radar sinyal parametreleri [6]	4
Şekil 1.5	Tipik bir radar zaman ekseni [8]	5
Şekil 1.6	Sinyal zarfının tespit örneği [5]	6
Şekil 1.7	Yüksek SGO'ya sahip hedef (1) ve yanlış algılama (2) örneği [4]	6
Şekil 1.8	Zamanın bir fonksiyonu olarak bir radar alıcısının tipik zarfı [9]	7
Şekil 1.9	Tipik bir UKSB Tekrarlayan Sinir Ağı mimarisi [11]	8
Şekil 1.10	FPGA tabanlı Uzun Kısa-Süreli Bellek Yapay Sinir Ağı Örneği [14]	9
Şekil 2.1	Önerilen ESA dedektörünün yapısı [16]	11
Şekil 2.2	Farklı SGO altında ESA dedektörü ve SYAO dedektörünün tespit olasılıkları [16]	12
Şekil 2.3	Tespitçi yapısının eğri altında kalan değerleri [10]	13
Şekil 2.4	-20 dB SGO'da önerilen tespitçi yapısının kullanılmasıyla gerçekleştirilen sinyal tespiti [10]	14
Şekil 2.5	-5 dB SGO'da algılama yöntemlerinin karşılaştırılması [15]	15
Şekil 2.6	Önerilen TSA tabanlı Otomatik Hedef Tanıma sistem modeli [18]	16
Şekil 2.7	Nihai UKSB model yapısı ve Derin Sinir Ağ model yapısı [20]	18
Şekil 2.8	Basit bir nöron mimarisi [21]	19
Şekil 2.9	Nöronun donanım uygulamasının temel yapısı [22]	20
Şekil 2.10	FPGA tabanlı YSA'da bir katmanın yapısı [22]	20
Şekil 2.11	Tam sayı veya kayan nokta kullanan YSA karşılaştırması [23]	21
Şekil 2.12	Farklı platformlarda UKSB modelinin yürütme süresi (ne kadar düşükse o kadar iyidir) [24]	22

Şekil 2.13	Farklı platformların birim güç başına performansı (ne kadar yüksekse o kadar iyi) [24]	23
Şekil 2.14	Hızlandırma (daha yüksek daha iyidir) [26]	24
Şekil 2.15	Güç Tüketimi (daha düşük daha iyidir) [26]	24
Şekil 3.1	Genel MATLAB/Simulink Modeli	26
Şekil 3.2	Darbesel sinyal, Taşıyıcı sinyal ve Modüle edilmiş darbesel sinyal Scope görüntüsü	30
Şekil 3.3	Modüle edilmiş ve gürültü eklenmiş darbesel sinyallerin Scope görüntüsü a)5 dB b)0 dB c)-5 dB	31
Şekil 3.4	Sadece gürültü sinyali elde etmek için oluşturulan MATLAB/Simulink Modeli.....	33
Şekil 3.5	Eşik değeri yöntemi için hazırlanan MATLAB/Simulink modeli	35
Şekil 3.6	Eşik değeri yöntemi ile sinyal tespiti için eşik değerinin hesaplanması örneği	36
Şekil 3.7	10 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması	37
Şekil 3.8	5 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması	37
Şekil 3.9	0 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması	37
Şekil 3.10	-5 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması	37
Şekil 3.11	SYAO yönteminin FPGA'de uygulanması.....	38
Şekil 3.12	SYAO yönteminin FPGA'de uygulanmasının RTL şeması	38
Şekil 3.13	SYAO Yöntemi FPGA Benzetim Çalışması.....	39
Şekil 3.14	10 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması	40
Şekil 3.15	5 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması	40
Şekil 3.16	0 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması	41
Şekil 3.17	-5 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması	41
Şekil 3.18	MATLAB UKSB sinir ağı mimarisi [30].....	42
Şekil 3.19	MATLAB UKSB yapay sinir ağı eğitim ekranı.....	43
Şekil 3.20	-5 dB SGO için eğitilen MATLAB UKSB sinir ağı'nın ağırlık ve bias değerleri	43

Şekil 3.21	MATLAB UKSB’de 5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi.....	44
Şekil 3.22	MATLAB UKSB’de 0 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi.....	44
Şekil 3.23	MATLAB UKSB’de -5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi.....	44
Şekil 3.24	Tipik bir UKSB hücre yapısı [31]	45
Şekil 3.25	FPGA UKSB Hiyerarşi	46
Şekil 3.26	IEEE-754 Tek Duyarlılık Kayan Nokta (Single Precision Floating Point).....	47
Şekil 3.27	FPGA UKSB Detaylı Hiyerarşi.....	48
Şekil 3.28	Çarpma işlemi için kullanılan IP çekirdeği	49
Şekil 3.29	Toplama işlemi için kullanılan IP çekirdeği.....	49
Şekil 3.30	Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği	50
Şekil 3.31	Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları	50
Şekil 3.32	Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği	51
Şekil 3.33	Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları	51
Şekil 3.34	Block Memory Generator IP çekirdeği	52
Şekil 3.35	Block Memory Generator IP çekirdeği parametreleri	52
Şekil 3.36	Sigmoid fonksiyonu değerlerinin IP çekirdeğine yüklenmesi.....	52
Şekil 3.37	CORDIC IP çekirdeği.....	53
Şekil 3.38	Bölme işlemi için kullanılan IP çekirdeği	53
Şekil 3.39	Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği	54
Şekil 3.40	Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları	54
Şekil 3.41	Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği	55

Şekil 3.42	Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları	55
Şekil 3.43	FPGA'de tasarımı tamamlanan bir UKSB hücresi	56
Şekil 3.44	UKSB Tekrarlayan Sinir Ağı Yapısı [11]	56
Şekil 3.45	UKSB hücrelerinin zincir yapısı	57
Şekil 3.46	UKSB hücresinin kendini tekrarlayan zincir yapısı	57
Şekil 3.47	UKSB hücresinin kendini tekrarlayan zincir yapısının FPGA uygulaması	58
Şekil 3.48	FPGA'de tasarımı tamamlanan UKSB sinir ağı	58
Şekil 3.49	MATLAB UKSB sınıflandırma katmanı ağırlık ve bias değerleri	59
Şekil 3.50	FPGA'de tasarımı tamamlanan sınıflandırma modülü	60
Şekil 3.51	FPGA'de sınıflandırma modülünün hiyerarşik yapısı	60
Şekil 3.52	6 numaralı test verisi	61
Şekil 3.53	UKSB FPGA Benzetim Çalışması	62
Şekil 3.54	FPGA benzetim çalışması sınıflandırma işlemi	63
Şekil 3.55	Benzetim çalışması örneği 1 (Darbesel sinyal mevcut)	63
Şekil 3.56	Benzetim çalışması örneği 2 (Darbesel sinyal mevcut değil)	63
Şekil 3.57	Benzetim çalışması örnek 3 (Darbesel sinyal mevcut)	64
Şekil 3.58	Benzetim çalışması örnek 4 (Darbesel sinyal mevcut değil)	64
Şekil 3.59	FPGA tabanlı UKSB'de 5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi	64
Şekil 3.60	FPGA tabanlı UKSB'de 0 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi	64
Şekil 3.61	FPGA tabanlı UKSB'de -5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi	65

TABLÖLAR LİSTESİ

	Sayfa
Tablo 2.1 FPGA Kaynak Kullanımı [22]	21
Tablo 2.2 Zynq ZC7020 için FPGA kaynak kullanımı [24].....	22
Tablo 3.1 MATLAB/Simulink Modelinde Kullanılan Bloklar ve Blok Parametreleri	27
Tablo 3.2 Modüle edilmiş darbesel sinyalin gücünün hesaplanması	32
Tablo 3.3 MATLAB/Simulink Modelleri ile Oluşturulan Eğitim ve Test Veri Setleri....	34
Tablo 3.4 SYAO yöntemi FPGA kaynak tüketimi	39
Tablo 3.5 Tipik bir UKSB hücresi eşitlikleri.....	45
Tablo 3.6 UKSB sinir ağının FPGA kaynak tüketimi	60

SİMGELER VE KISALTMALAR LİSTESİ

ASIC	Uygulamaya Özel Tümlşik Devre (Application Specific Integrated Circuit)
CFAR	Sabit Yanlıř Alarm Oranı (Constant False Alarm Rate)
CPU	Merkezi İşlem Birimi (Central Processing Unit)
ÇKA	Çok Katmanlı Algılayıcı
DSA	Derin Sinir Ağı
DTK	Darbe Tanımlama Kelimesi
ELINT	Elektronik İstihbarat (Electronic Intelligence)
ESM	Elektronik Savaş Destek Önlemleri (Electronic Warfare Support Measures)
ESA	Evriřimsel Sinir Ağı
FPGA	Alan Programlanabilir Kapı Dizileri (Field Programmable Gate Array)
FMSD	Frekans Modülasyonlu Sürekli Dalga
GPU	Grafik İşleme Birimi (Graphics Processing Unit)
GPGPU	Genel Amaçlı Grafik İşleme Birimi (General-Purpose GPU)
GOP/s	Saniye Başına Milyar İşlem (Giga Operations Per Second)
LSTM	Uzun Kısa-Süreli Bellek (Long Short-Term Memory)
RNN	Tekrarlayan Sinir Ağı (Recurrent Neural Network)
RTRL	Gerçek Zamanlı Tekrarlayan Öğrenme (Real Time Recurrent Learning)
RTL	Yazmaç Transfer Seviyesi (Register Transfer Level)
SGO	Sinyal Gürültü Oranı
SYAO	Sabit Yanlıř Alarm Oranı
TO	Tespit Olasılığı
TSA	Tekrarlayan Sinir Ağı
UKSB	Uzun Kısa-Süreli Bellek
YSA	Yapay Sinir Ağı

1. GİRİŞ

Elektronik Harp (EH) ülkelerin sahip oldukları EH teknolojileri ve istihbarat bilgileri ile elektromanyetik tayfin kontrolünü ele geçirme savaşıdır. Bu sayede, çeşitli yöntemlerin kullanılmasıyla elektromanyetik spektrumun istenmeyen düşman güçler tarafından kullanımını engellerken, dost kuvvetler tarafından kullanımını ise askeri hedefler doğrultusunda en yararlı hale getirilmesini hedefleyen sistemlerdir. Elektronik harp; üç başlık altında incelenebilir [1]:

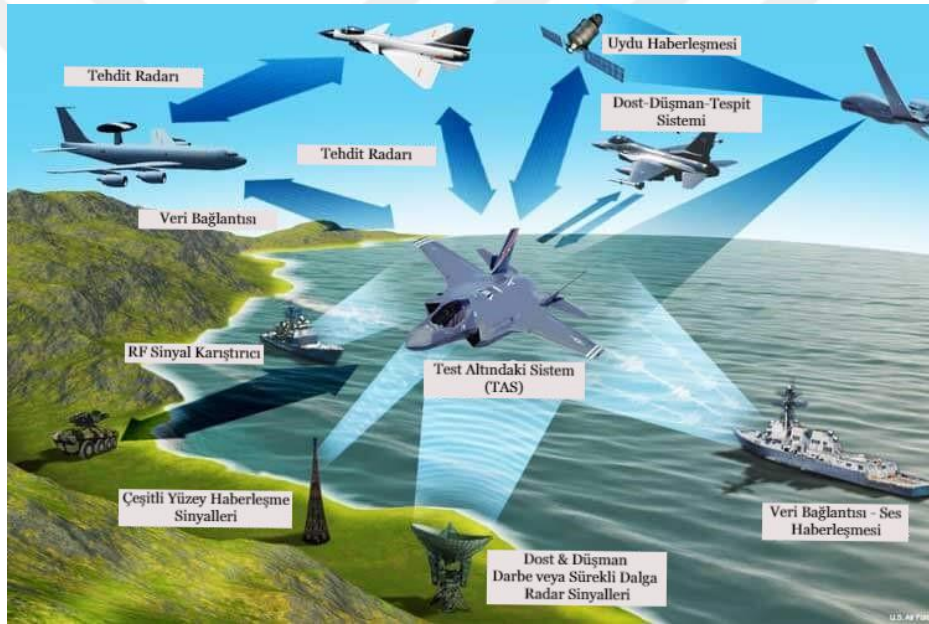
Elektronik Taarruz (ET): Başta elektromanyetik radyo dalgaları ve kullanılan radar frekansları olmak üzere, düşman güçler tarafından elektromanyetik spektrumun kullanılmasının engellenmesi, istihbarat paylaşım kabiliyetlerinin düşürülmesi ve elektronik haberleşmenin engellenmesidir [1].

Elektronik Savunma (ES): Çeşitli elektronik sistemleri, haberleşme altyapısını ve çeşitli askeri teçhizatları, düşman güçler tarafından gelebilecek elektronik saldırılara karşı korumaktadır. Bunun en kolay yöntemi ise üretilen tüm elektronik sistemleri, elektronik saldırıya karşı korumalı geliştirmektir. Buna ek olarak, elektronik harp savunması amacıyla; Düşmana Karşı Tedbir Atma, Elektronik Savaş Kendini Koruma, Erken İkaz Sistemleri ve Elektromanyetik Dalga Karıştırıcı gibi çeşitli sistemler ve yöntemler mevcuttur [1].

Elektronik Destek (ED): Elektromanyetik spektrumun pasif şekilde kullanımı ile harp bölgesinde bulunan düşman ve dost birliklerinin hızlarını, bu birliklerin hareket yönlerini, konumlarını ve ayrıca kendi aralarında gerçekleştirilen haberleşmelerin içeriğinin tespit edilmesinde önemli görev üstlenir. Ayrıca istihbarat bilgilerinin elde edilmesinde, düşman güçler tarafından yayılan sinyallerin tespit ve analiz edilmesinde de önemli rol oynamaktadır [1].

ED sistemleri, düşman elektromanyetik dalga yayıcılarından gelen sinyallerin toplanmasında ve analiz edilmesinde önemli rol oynamaktadır. Ancak bununla birlikte, çoğu durumda sinyal hakkında önceden bilgiye sahip olmadığından ve bazen sinyal çok zayıf veya çok gürültülü olduğundan, anlamlı sinyalleri toplamakta zorluk çekebilirler. ED sistemleri, işlenmiş bilgileri dost elektronik taarruz ve elektronik savunma sistemlerine sağlar. Bilginin başarılı bir şekilde elde edilmesi, elektronik taarruz ve elektronik savunma sistemlerinin görevlerini yerine getirmek için önemli bir ön koşuldur [2].

Dost kuvvetleri düşman tehditlerinden korumak için, tüm tehdit edici radar sinyalleri dikkatlice tespit edilmeli ve analiz edilmelidir. Şekil 1.1’de elektronik harpte bulunan örnek elektromanyetik dalgalar gösterilmiştir. Çeşitli nedenlerden dolayı hedef sinyallerin saptanması giderek daha zor hale gelmektedir: Birincisi, düşman elektromanyetik dalga yayıcılarından gelen sinyallerin frekans bandı ve yönü gibi bilgileri önceden elde etmek neredeyse imkânsızdır. İkincisi, radar sinyali, gürültü ve iletim kanalı koşulları gibi çeşitli nedenler tarafından kolayca bozulur. Üçüncüsü, modern savaşta tanımlanamayan nesneleri aramak ve izlemek için birçok yayıcı türü aynı anda çalıştırılır. Çeşitli sinyaller birbirleri ile etkileşime girerek hedef sinyali çıkarmayı zorlaştırır. Son olarak, düşman radar sinyallerinin toplanması genellikle uzun mesafelerden gerçekleştirilir, bu da güç zayıflamasına neden olur[2].

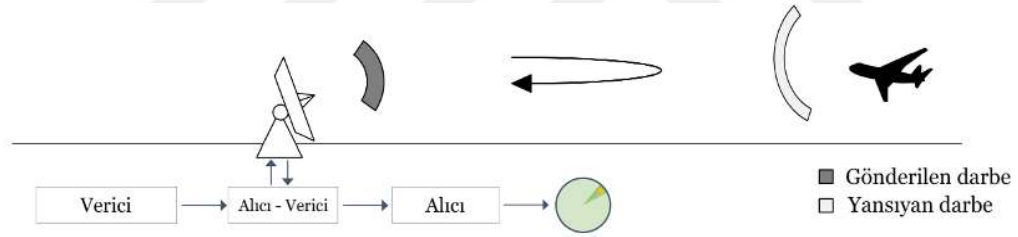


Şekil 1.1 Elektronik harpte bulunan örnek elektromanyetik dalgalar [1]

Radar İkaz Alıcı (Radar Warning Receiver, RWR) sistemleri, radar sistemleri tarafından yayılan elektromanyetik dalgaları tespit etme prensibi üzerine çalışırlar. Bu sistemler tehdit oluşturabilecek bir radar sinyali tespit edildiğinde bir uyarı oluşturmaktadırlar. Oluşturulan bu uyarı daha sonra algılanan tehditten kaçınmak için manuel veya otomatik olarak kullanılabilir. Ayrıca diğer dost elektronik savaş sistemlerine yararlı bilgiler sağlamak için radar sinyallerinin tespit ve analiz edilmesi kritik bir öneme sahiptir. Zayıf bir radar sinyalini tespit etmek için gürültülü ortamlarda uygulanabilen sinyal toplama yöntemleri gereklidir. Bu doğrudan elektronik destek sistemlerinin algılama performansı ile ilgilidir.

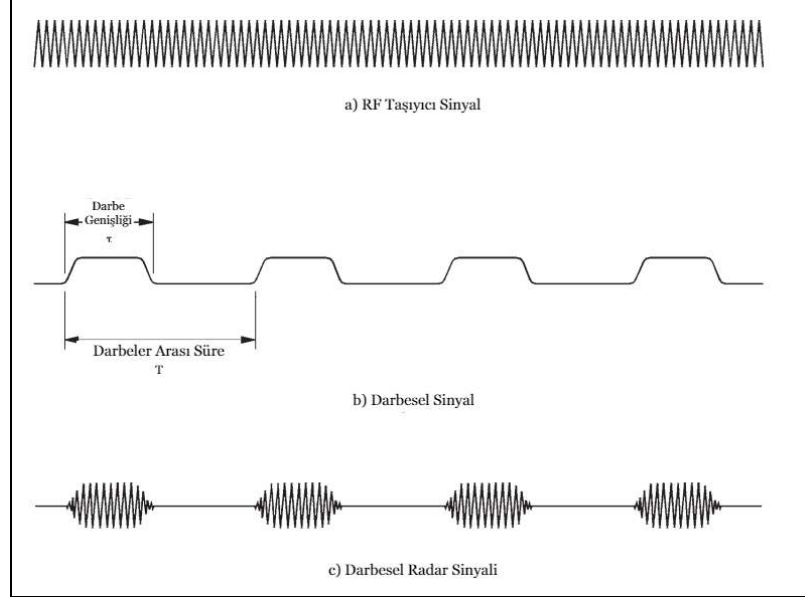
Elektromanyetik dalgalar yardımıyla cisimlerin konum, büyüklük ve hız gibi parametrelerini tespit etmekte kullanılan radar sistemleri, elektronik harpte kullanılan en kritik sistemlerden bir tanesidir. Radarlar hem sivil hem de askeri amaçlarla kullanılırlar. Sivil amaçla kullanımda arama/kurtarma ve güvenlik, navigasyon sistemleri ve hava durumu radarları, hava trafiği kontrolü ve uçuş yönetimi sistemleri, trafik yönetimi için çeşitli radar sistemleri ve gemi trafiği gibi çeşitli alanlarında kullanılırlar. Askeri amaçla kullanımda ise askeri operasyonlarda konumun elde edilmesi, erken ikaz sistemleri, gözetleme, hedef sınıflandırma, hava saldırısı savunması ve güdüm sistemleri gibi görevlerde kullanılırlar.

Radar sistemlerinde Şekil 1.2’de gösterildiği gibi elektromanyetik dalga darbeleri gönderilir ve bu dalgaların sahip oldukları elektromanyetik enerji hedefe ulaştıktan sonra hedeften yansıyarak geri döner, bu yansıyan sinyallere yankı (Echo) sinyali denir. Radar sistemleri için yansıyan bu sinyalin geri gelme süresi, yönü ve şiddeti kritiktir. Bu parametreler ile tespit edilen hedefin özellikleri elde edilir. Radarnın gönderdiği elektromanyetik dalgadan yansıyıp tekrar radara ulaşan yankının bir eşik değerinin (Threshold) üzerinde olması gereklidir [3].



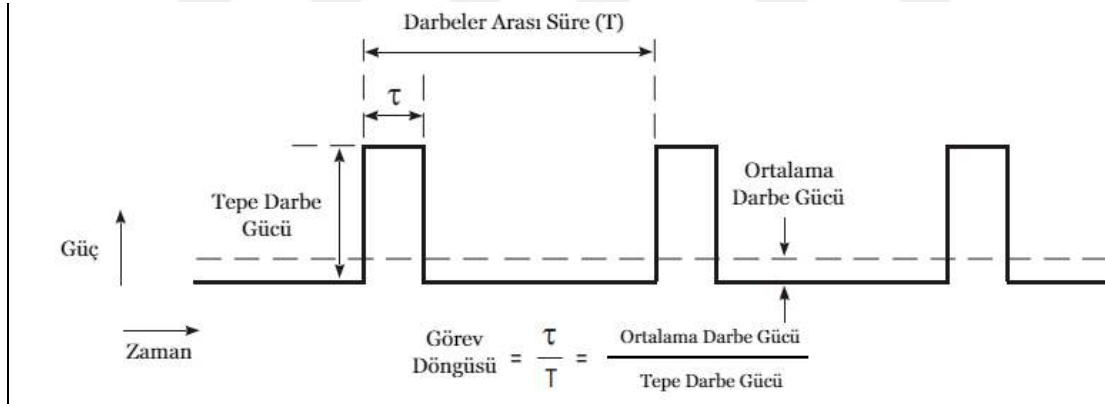
Şekil 1.2 Radar sistemlerinde gönderilen ve geri alınan elektromanyetik darbeler [4]

Radar sistemleri, darbe radarları ve sürekli dalga radarları olarak ayrılabilir. Darbe radar sistemleri popüler olarak kullanılan radar türüdür. Bu radar sistemleri elektromanyetik dalgaları gönderip yansıma ile arasında geçen zamanı işleyerek hedef bilgisini elde eden sistem mantığına göre çalışırlar. Çoğunlukla sinyalin gönderilmesi ve alınması için tek anten içerirler. Elektromanyetik dalga darbelerini gönderdikten sonra hedeften yansıyıp geri gelmelerini beklerler. Sürekli dalga radarları ise sinyalin gönderilmesi ve alınması için, verici ve alıcı olmak üzere iki farklı anten kullanarak Doppler kayması ilkesine uygun olarak çalışırlar [3]. Darbe sinyalli bir radar sisteminde, radar vericisinin amacı, anten tarafından iletilecek darbeli bir RF sinyali üretmektir. Şekil 1.3 sürekli ve sinüzoidal bir RF taşıyıcısını modüle etmek için bir darbe dizisinin kullanıldığı bir darbeli radar vericisinde bulunan dalga biçimlerini göstermektedir. Ortaya çıkan dalga biçimi, darbeli bir sinüs dalgasıdır [5].



Şekil 1.3 Darbeli bir radar vericisindeki sinyal dalga biçimleri [5]

Şekil 1.4, darbe radarının temel parametreleri arasındaki ilişkiyi göstermektedir. Şekil 1.5 tipik bir radar zaman eksenini göstermektedir.



Şekil 1.4 Darbeli radar sinyal parametreleri [6]

Darbe Genişliği, τ , (Pulse Width): Radar sistemi tarafından iletilen enerji darbesinin süresidir ve tek bir darbenin zaman süresi ' τ ' ile temsil edilir.

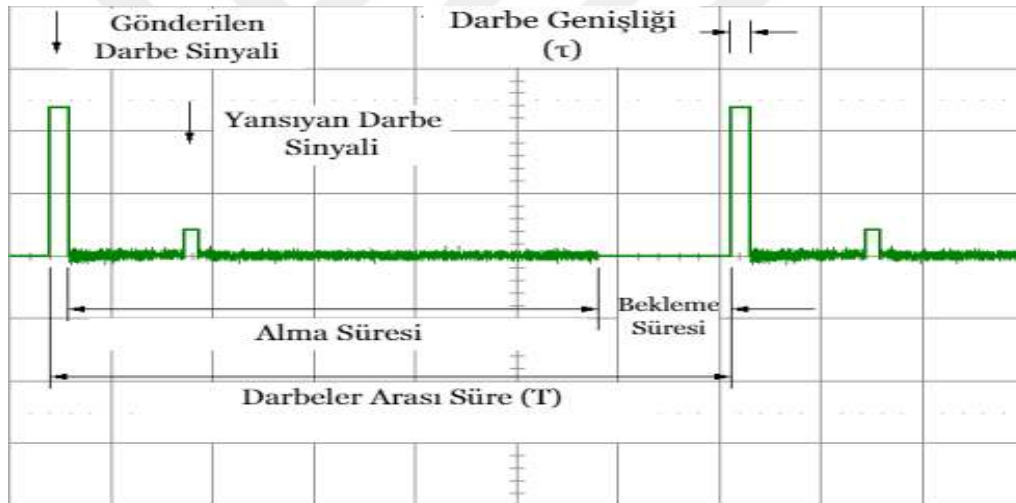
Darbeler Arası Süre, T , (Interpulse Period): Bir darbenin başlangıcı ile bir sonraki darbenin başlangıcı arasındaki süre, darbeler arası süre veya darbe tekrarlama süresi olarak adlandırılır.

Darbe Tekrarlama Frekansı, PRF, (Pulse Repetition Frequency): Bir saniyede meydana gelen darbe sayısı olarak tanımlanır. Darbe Tekrarlama Frekansı ile Darbeler Arası Süre arasında $PRF=1/T$ bağlantısı vardır.

Radar Görev Döngüsü (Radar Duty Cycle): Darbe genişliğinin darbeler arası süreye oranı (τ/T) görev döngüsü (duty cycle) olarak bilinir. Bu oran radar vericisinin çalıştığı süreyi temsil eder. Görev döngüsü ayrıca vericinin ortalama gücü ile tepe darbe gücü arasındaki ilişkiyi de ifade eder.

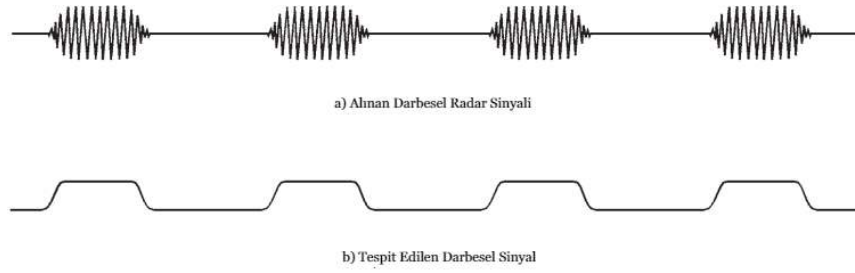
Tepe Darbe Gücü (Peak Power): Belirli bir yönde maksimum açık menzilli hedef tespiti için iletilebilen radar anteninin maksimum gücüdür [7].

Ortalama Darbe Gücü (Mean Power): Tüm yönlerde maksimum açık menzilli hedef tespiti için iletilebilen radar anteninin ortalama gücüdür [7].



Şekil 1.5 Tipik bir radar zaman eksenini [8]

Çoğu radar alıcısı, modüle edilen orijinal dalga formunu geri kazanmak için alınan sinyalin zarfını tespit ederek çalışır. Zarf tespit örneği Şekil 1.6'da gösterilmiştir. Yüksek frekanslı taşıyıcı sinyal, radar alıcısı tarafından alınan sinyalden çıkarılır ve zarfın yalnızca pozitif kısmı korunur. Tespit edilen darbeler daha sonra daha fazla işlem ve görüntüleme için güçlendirilir [5].

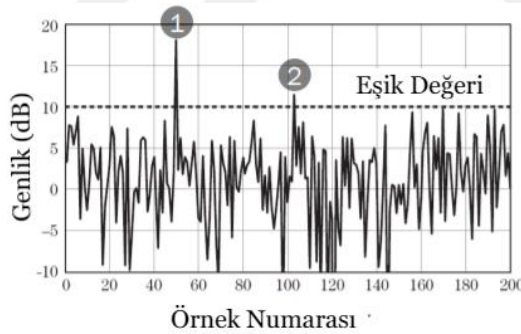


Şekil 1.6 Sinyal zarfının tespit örneği [5]

Sinyaldeki gürültü, istatistiksel özelliklerle karakterize edilen rastgele, istenmeyen bir sinyaldir. Gürültü kaynakları dahili (ekipman kusurları) veya harici (diğer RF aktarımları), pasif (parazit yansımalar) veya aktif (karıştırıcılar) olabilir. Anlamlı ve istenmeyen sinyal arasındaki güç oranı, Sinyal-Gürültü-Oranı (SGO) (Signal-to-Noise-Ratio, SNR) olarak tanımlanır [4].

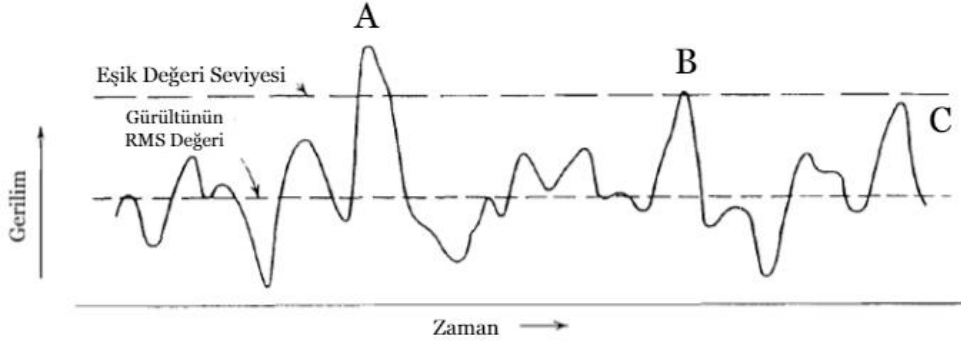
$$SGO (SNR) = \frac{P_{sinyal}}{P_{Gürültü}} \quad (1.1)$$

Şekil 1.7’de yüksek SGO’ya sahip hedef (1) ve yanlış algılama (2) örneği görülmektedir. Radar sistemleri, gürültü ve düşük SGO’ya sahip hedefler arasında ayırım yapmakta zorlanmaktadır [4].



Şekil 1.7 Yüksek SGO’ya sahip hedef (1) ve yanlış algılama (2) örneği [4]

Bir radar alıcısının zayıf bir yankı sinyalini algılama yeteneği, frekans spektrumunda mevcut olan gürültü ile sınırlıdır. Alıcının algılayabileceği en zayıf sinyal, minimum algılanabilir sinyal (Minimum Detectable Signal, MDS) olarak adlandırılır. İstatistiksel yapısı nedeniyle minimum algılanabilir sinyalin ne olduğunu tanımlamak zordur ve bir hedefin mevcut olup olmadığına karar verme kriteri çok iyi tanımlanmayabilir. Algılama Şekil 1.7’de ve Şekil 1.8’de noktalı çizgiler ile gösterildiği gibi alıcıda bir eşik seviyesi oluşturmaya dayanır. Yankı ve gürültü karışımı olan alıcıdaki sinyal bu eşik geçtiğinde hedef olarak algılanır. Buna eşik algılama yöntemi (Threshold detection) denir [9].



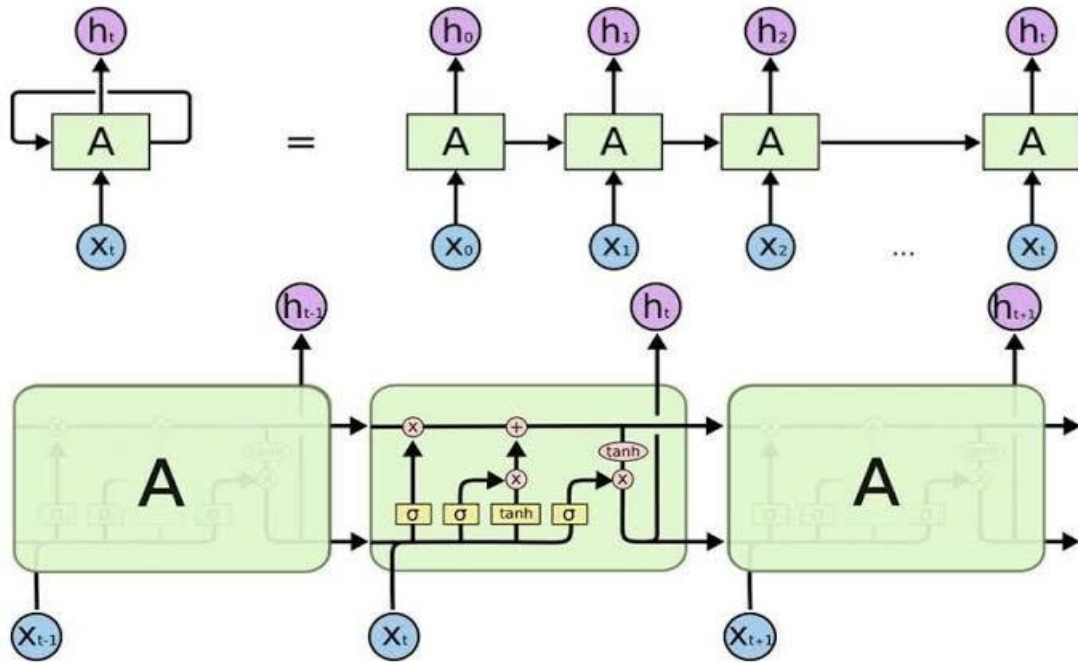
Şekil 1.8 Zamanın bir fonksiyonu olarak bir radar alıcısının tipik zarfı [9]

A, B ve C noktaları gürültü içeren sinyalleri temsil eden üç hedeftir. A ve B geçerli sinyal tespitleridir, ancak C hatalı tespittir. Eğer eşik seviyesi uygun şekilde ayarlanırsa ve sadece bir gürültü sinyali varsa bu eşik aşamaz, ancak gürültüyle birlikte güçlü bir sinyal mevcutsa bu eşik aşılabilecektir. Eğer eşik seviyesi düşük ayarlanırsa, gürültü sinyali bunu aşabilir ve bir hedefle karıştırılabilir. Buna *yanlış alarm (false alarm)* denir. Eğer eşik seviyesi yüksek ayarlanırsa, gürültü sinyali yanlış alarmlara neden olacak kadar büyük olmayabilir, ancak bu sefer de zayıf hedef yankıları eşik geçemeyebilir ve tespit edilemeyebilir. Buna da *kaçırılan algılama (missed detection)* denir. Şekil 1.8'de A'daki sinyal gürültüden daha büyük bir genliğe sahiptir. Bu nedenle hedef tespiti herhangi bir zorluk ve belirsizlik olmadan mümkündür. Ardından, B'de sinyale eşlik eden gürültü miktarı ile sinyal artı gürültü kombinasyonu eşikini aşp hedef tespiti yapılacak şekilde yeterince büyüktür. Bu nedenle, gürültünün varlığı bazen zayıf sinyallerin algılanmasını sağlayabilir. Ancak C hedefi algılama eşikini geçmez ve bu nedenle C hedefi tespit edilemez. Eğer eşik seviyesi daha düşük olsaydı, C gibi zayıf sinyaller tespit edilebilirdi ancak düşük eşik seviyesi yanlış alarmlara neden olabilirdi. Sonuç olarak eşik değeri olması gerekenden düşük ayarlanırsa yanlış hedef göstergeleri elde edilir, ancak olması gerekenden yüksek ayarlanırsa hedefler kaçırılabilir. Bu nedenle olası hatalardan kaçınmak için uygun eşik seviyesinin seçimi çok önemlidir [9].

Düşük SGO seviyelerinde klasik eşik değeri yöntemi (SYAO yöntemi) ile radar sinyal tespiti aşamasında hatalı sonuçlar elde edilebilir. Sinyal tespit çalışmalarında, alınan sinyalin gürültüden arındırılmasına ve tespit edilmesine yönelik literatürde çeşitli uygulamalar mevcuttur: Uyumlu filtre (Matched filter) uygulamaları, Dalgacık dönüşümü (Wavelet transform) uygulamaları, Enerji Dedektörü uygulamaları ve Zaman-Frekans Analizi Dedektörü (Time-Frequency Analysis Detector) uygulamaları örnek olarak

verilebilir. Bu tez çalışmasında da bu konuda son zamanlarda popüler çalışma alanlarından bir tanesi olan yapay zekâ yöntemi ile sinyal tespit çalışmalarına odaklanılmıştır ve elde edilen tüm sonuçlar, sinyal tespit uygulamalarında en yaygın yöntem olarak kullanılan eşik değeri yöntemi (SYAO yöntemi) ile karşılaştırılmıştır.

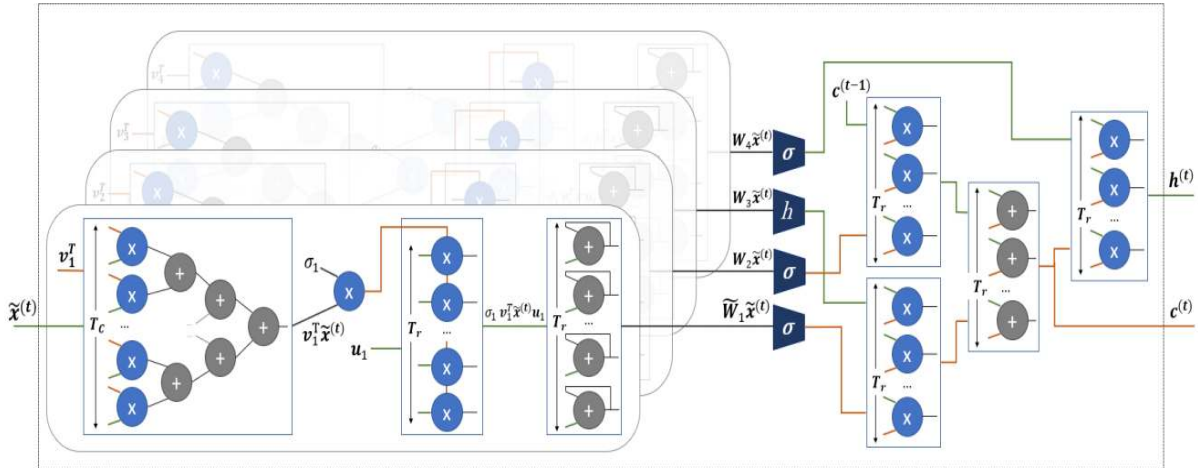
Zaman serisi verilerin işlenmesi hedefiyle son zamanlarda derin öğrenmenin de ilerlemesiyle yapay sinir ağları popüler olarak kullanılmaya başlanmıştır. Bir yapay sinir ağı türü olan Uzun Kısa-Süreli Bellek (UKSB) (Long Short-Term Memory, LSTM), geleneksel yapay sinir ağlarındaki sürekli azalan gradyanlar problemine çözüm getirebildiği için zaman serisi video ve ses verilerinde popüler olarak kullanılmaktadır. Sürekli azalan gradyanlar problemi insanların beynindeki unutma davranışı olarak da düşünülebilir. Geleneksel yapay sinir ağları verinin uzunluğu arttıkça bilgiler arasında bağlantı kurulmasında çeşitli zorluklar yaşamaktadırlar. UKSB bu problemi yapısındaki geçit yapılarını kullanarak hafıza hücreleri arasındaki bilgi transferini en iyi hale getirerek çözer. Tipik bir UKSB Tekrarlayan Sinir Ağı (TSA) (Recurrent Neural Network, RNN) mimarisi Şekil 1.9'da gösterilmiştir [10].



Şekil 1.9 Tipik bir UKSB Tekrarlayan Sinir Ağı mimarisi [11]

Tekrarlayan Sinir Ağları sıralı ve zamansal verilerde uzun süreli bağımlılıkları yakalama yeteneğine sahip bir sinir ağı sınıfıdır. UKSB yapay sinir ağları, çeşitli uygulamalarda büyük başarılar elde eden en popüler tekrarlayan sinir ağı türlerinden biridir. Bu uygulamalar yüksek hesaplama hızı ve düşük güç tüketimi gerektirdiğinden, UKSB'lerin özelleştirilmiş donanım hızlandırma çözümleri, yukarıdaki gereksinimleri karşılayarak bilgi işlem hızını artırmanın önemli bir yolu haline gelmektedir [12].

Tekrarlayan sinir ağlarının doğası nedeniyle, zamansal verileri ve bağımlılıkları işlemesi gerektiğinde bilgi işlem modeli karmaşık hale gelmektedir ve çok sayıda paralel bellek erişimi gerektirmektedir. Yüksek hesaplama karmaşıklığı ve yüksek paralel bellek erişimi, donanım tabanlı UKSB modellerine yönelimi artırmıştır. Çeşitli çalışmalarda GPU'lar, FPGA'lar ve ASIC'ler, UKSB yapay sinir ağlarında hızlandırıcı olarak kullanılmıştır. Çok sayıda platform arasında FPGA, esnek yeniden yapılandırılabilirlik ile yüksek performans elde edebildiği için sinir ağlarını hızlandırmak için umut verici bir çözüm potansiyeli göstermektedir [12]. Şekil 1.10'da FPGA tabanlı UKB sinir ağı örneği görülmektedir. Evrimsel Sinir Ağı (ESA) (Convolutional Neural Network, CNN) hızlandırılması üzerine yapılan çalışmada [13], FPGA'nın GPU ile karşılaştırılabilir, ancak GPU'ya nazaran çok daha az enerji tüketimi ile yüksek performans elde edebileceğini ortaya koymuştur.



Şekil 1.10 FPGA tabanlı Uzun Kısa-Süreli Bellek Yapay Sinir Ağı Örneği [14]

1.1. Tezin Amacı

Elektronik harp destek sistemleri, düşman elektromanyetik dalga yayıcılarından gelen sinyallerin toplanmasında ve analiz edilmesinde önemli rol oynamaktadır. Dost kuvvetleri düşman tehditlerinden korumak için, tüm tehdit edici radar sinyalleri dikkatlice tespit edilmeli ve analiz edilmelidir. Çeşitli nedenlerden dolayı hedef sinyallerin saptanması giderek daha zor hale gelmektedir ve düşük SGO seviyelerinde Sabit Yanlış Alarm Oranı (SYAO) yöntemi ile sinyal tespit aşamalarında hatalı sonuçlar elde edilebilmektedir. Bu sorunun önüne geçebilmek için literatürde çeşitli çalışmalar yapılmaktadır ve olumlu sonuçlar elde edilebilmektedir.

Bu tez çalışmasında literatür çalışmalarına ek olarak, sinyal almacı tarafından toplanan darbesel sinyal örnekleri üzerinde herhangi bir işlem yapmadan sadece ham veriler kullanılarak düşük SGO seviyelerinde SYAO yöntemi ile başarı oranının düşük olduğu veya tespit edilemeyen darbesel sinyalleri tespit edebilmek için FPGA tabanlı bir Uzun Kısa-Süreli Bellek yapay sinir ağı mimarisi önerilmiştir. Tez çalışmasının sonucunda SYAO yöntemi ile sinyal tespiti yapılamayan SGO seviyelerinde önerilen mimari ile başarılı sonuçlar elde edilebileceği FPGA benzetim çalışmaları ile gösterilmiştir. UKSB sinir ağının FPGA’de sentezlenmesi sonucu elde edilen performans ölçütleri paylaşılmıştır. Bu performans ölçütleri ayrıca SYAO yönteminin FPGA’de uygulanması sonucu elde edilen performans ölçütleri ile karşılaştırılmıştır.

1.2. Tezin Yapısı

Bu tez giriş bölümünü takip eden 3 bölümden oluşmaktadır. İkinci bölümde çeşitli yapay sinir ağları kullanılarak gerçekleştirilen sinyal tespit çalışmaları ve UKSB sinir ağı başta olmak üzere çeşitli yapay sinir ağlarının FPGA üzerinde uygulanmasına ait literatür çalışmaları paylaşılmıştır. Üçüncü bölüm UKSB sinir ağını eğitmek ve UKSB sinir ağı ile SYAO yöntemini test etmek amacıyla veri setlerinin istenilen özelliklerde üretilmesi ve örneklenmesi için oluşturulan MATLAB/Simulink modeli çalışması, SYAO sinyal tespit yöntemi için gerçekleştirilen MATLAB/Simulink çalışması ve SYAO yönteminin FPGA ile tasarım ve benzetim çalışmalarını içermektedir. Ayrıca bu bölümde UKSB sinir ağı MATLAB projesi ve UKSB sinir ağının FPGA üzerinde oluşturulması için yapılan tasarım çalışmaları detaylı olarak paylaşılmıştır. Son olarak bu bölümde test veri setlerinin FPGA’de tasarlanan UKSB sinir ağına uygulanması ile elde edilen FPGA benzetim çalışmalarının

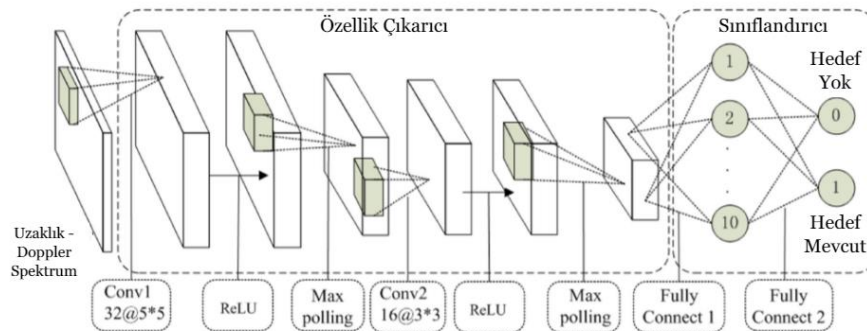
detayları paylaşılmıştır. Dördüncü bölümde bu tez çalışması ile elde edilen tüm sonuçlar ve bu sonuçların karşılaştırılması verilmiştir. Önerilen FPGA tabanlı UKSB sinir ağı yönteminin SYAO yöntemine göre avantaj ve dezavantajlarından bahsedilmiştir. Bu tez çalışmasında sunulan mimarinin gerçek zamanlı uygulamalar için kullanılabilirliği değerlendirilmiştir. Ayrıca bu tez kapsamına katılmamış fakat gelecekte yapılmasının faydalı olduğu düşünülen çalışmalardan bahsedilmiştir.

2. LİTERATÜR TARAMASI

2.1. Yapay Sinir Ağları ile Radar Sinyali Tespit Çalışmaları

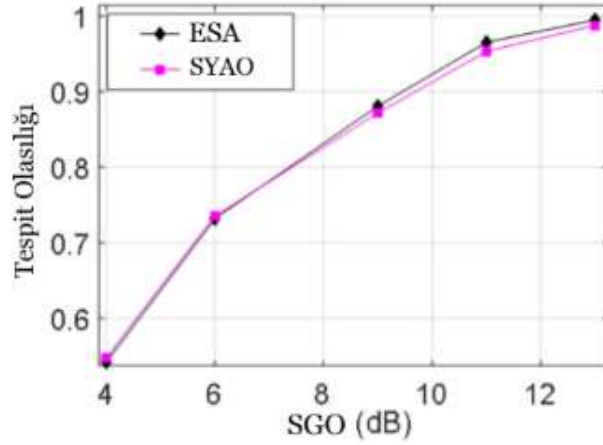
Hedef tespiti, yaygın olarak kullanılan en önemli radar uygulamalarından biridir. Hedef tespiti bir hedeften gelen bir yankıdan (hedef mevcut) veya sadece gürültüye (hedef yok) karşılık gelip gelmediğini ayırt eden bir tür sınıflandırma olarak kabul edilebilir. Derin Sinir Ağı (DSA) (Deep Neural Network, DNN), sınıflandırma için popüler bir konudur ve farklı bilim alanlarında başarıyla uygulanmıştır. Son zamanlarda, birçok araştırmacı radar uygulamaları için DSA'ları önermektedir [16].

Wang, Tang ve Liao tarafından yapılan bu çalışmada [16], Evrimsel Sinir Ağ (ESA) (Convolutional Neural Network, CNN) tabanlı bir dedektör radar hedef tespiti için önerilmiştir. Şekil 2.1'de paylaşılan ESA dedektörü, hedefin sinyal gürültü oranı büyük olduğunda SYAO yönteminden daha iyi bir performansa sahiptir. Tesadüf tespiti (Coincidence detection), ESA dedektörünün performansını artırmaktadır. Tesadüf tespitinde, aynı yanlış alarm olasılığı (YAO) (Probability of false alarm) için ESA dedektörünün tespit olasılığı (TO) (Probability of detection, PD) SYAO tabanlı dedektörlerin TO'sundan daha büyüktür. Yapılan bu çalışmada sonuçlar benzetim verilerine dayanmaktadır.



Şekil 2.1 Önerilen ESA dedektörünün yapısı [16]

Bu çalışmada araştırmacılar, ESA dedektörünün performansı ile SYAO dedektörünün performansını karşılaştırmışlardır. Şekil 2.2 iki dedektörün tespit olasılığını göstermektedir. SGO 6 dB'den büyük olduğunda ESA dedektörünün tespit olasılığının SYAO dedektörünün tespit olasılığından daha büyük olduğu gözlemlenmektedir [16].



Şekil 2.2 Farklı SGO altında ESA dedektörü ve SYAO dedektörünün tespit olasılıkları [16]

Akyon, Nuhoglu, Alp ve Arıkan tarafından yapılan çalışmada [10] araştırmacılar, elektronik savaş sistemlerindeki dijital almaçlar ile toplanan, ortamdaki tehdit radar sistemlerinin göndermiş olduğu darbesel sinyal verileri üzerinden eşzamanlı olarak sinyal gürültü oranını tahmin eden ve darbenin tespitini gerçekleştiren bir çalışma sunmuşlardır. Ayrıca tespit edilen darbesel bölge üzerindeki kiplmeyi sınıflandıran, çok görevli öğrenme ve tekrarlayan yapay sinir ağı tabanlı yeni bir metot önermişlerdir.

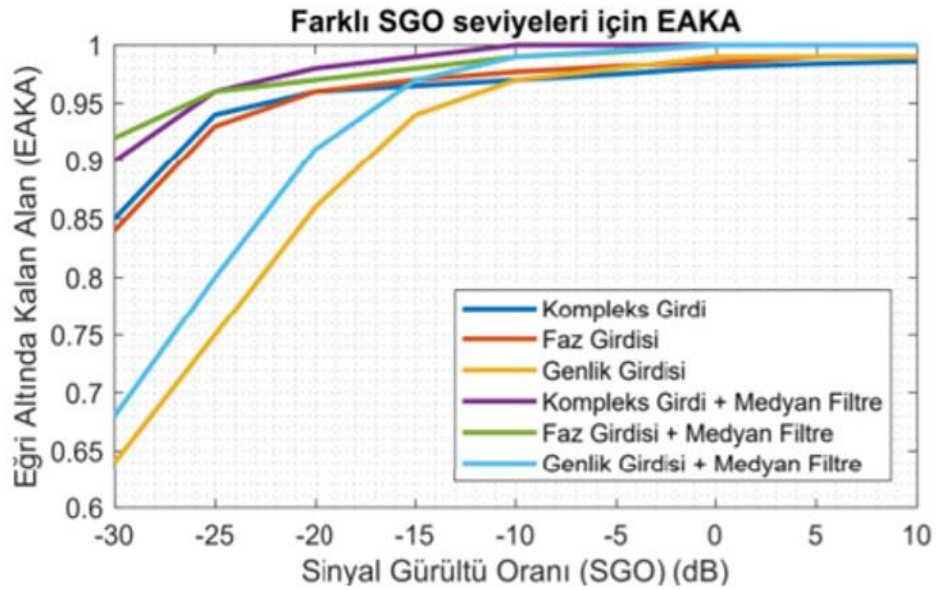
Önerilen bu metot, sistemin girdisi olarak alıcı tarafından elde edilen örneklerden herhangi bir öznelilik çıkarma işlemi gerçekleştirilmeden, ham IQ (Inphase-Quadrature) bilgilerini kullanmaktadır. Sınıflandırma başarı oranını artırmak için, çeşitli farklı sinyal gürültü oranı seviyelerine göre eğitilmiş modeller kullanılmıştır. Ham IQ verileri üzerinden tahmin edilen sinyal gürültü oranı değerine göre, uygun model otomatik olarak seçilmektedir. Gerçekleştirilen detaylı benzetim çalışmalarında, -30 dB sinyal gürültü oranı seviyesinde, 1.5 dB ortalama mutlak hata ile sinyal gürültü oranı tahmini, %90 başarı oranıyla darbesel sinyalin tespitini ve %84 olasılıkla başarılı kipleme sınıflandırması gerçekleştirebildiği gözlemlenmiştir. Geleneksel bir elektronik savaş almamacının darbe tespiti gerçekleştirebildiği en düşük sinyal gürültü oranı seviyesinin 10 dB olduğu göz önüne

alındığında, önerilen bu uygulamanın geleceğin elektronik savaş almaç yapıları için oldukça faydalı bir teknolojik kazanım olduğu düşünülmektedir [10].

Bu çalışmadaki [10] veri setinde -30 ile 10 dB sinyal gürültü oranı arasında, güç seviyeleri 1 mW ile 1 W arasında değişiklik gösteren sinyaller kullanılmıştır. Veri setinde darbe genişlikleri 5-25 μ s arasında değişen, 100 MHz ile örneklenmiş, merkez frekansı -5 ile 5 MHz arasında değişen, frekans ve faz atlamalı sinyallerin çip süreleri en az 0.4 μ s olan, bant genişliği 5 ile 40 MHz arasında değişen kipler bulunmaktadır.

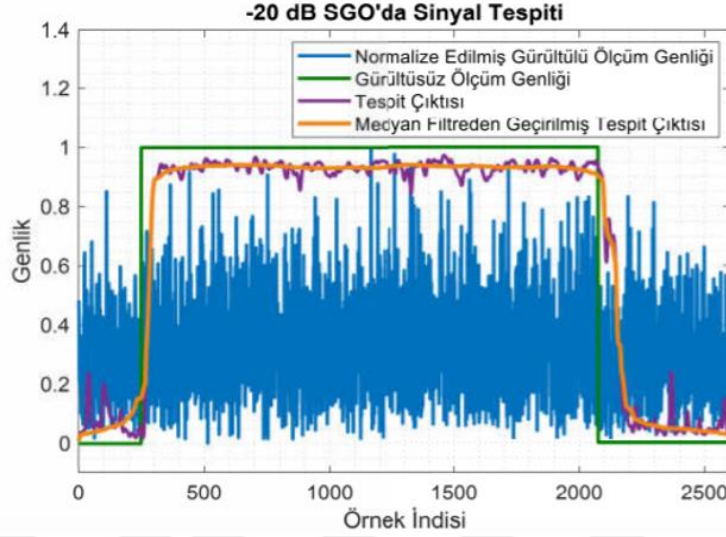
Önerilen bu yöntemler, Python'da Tensorflow kütüphanesi kullanarak uygulanmıştır. Elde edilen sonuçlarda; sistemin girdisi olarak sadece genlik, sadece faz ve kompleks yapı kullanmanın sonuçları, tespitçi yapısının çıktısına medyan filtrenin uygulanmasının etkileri ve çoklu sınıflandırıcı ile tek sınıflandırıcı kullanmanın sonuçları karşılaştırılmıştır [10].

Sadece genlik, sadece faz ve kompleks girdileri kullanıldığında farklı sinyal gürültü oranı seviyelerinde elde edilen tespit başarıları Şekil 2.3'de paylaşılmıştır. Ayrıca Şekil 2.3'de sistemin çıktısına medyan filtre uygulanması ile elde edilen sonuçlar da paylaşılmıştır. Şekil 2.3'de görüldüğü üzere, önerilen Uzun Kısa-Süreli Bellek temelli ve medyan filtreli tespitçi yapısının -30 dB sinyal gürültü oranı seviyesinde bile 0.9 EAKA (ROC eğrisi altında kalan alan) ile yüksek tespit başarısı elde edilmektedir [10].



Şekil 2.3 Tespitçi yapısının eğri altında kalan değerleri [10]

Şekil 2.4’de önerilen tespitçi yapısının kullanılmasıyla -20 dB sinyal gürültü oranı seviyesinde sinyal tespit çalışması gösterilmektedir.



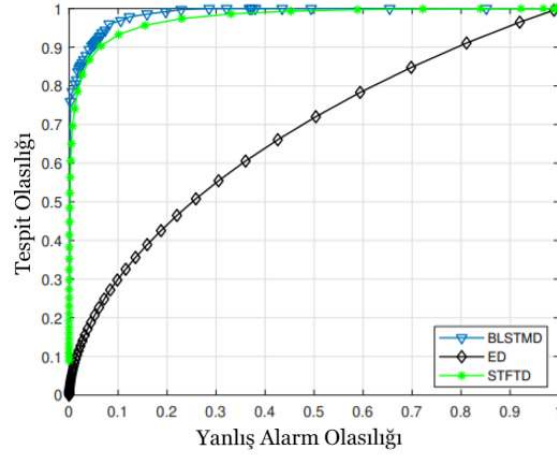
Şekil 2.4 -20 dB SGO’da önerilen tespitçi yapısının kullanılmasıyla gerçekleştirilen sinyal tespiti [10]

Radar sinyallerinin tespiti, elektronik harp pasif sistemleri için önemli bir adımdır. Bu sistemler alınan sinyal hakkında önceden bilgi sahibi olmadıklarından dolayı uyumlu filtre (Matched filter) ve genel olabilirlik oranı testlerinin (Likelihood ratio test) uygulanması çoğu zaman mümkün değildir [15].

Nuhoglu, Alp ve Akyon yaptıkları başka bir çalışmada [15] darbe üzerinde kasıtlı modülasyon (Intentional modulation) kısıtlaması olmaksızın alınan darbeleri otomatik olarak tespit etmek için yeni bir yöntem önermişlerdir. Önerdikleri yöntem, Çift Yönlü Uzun Kısa-Süreli Bellek tabanlı (Bidirectional Long Short-Term Memory, BLSTM) bir gürültü giderici ve otomatik kodlayıcı içeren bir bilişsel algılayıcı kullanır. Bu çalışmada algılama için yeni bir kayıp fonksiyonu da (Loss function) geliştirilmiştir.

Önerilen yöntemin performansını iyi bilinen iki algılayıcı ile karşılaştırılmıştır: Enerji Dedektörü ve Zaman Frekans Alanı Dedektörü (Time-Frequency Domain Detector). Yapılan deneyler, önerilen yöntemin düşük yanlış alarm olasılığı olan bir sinyalin varlığını tespit edebildiğini ve tüm SGO durumlarında diğer yöntemlerden daha iyi performans sergilediğini göstermektedir [15].

Deney sonuçları bu çalışmada [15] önerilen yöntemin, alıcı işletim karakteristik (Receiver Operating Characteristic, ROC) analizine göre diğer yöntemlerden daha iyi performans sergilediğini göstermektedir.



Şekil 2.5 -5 dB SGO'da algılama yöntemlerinin karşılaştırılması [15]

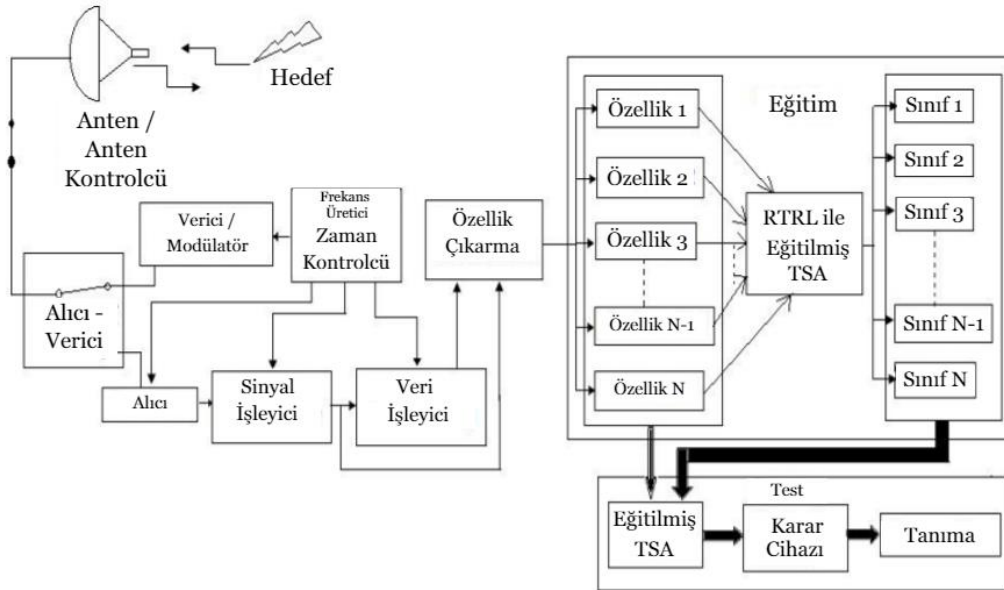
Şekil 2.5, çeşitli algılama dedektörlerin (Short-Time Fourier Transform (STFT), ED, ve BLSTM) -5 dB SGO'daki alıcı işletim karakteristiği analizini göstermektedir. Elektronik destek sistemlerinin (ELINT/ESM) genellikle 10 dB SGO seviyesinde çalıştığı düşünülürse bu düşük bir SGO seviyesidir. Gözleendiği gibi, BLSTM, tüm yanlış alarm seviyeleri olasılığında en yüksek algılama olasılığına sahip olarak diğer dedektörlerden (ED ve STFT) açıkça daha iyi performans gösterir [15].

Sailaja, Sahoo, Panda ve Baghel yaptıkları çalışmada [17], tekrarlayan sinir ağı kullanan darbe tespiti için yeni bir yaklaşım sunmaktadırlar. 13-bit ve 35-bit barker kodları, TSA'ya giriş sinyali kodları olarak kullanılır. Darbesel radar algılama sistemi, TSA kullanılarak simüle edilir.

Simülasyon sonuçları, tekrarlayan sinir ağının otomatik korelasyon işlevi (OKİ) (Autocorrelation Function, ACF) algoritması gibi bazı geleneksel algoritmalarından daha iyi sinyal-yan lob oranı (Signal-to-Sidelobe ratio, SSR) ve Doppler kaydırma performansı sağladığını göstermektedir. Ayrıca TSA tabanlı sistemin, Çok Katmanlı Algılayıcı (ÇKA) (Multilayer Perceptron, MLP) tabanlı sisteme göre daha hızlı yakınsadığı da gözlemlenmiştir. Bu nedenle önerilen TSA, darbeleri radar sinyal tespiti için verimli bir yöntemdir [17].

Radardaki en temel problemten bir tanesi, bir cismin veya fiziksel bir olgunun tespit edilmesidir. Bu, hedef bilgilerini içeren yankının parazit yankılarla (Clutter) çevrelendikten sonra bile, alıcıdaki sinyal ve gürültü içeriği arasında uygun bir ayırım yapılmasını gerektirir. Geleneksel olarak, farklı düzeylerde başarı ile bu ayırımı gerçekleştirmek için bir dizi sinyal işleme işlemi gerçekleştirilir. Bu sinyal işleme operasyonları dizisi, bir Yapay Sinir Ağı (YSA) ile desteklenebilir. TSA, giriş modellerindeki zamansal değişimlerini izleyebilen dinamik bir YSA'dır [18].

Bhattacharyya ve Sarma tarafından yapılan çalışma [18], darbe radarlarından gelen sinyallerin işlem adımlarını açıklar, böylece bunlar, hedef ve yanlış yankıları ayrt etmek için bir TSA'yı eğitmek için kullanılabilir. Şekil 2.6'da önerilen TSA tabanlı model görülmektedir. Deneysel sonuçlar, önerilen sistemin, değişen mesafelerde termal gürültü ve zemin parazit yankıları ile çevrili hedef yankıları ile etkili bir şekilde çalıştığını göstermektedir.



Şekil 2.6 Önerilen TSA tabanlı Otomatik Hedef Tanıma sistem modeli [18]

TSA'lar ile ilgili zorluklar, boyutsallık ve genelleme problemlerini içerir. Radar sinyali gibi çok boyutlu veriyi öğrenmek için ya çok sayıda örnek gerekir ya da TSA'nın öğrenmesi için önce nispeten az sayıda özneliğin çıkarılmasıyla veriler azaltılmalıdır. Mevcut veri örneklerinin sayısı çok küçükse, eğitim verileri ağ için yetersiz olacak ve bu da test verileri üzerinde zayıf genelleme yapılmasına neden olacaktır.

Bu nedenle, bir TSA tasarlamak, belirli bir problem ve ilişkili veri seti için en iyi mimariyi belirlemek için deney yapmayı gerektirir. Elde edilen verilerle sistemi eğitmek için gereken sürenin yüksek olduğu görülmüştür. Bu, sistemin yalnızca bir kez eğitilmesi gerekmesine rağmen, yüksek performanslı hesaplama çerçevesi ile daha da azaltılabilir. Modüler veya dağıtılmış tasarımdaki sistem, mevcut olanlardan daha iyi sonuçlar sağlayacaktır [18].

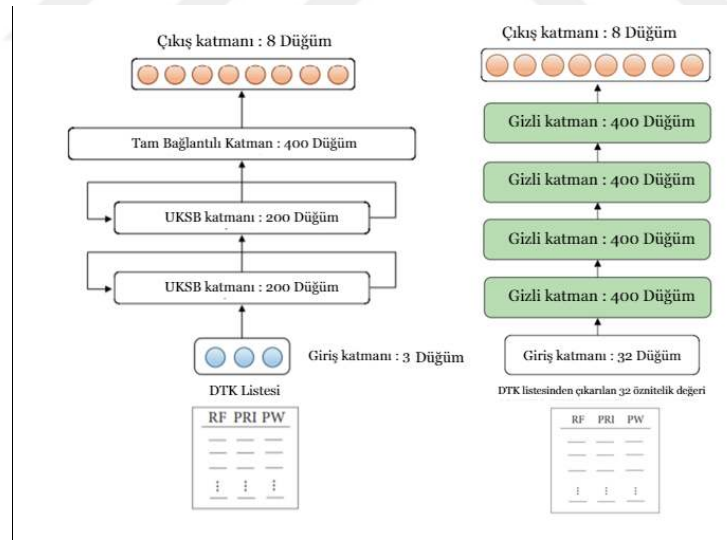
Nguyen, Ngo ve Long Do yaptıkları çalışmada [19], çeşitli gürültü seviyeleri altında değişen dalga biçimlerine ve darbe genişliklerine sahip radar darbelerinin sıralı tespiti için derin öğrenme tabanlı bir çerçeve tanıtmışlardır. Yöntem iki aşamaya ayrılmıştır. İlk aşamada, bir Konvolüsyonel Sinir Ağı eğitilir. İkinci aşamada, bir optimizasyon problemi çözülerek segmentteki değişim noktaları bulunur ve daha sonra darbe konumlarını tahmin etmek için önceden tespit edilen kenarlarla birleştirilir. Önerilen şema, eşik tabanlı kenar algılama (Threshold-based edge detection, TED) yönteminin aksine bir gürültü tabanı tahmini gerektirmez. Benzetimler ayrıca, sunulan yöntemin oldukça gürültülü durumlarda eşik tabanlı kenar algılama yönteminden önemli ölçüde daha iyi performans sağladığını göstermektedir.

Araştırmacılar geniş bir SGO aralığında çeşitli dalga biçimleriyle radar darbelerinin saptanmasına yönelik derin öğrenmeye dayalı bir yaklaşım sunmuşlardır. Önerilen yöntem, sınıflandırma için klasik bir değişim noktalarını bul algoritmasını (Find-change-points algorithm) konvolüsyonel bir sinir ağı ile birleştirir. Simüle edilmiş veriler üzerinde yapılan deneyler, önerilen yöntemin, özellikle düşük SGO seviyelerinde, eşik tabanlı kenar algılama algoritmasından çarpıcı bir şekilde daha iyi performans sergilediğini göstermektedir. Örneğin, önerilen sistem $SGO = 6$ dB seviyesinde, algılama başarı oranını %52,81'den %78,39'a yükseltmektedir [19].

Karıştırma (Jamming), bozucuların düşman radarına karşı sinyaller yayarak alıcıyı bozduğu bir elektronik savaş şeklidir. Bir tehdit sinyaline karşılık gelen etkili bir karıştırma tekniğini belirlemeye yönelik geleneksel yöntem, sinyal türleri için uygun karıştırma yöntemini depolayan kütüphaneye dayanmaktadır. Ancak, yeni türden bir tehdit sinyali veya mevcut türlerden farklı olarak değiştirilmiş bir tehdit sinyali alındığında kütüphanenin kullanımının bir sınırı vardır [20].

Lee, Jo ve Park yaptıkları çalışmada [20], derin öğrenme kullanarak alınan bir tehdit sinyali için uygun karıştırma (jamming) tekniğini tahmin etmenin iki yöntemini incelemişlerdir. Bu yöntemler, Darbe Tanımlama Kelimesi (DTK) (Pulse Description Word, PDW) listesinden manuel olarak çıkarılan öznitelik değerleri üzerinde derin bir sinir ağı kullanmak ve öznitelik çıkarma işlemi olmadan giriş olarak DTK listesi kullanan UKSB kullanmaktır. Şekil 2.7 bu çalışmada bahsedilen DSA ve UKSB model yapılarını göstermektedir.

Derin sinir ağı, önceden öznitelik çıkarmı gerektirir ve bu, derin sinir ağı modelinin eğitiminden önce bir kez gerçekleştirilir. Modelin dönem başına eğitim hızı, UKSB modelinden daha hızlıdır. Ancak, model tahmin performansı DTK listesinden çıkarılan öznitelik değerlerine bağlı olarak değiştiğinden, öznitelik çıkarma yönteminin uygun şekilde seçilmesi gerekmektedir. UKSB modeli, öznitelik değerlerinin çıkarılmasına gerek kalmadan radar sinyalinin DTK listesini doğrudan kullanarak eğitim ve tahmin gerçekleştirebilme avantajlarına sahiptir. UKSB modelinin tahmin doğruluğu, derin sinir ağı modelinden ortalama olarak daha yüksektir. Ancak, eğitim süresinin, çıkarılan öznitelikler üzerinde eğitilmiş bir derin sinir ağı modelinden daha uzun sürmesi dezavantajı vardır [20].

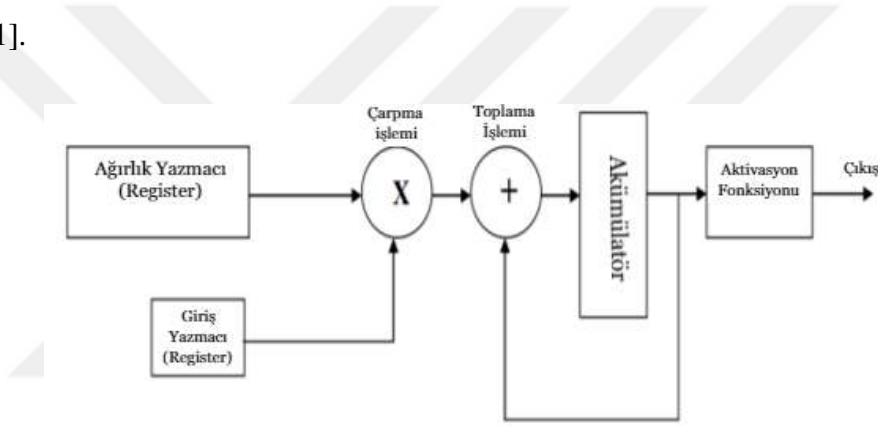


Şekil 2.7 Nihai UKSB model yapısı ve Derin Sinir Ağ model yapısı [20]

Bilinen sinyal tiplerinin testinde öznitelikleri çıkarılan DSA için ortalama doğruluk %98,46 iken UKSB modeli için ortalama doğruluk %99,36 olarak elde edilmiştir. Öznitelikleri çıkarılan DSA kullanırken bilinmeyen türler için ortalama doğruluk %92,45 iken, UKSB modeli için ortalama doğruluk %93,53 olmuştur. Her iki durumda da, UKSB modeli biraz daha yüksek doğruluk göstermiştir [20].

2.2. FPGA’de Yapay Sinir Ağı Uygulamaları

Abrol ve Mahajan tarafından hazırlanan çalışmada [21], Alan Programlanabilir Kapı Dizileri (FPGA) üzerinde bir yapay sinir ağı uygulaması sunulmaktadır. Bir sinir ağına paralel yapısı, belirli görevlerin hesaplanması için potansiyel olarak hızlı olmasını sağlar. Bir sinir ağına donanımsal olarak gerçekleştirilmesi, büyük ölçüde tek bir nöronun verimli bir şekilde uygulanmasına bağlıdır. FPGA tabanlı yeniden yapılandırılabilir bilgi işlem mimarileri, sinir ağlarının donanım uygulaması için uygundur. YSA’lar doğaları gereği paralel hesaplamalar gerektirir. Mikroişlemciler ve DSP’ler paralel tasarımlar için uygun değildir. ASIC’ler ve FPGA’lar tarafından paralel modüller tasarlamak mümkün olabilir, ancak bu tür yongaları geliştirmek pahalı ve zaman alıcıdır. FPGA’lar sadece paralellik sunmakla kalmaz, aynı zamanda esnek tasarımlar, maliyet ve tasarım döngüsünde tasarruf sağlar [21].



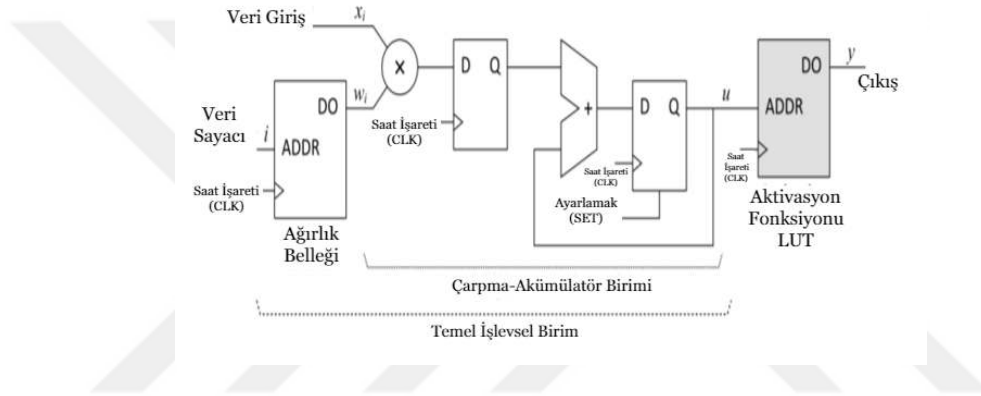
Şekil 2.8 Basit bir nöron mimarisi [21]

Bu [21] çalışma, herhangi bir sinir ağına kullanılacak bir nöronun tasarımını ve uygulamasını sunmaktadır. Nöron içinde tasarlanan aktivasyon fonksiyonu bir sigmoid fonksiyondur. Bu tasarım, yeniden yapılandırılabilir bir cihaz (FPGA) programında şematik düzenleyici araçları kullanılarak gerçekleştirilmiştir. FPGA teknolojisinin düşük kaynak tüketimi ve yeniden programlanabilirliği, YSA’yı uygulamak için çok güçlü bir seçenek haline gelmiştir. Şekil 2.8’de bu çalışmada tasarlanan basit bir nöron mimarisi görülmektedir [21].

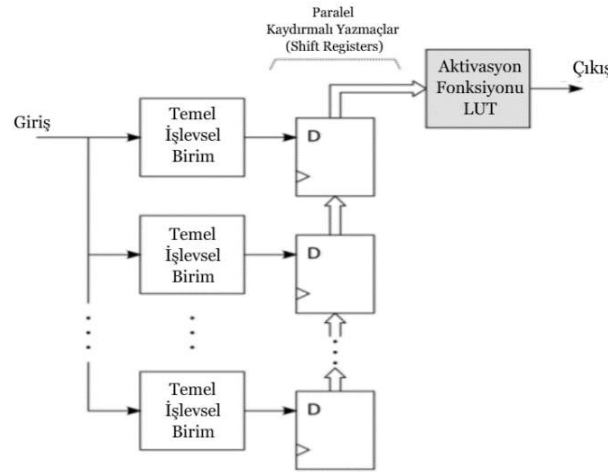
Bir YSA’da çok sayıda yüksek düzeyde birbirine bağlı basit doğrusal olmayan işlem birimi, belirli bir sorunu çözmek için paralel olarak çalışır. Paralellik, modülerlik ve dinamik adaptasyon, tipik olarak YSA’larla ilişkilendirilen üç özelliktir. FPGA tabanlı yeniden yapılandırılabilir bilgi işlem mimarileri, bir YSA’nın ağırlıklarını ve topolojilerini uyarlamak için eşzamanlılıktan yararlanabileceği ve hızla yeniden yapılandırılabileceği için YSA’ları

uygulamak için çok uygundur. YSA'lar çeşitli gerçek yaşam uygulamaları için uygundur ve yaygın olarak kullanılmaktadır. Bu uygulamaların büyük bir kısmı gömülü bilgisayar sistemleri olarak gerçekleştirilmektedir. Teknolojideki sürekli gelişmelerle birlikte, FPGA'lar daha güçlü ve güç açısından daha verimli hale gelmektedir ve YSA'ların FPGA uygulamasını mümkün kılmaktadır [22].

Bu çalışmada [22] araştırmacılar Škoda, Lipić, Srp ve diğerleri, Şekil 2.9 ve Şekil 2.10'de gösterildiği gibi FPGA tabanlı bir YSA çerçevesi önermektedir. Önerilen çerçevenin avantajlarını göstermek için düşüş algılama (Fall detection) sisteminde bir YSA uygulaması çalışması sunmuşlardır.



Şekil 2.9 Nöronun donanım uygulamasının temel yapısı [22]



Şekil 2.10 FPGA tabanlı YSA'da bir katmanın yapısı [22]

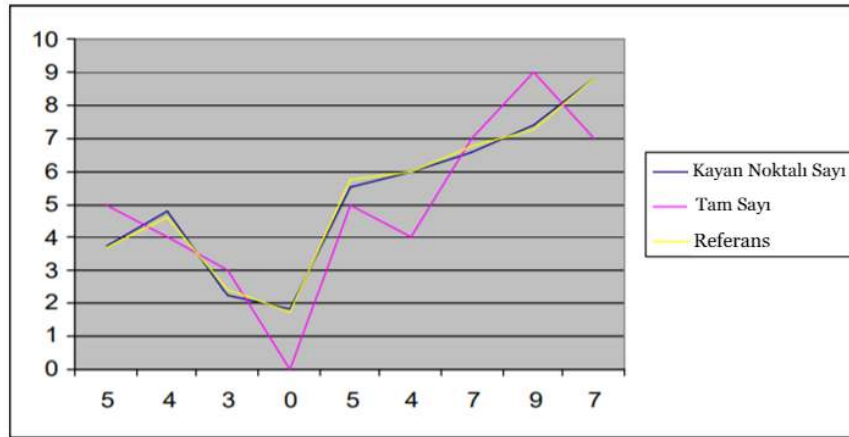
Uygulama sonuçlarını değerlendirmek için, bu [22] çalışmada önerilen yapay sinir ağı çerçevesinin bir katmanı başarıyla sentezlenmiştir ve 85 MHz çalışma frekansı ile FPGA kaynak kullanımının bir özeti Tablo 2.1'de verilmiştir.

Tablo 2.1 FPGA Kaynak Kullanımı [22]

Kaynak Türü	Kullanılabilir Kaynak	Kullanılan Kaynak	Kullanım Yüzdesi
DSP48E	288	176	%61
Flip-flop	32640	2825	%9
LUT	32640	20197	%62

Raeisi ve Kabir yaptıkları çalışmada [23], YSA'ların FPGA uygulamalarında matematiksel işlemlerde kullanılan verilerin, veri türü seçimi ile ilgili çeşitli sonuçlar ortaya koymuşlardır. YSA'larda ağırlıklar hem tamsayı hem de kayan nokta (Floating point) olabilir; ancak, doğru bir YSA gerekiyorsa, ağırlıklar kayan nokta olmalıdır. Şekil 2.11, istenen çıktıya kıyasla ağırlıkları tamsayı ve ağırlıkları kayan nokta olarak gösteren YSA çıktısını göstermektedir.

Şekil 2.11'de gösterildiği gibi, YSA'nın tamsayı ağırlıklı çıktısı hatalı olduğu için kabul edilemez. Ancak, kayan nokta ağırlıklı YSA çıktısı doğrudur ve kayan noktalı ağırlıkların kullanılması gerektiğini kanıtlar. Dijital mantık üzerinde kayan noktalı sayıları uygulamak için farklı algoritmalar mevcuttur. Bunların en ünlülerinden biri IEEE-754 formatıdır [23].



Şekil 2.11 Tam sayı veya kayan nokta kullanan YSA karşılaştırması [23]

Tekrarlayan sinir ağları, konuşma tanıma, makine çevirisi, sahne analizi ve görüntü yazısı oluşturma gibi birçok uygulamada UKSB mimarisinin kullanılmasından elde edilen başarı nedeniyle son zamanlarda popülerlik kazanmıştır. TSA'lar hafızayı tutma ve veri dizilerini öğrenme yeteneğine sahiptir [24].

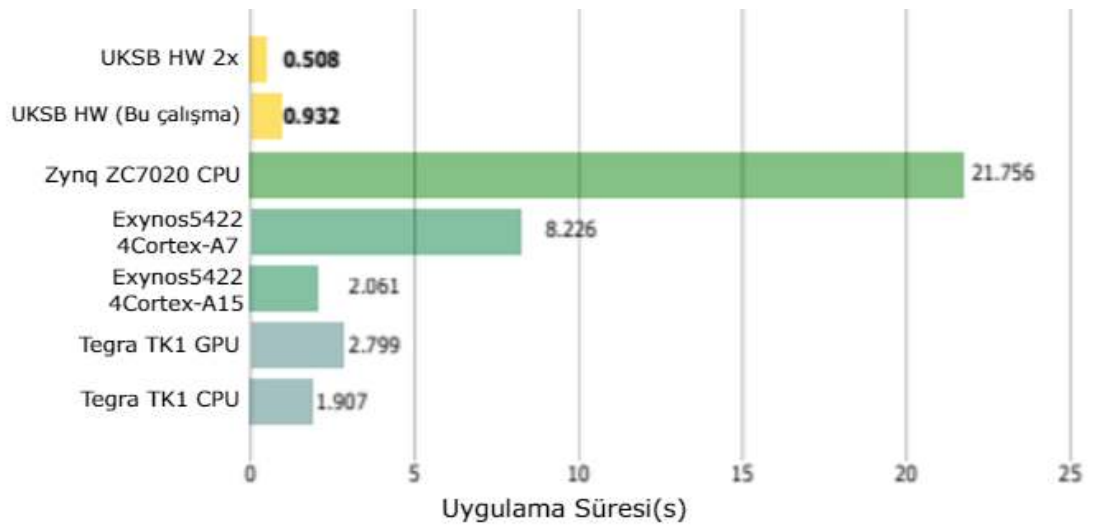
Chang, Martini ve Culurciello yaptıkları bu çalışmada [24], Xilinx Zynq 7020 FPGA üzerinde UKSB tekrarlayan sinir ağının bir donanım uygulamasını sunmuşlardır ve karakter düzeyinde bir dil modeli kullanarak test etmişlerdir. Uygulama, Zynq 7020 FPGA'de yerleşik ARM Cortex-A9 CPU'dan 21 kat daha hızlıdır.

Bu [24] çalışma, UKSB modülünün bir donanım uygulamasını sunarak karakter düzeyinde bir model ile Shakespeare benzeri metni başarıyla üretmiştir. FPGA kaynak kullanımı Tablo 2.2'de gösterilmektedir. Modül 142 MHz'de çalışmaktadır ve üzerindeki toplam güç 1.942 W'tır [24].

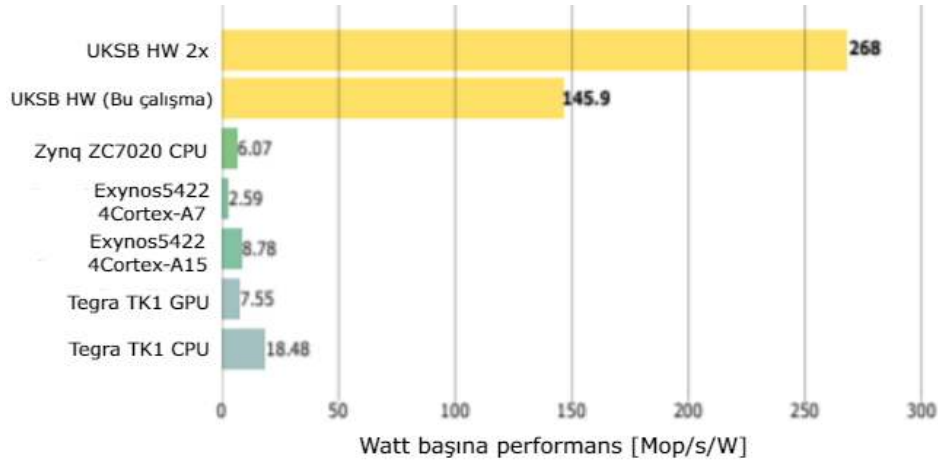
Tablo 2.2 Zynq ZC7020 için FPGA kaynak kullanımı [24]

Kaynak Türü	Kullanılan Kaynak	Kullanım Yüzdesi
Flip-Flop	12960	12.18
LUT	7201	13.54
Memory LUT	426	2.45
BRAM	16	11.43
DSP48	50	22.73
BUFG	1	3.12

Şekil 2.12, zamanlama sonuçlarını göstermektedir. Uygulanan donanım UKSB'nin, 142 MHz'lik daha düşük saat frekansında çalışmasına rağmen, diğer platformlardan önemli ölçüde daha hızlı olduğu gözlemlenmektedir. Ayrıca her platformun güç tüketimi ölçülmüştür ve Şekil 2.13, birim başına güç performansını göstermektedir [24].



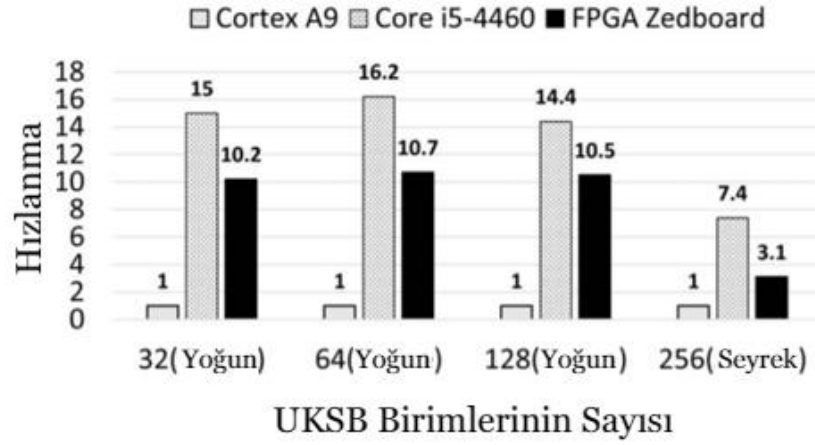
Şekil 2.12 Farklı platformlarda UKSB modelinin yürütme süresi (ne kadar düşükse o kadar iyidir) [24]



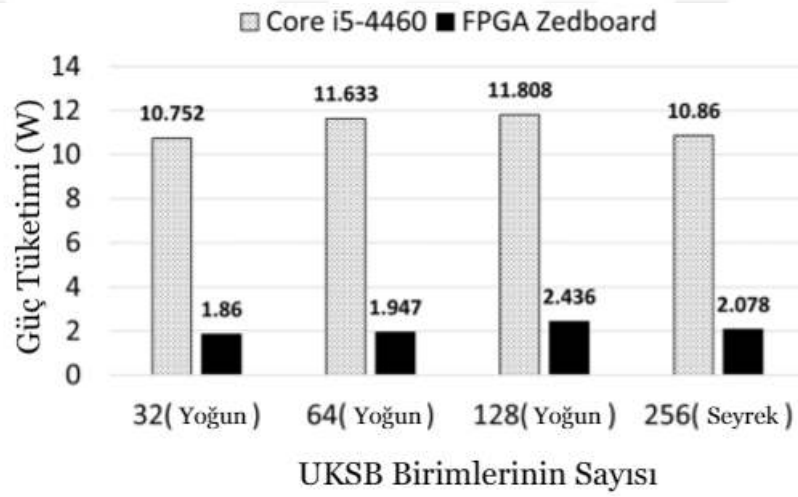
Şekil 2.13 Farklı platformların birim güç başına performansı (ne kadar yüksekse o kadar iyi) [24]

Rashid, Manjunath, Paneliya ve diğerleri yaptıkları bu çalışmada [25], çok kanallı beyin EEG sinyallerinde çeşitli yapay bozuklukları tespit etmek için UKSB tabanlı bir sinir ağı mimarisi önermişlerdir. Önerilen donanım çözümü Verilog HDL'de tasarlanmış, Artix-7 FPGA üzerinde sentezlenmiş/uygulanmış ve 52,6 MHz çalışma frekansında 17 mW güç tüketmektedir. Önerilen yöntem, ikili sınıflandırma doğruluğu açısından Evrimsel Sinir Ağı ve Otoregresif (Auto Regressive, AR) modellerine dayalı diğer tekniklerden ortalama %93,1'lik ikili yapı algılama doğruluğu ile daha iyi bir performans elde etmektedir. Deneysel sonuçlar, FPGA uygulamasının, güç tüketimi açısından ARM CPU'nun en verimli konfigürasyonundan 13,5 kat daha iyi performans sağladığını göstermektedir [25].

Zhang, Wang, Gong ve diğerleri yaptıkları çalışmada [26], UKSB sinir ağı katmanı için FPGA tabanlı bir donanım hızlandırıcı mimarisi sunmuşlardır ve ileri hesaplama sürecini paralelleştirmek için boru hattı (Pipeline) yöntemini kullanmışlardır. Ayrıca, yoğun ağdan daha az depolama kaynağı tüketen seyrek bir UKSB gizli katmanı uygulamışlardır. Hızlandırma performansı Şekil 2.14'de görüldüğü gibi Cortex-A9'dan on kat daha hızlı ancak Core i5'ten biraz daha yavaştır. Aynı yoğun UKSB sinir ağı katmanını çalıştırırken, önerilen FPGA tabanlı hızlandırıcı, ARM Cortex-A9 işlemci ve Core-i5 işlemciden Şekil 2.15'de görüldüğü gibi güç açısından daha verimlidir [26].



Şekil 2.14 Hızlandırma (daha yüksek daha iyidir) [26]



Şekil 2.15 Güç Tüketimi (daha düşük daha iyidir) [26]

UKSB tekrarlayan sinir ağıları veri anormallik tespitinde de iyi bir performans gösterirler. UKSB TSA'nın tekrarlanabilirliği nedeniyle, genel amaçlı işlemciler UKSB TSA'yı verimli bir şekilde uygulayamazlar. FPGA üzerindeki mevcut model optimizasyonlarının çoğu, UKSB hücrelerine veya büyük ölçekli model hızlandırmalarına yöneliktir. Kısa veri örnekleme aralıkları, yüksek hız gereksinimleri ve yüksek hassasiyet gereksinimleri olan uçak anormallik algılama modeli için mevcut modellerin doğruluğu ve hızı yetersizdir [27].

Sun, Zhu, Zheng ve diğerkleri yaptıkları alıřmada [27], mevcut modellerin doėruluėunu ve hızını optimize etmek iin FPGA tabanlı bir UKSB TSA hızlandırıcı oėermişlerdir. Bu alıřma doėruluktan dn vermeden hesaplama hızında optimizasyonu saėlarken, FPGA'de kullanılan kaynakları ve performansı dengelemektedir. oėerilen hızlandırıcının en yüksek performansı, mevcut diėer yöntemlerden daha stn olan 13.45 GOP/s'ye ulaşmaktadır.

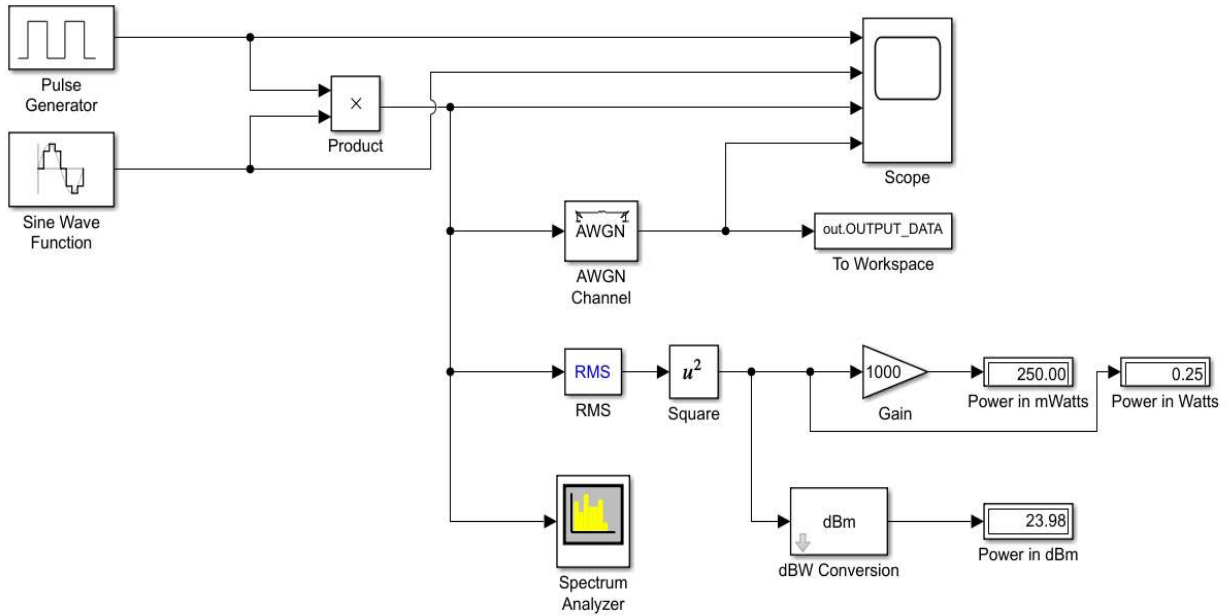
Liu, Liang, Huang ve Dinavahi tarafından gerekleřtirilen alıřmada [28], Daha Fazla Elektrik Enerjisi Kullanan Uak (More Electric Aircraft, MEA) zerindeki gerek zamanlı arıza tespiti ve izolasyonu ile bařa ıkmak iin, iki ařamalı bir FPGA tabanlı sınır aėı yöntemi ortaya koyulmuřtur: TensorFlow kullanılarak evrimdiřı yapı ve FPGA zerinde gerek zamanlı izleme. MEA'nın durumunu gerek zamanlı olarak izleme hedefine ulaşmak iin, FPGA zerinde UKSB tabanlı donanım uygulaması oėerilmiştir [28].

3. TASARIM ÇALIŞMALARI

Bu tez çalışması ile düşük SGO seviyesine sahip darbesel sinyallerin, FPGA tabanlı bir Uzun Kısa-Süreli Bellek tekrarlayan sinir ağı mimarisi ile tespit işlemlerine ait çalışmalar sunulmaktadır. Benzetim sonuçlarına ek olarak bu çalışmanın FPGA üzerinde sentezlenmesi sonucu elde edilen performans ölçütleri (FPGA kaynak tüketimi ve çalışma frekansı) paylaşılmaktadır. Ayrıca önerilen FPGA tabanlı UKSB çalışmasının SYAO yöntemi ile performans ölçütleri açısından karşılaştırılması amacıyla SYAO yöntemi de FPGA’de tasarlanmış, FPGA benzetim çalışmaları yapılmış ve performans ölçütleri paylaşılmıştır. Bu tez aşamasında gerçekleştirilen tasarım çalışmaları 7 bölümden oluşmaktadır.

3.1. MATLAB/Simulink Modeli İle Eğitim ve Test Veri Setlerinin Elde Edilmesi

Tez çalışmasının ilk aşaması UKSB sinir ağını eğitmek ve eğitilen UKSB sinir ağı ve SYAO yöntemini test edebilmek için veri setleri elde etmektir. Bu amaçla MATLAB/Simulink ile Şekil 3.1’de gösterilen Simulink modeli oluşturulmuştur.

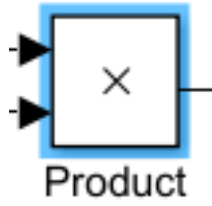


Şekil 3.1 Genel MATLAB/Simulink Modeli

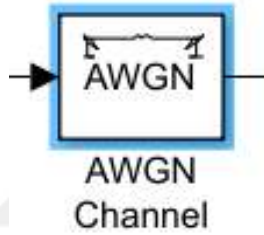
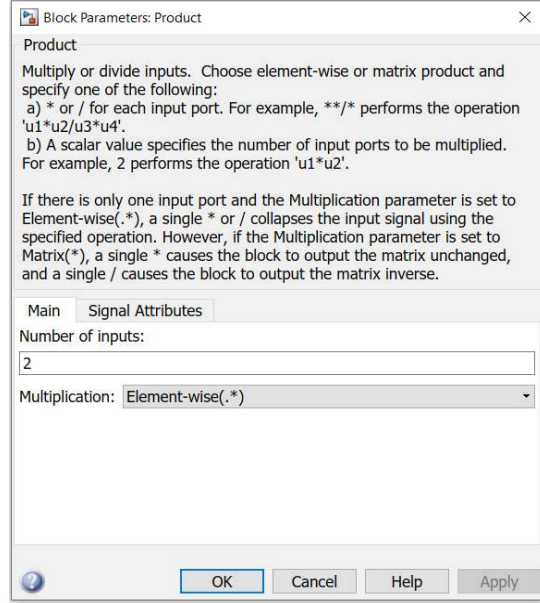
MATLAB/Simulink modeli kullanılarak istenen SGO seviyesinde gürültü eklenebilen taşıyıcı frekans ile modüle edilmiş darbesel sinyaller üretilebilmektedir. Üretilen bu darbesel sinyaller örneklenerek eğitim ve test veri setleri elde edilmektedir. Ayrıca üretilen darbesel sinyallerin gücü ölçülmektedir. MATLAB/Simulink modelinde kullanılan bloklar ve bu blokların parametreleri Tablo 3.1’de gösterilmiştir.

Tablo 3.1 MATLAB/Simulink Modelinde Kullanılan Bloklar ve Blok Parametreleri

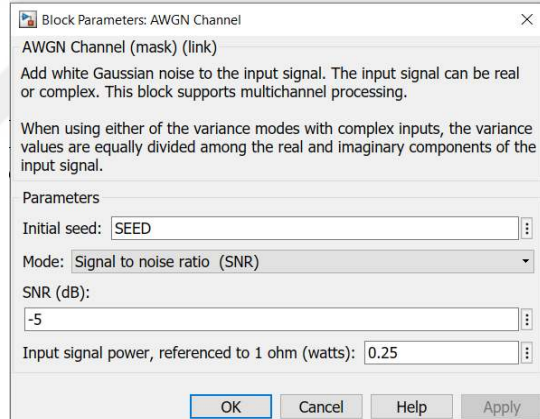
Pulse Generator Yan sütunda blok parametreleri verilen <i>Pulse Generator</i> bloğu istenen özelliklerde darbesel sinyaller üretmek için kullanılmaktadır.	Block Parameters: Pulse Generator Pulse Generator Output pulses: <pre> if (t >= PhaseDelay) && Pulse is on Y(t) = Amplitude else Y(t) = 0 end </pre> Pulse type determines the computational technique used. Time-based is recommended for use with a variable step solver, while Sample-based is recommended for use with a fixed step solver or within a discrete portion of a model using a variable step solver. Parameters Pulse type: Time based Time (t): Use simulation time Amplitude: 1 Period (secs): 1e-3 Pulse Width (% of period): 2 Phase delay (secs): 10e-6 ☒ Interpret vector parameters as 1-D OK Cancel Help Apply
Sine Wave Function Yan sütunda blok parametreleri verilen <i>Sin Wave Function</i> bloğu istenen özelliklerde taşıyıcı sinyal üretmek için kullanılmaktadır.	Block Parameters: Sine Wave Function Sine Wave Output a sine wave: $O(t) = \text{Amp} * \sin(\text{Freq} * t + \text{Phase}) + \text{Bias}$ Sine type determines the computational technique used. The parameters in the two types are related through: $\text{Samples per period} = 2 * \pi / (\text{Frequency} * \text{Sample time})$ $\text{Number of offset samples} = \text{Phase} * \text{Samples per period} / (2 * \pi)$ Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur. Parameters Sine type: Time based Time (t): Use simulation time Amplitude: 1 Bias: 0 Frequency (rad/sec): $F_c * 2 * \pi$ Phase (rad): 0 Sample time: Tsample ☒ Interpret vector parameters as 1-D OK Cancel Help Apply

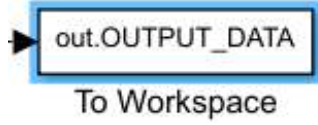


Yan sütunda blok parametreleri verilen *Product* bloğu üretilen taşıyıcı sinyal ile darbesel sinyali birleştirmek (modülasyon) için kullanılmaktadır.

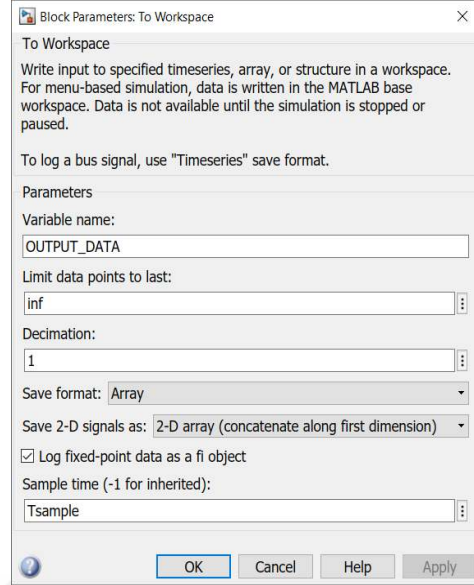


Yan sütunda blok parametreleri verilen *AWGN* bloğu modüle edilmiş darbesel sinyale istenen SGO seviyesinde Gaussian gürültüsü eklemek için kullanılmaktadır. Farklı *SEED* değerleri ile farklı gürültüler elde edilmektedir.

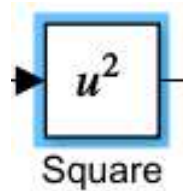
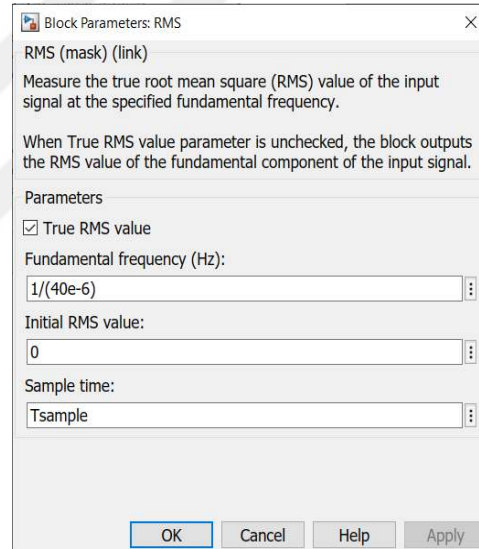




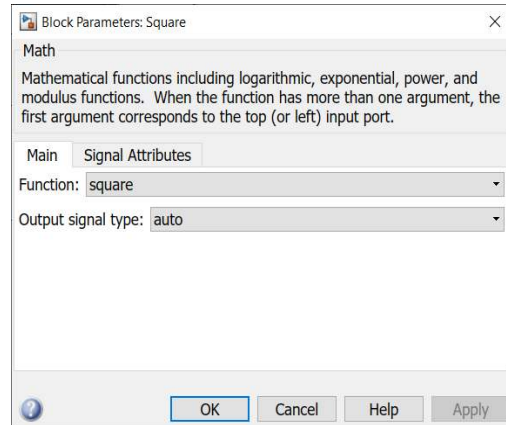
Yan sütunda blok parametreleri verilen *To Workspace* bloğu modüle edilmiş ve gürültü eklenmiş darbesel sinyalleri istenen örnekleme süresiyle örnekleme ve MATLAB/Workspace alanında değişken olarak kaydetmektedir.

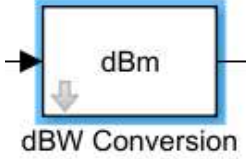


Yan sütunda blok parametreleri verilen *RMS* bloğu modüle edilmiş darbesel sinyallerin RMS değerini hesaplar.

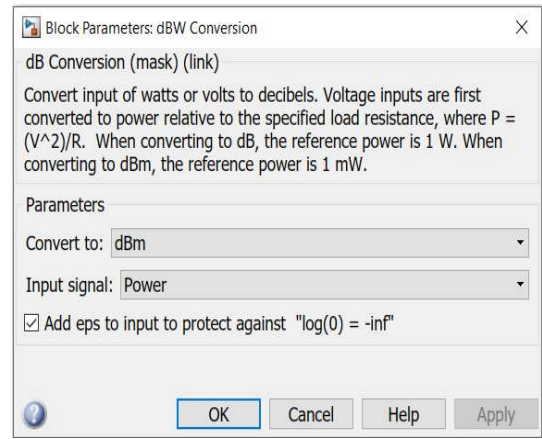


Yan sütunda blok parametreleri verilen *Square* bloğu modüle edilmiş darbesel sinyallerin RMS değerinin karesini hesaplar. Bu değer sinyalin gücünü hesaplamak için kullanılmaktadır.

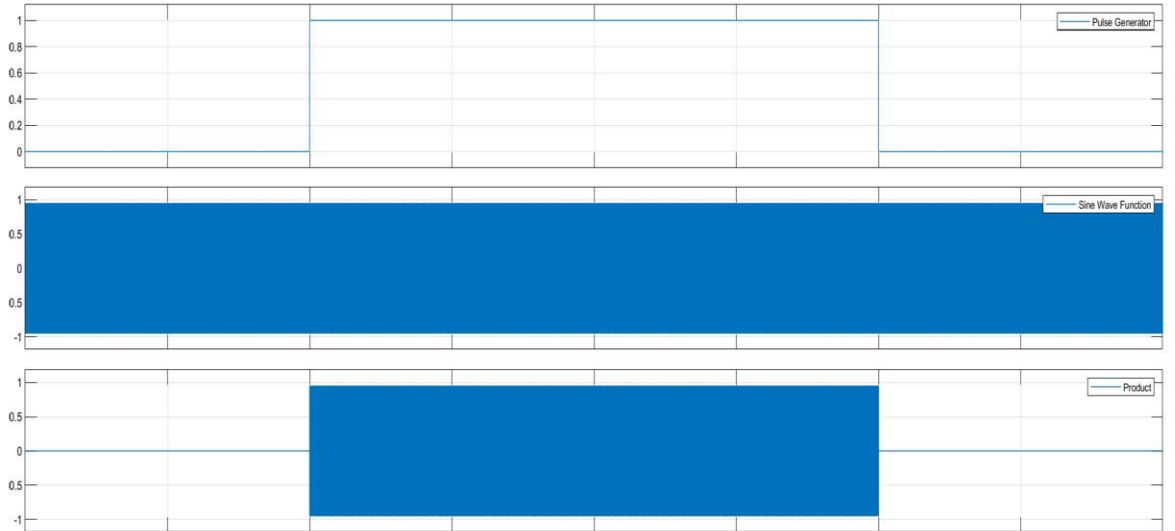




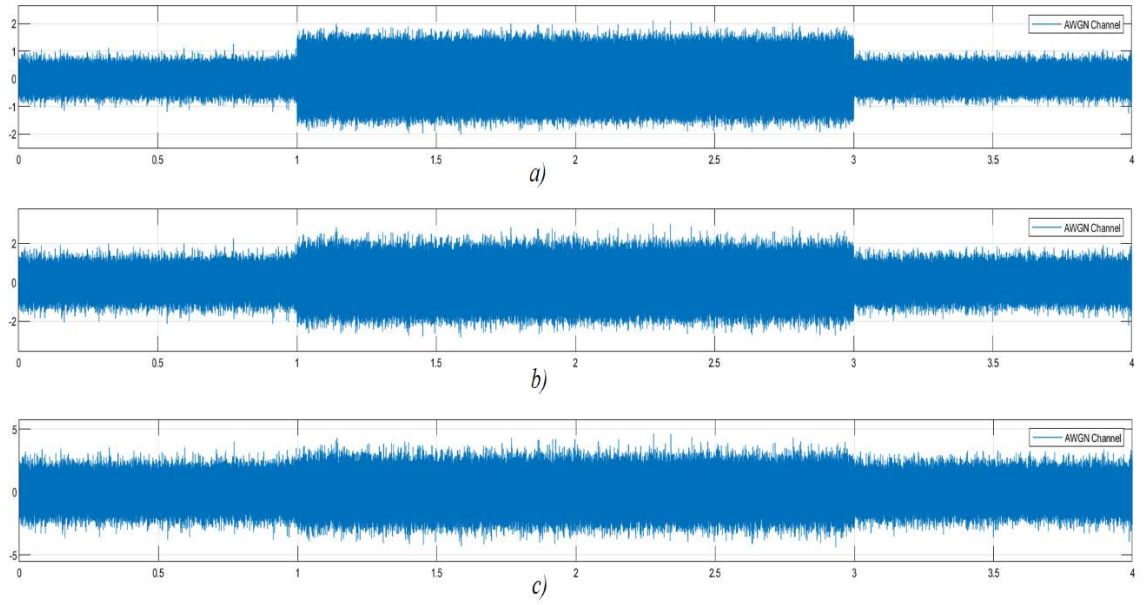
Yan sütunda blok parametreleri verilen *dB Conversion* bloğu modüle edilmiş darbesel sinyallerin gücünü Watt cinsinden dBm cinsine çevirir. Elde edilen bu güç değeri Spektrum Analizör bloğu tarafından ölçülen güç değeri ile karşılaştırılır.



Şekil 3.2 ve Şekil 3.3’de örnek olarak 5 dB, 0 dB ve -5 dB SGO seviyelerinde oluşturulan darbesel sinyallerin Scope görüntüsü verilmiştir. Scope’un 1. girişi üretilen darbesel sinyal, 2. girişi üretilen taşıyıcı sinyal, 3. girişi modüle edilmiş darbesel sinyalleri ve 4. girişi modüle edilmiş ve gürültü eklenmiş darbesel sinyalleri göstermektedir.



Şekil 3.2 Darbesel sinyal, Taşıyıcı sinyal ve Modüle edilmiş darbesel sinyal Scope görüntüsü



Şekil 3.3 Modüle edilmiş ve gürültü eklenmiş darbesel sinyallerin Scope görüntüsü a) 5 dB b) 0 dB c) -5 dB

Modüle edilmiş sinyalin gücünü ölçmek için iki farklı yöntem uygulanmıştır:

Tablo 3.2 1. Yöntemde görüldüğü gibi çeşitli MATLAB/Simulink blokları kullanılarak ve 3.1 eşitliği yardımıyla sinyalin gücü hesaplanmaktadır.

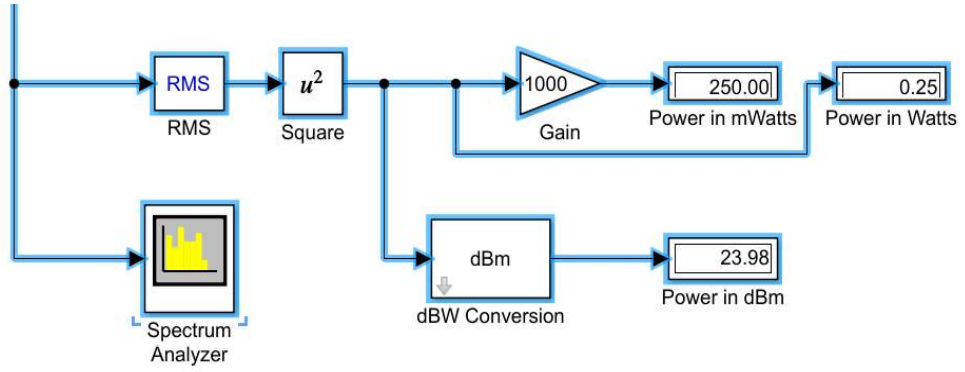
$$P = \frac{V_{RMS}^2}{R} \rightarrow P = V_{RMS}^2 \quad (3.1)$$

(AWGN bloğu giriş sinyal gücü parametresi için $R = 1$ ohm olarak referans alınmıştır.)

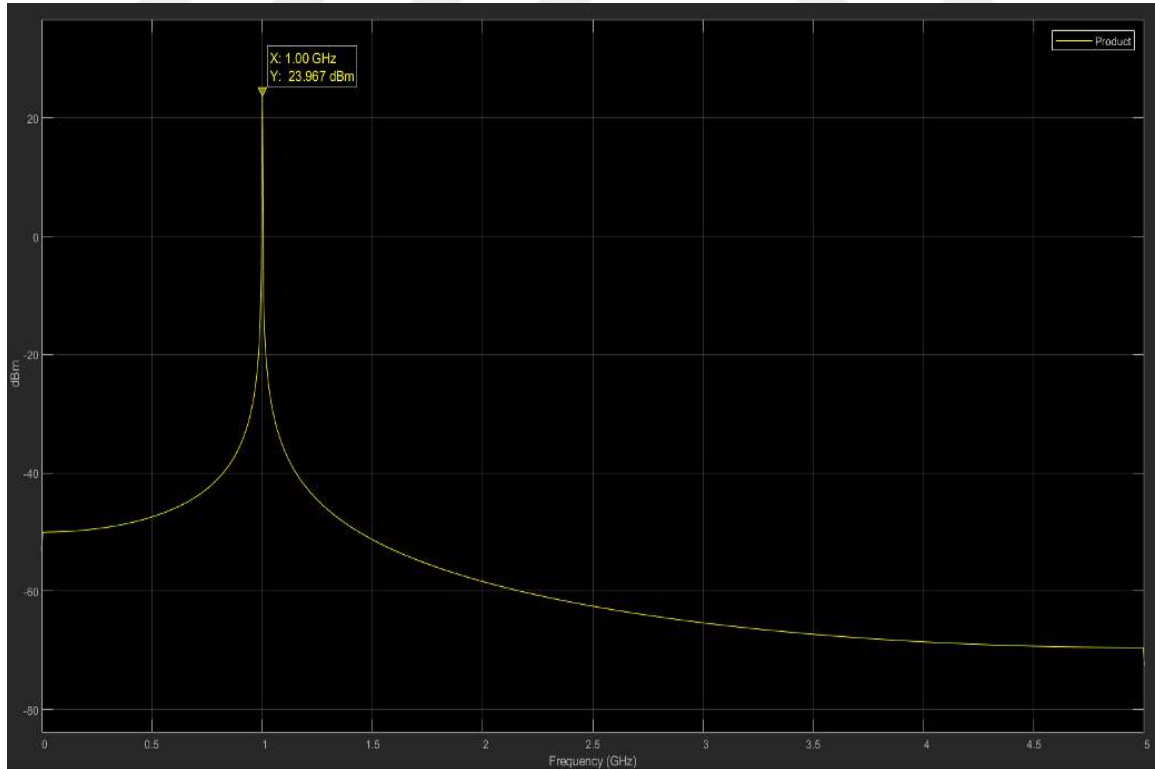
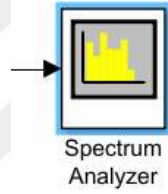
Tablo 3.2 2. Yöntemde görüldüğü gibi Spektrum Analizör bloğu kullanılarak sinyalin gücü ölçülmektedir ve hesaplanan sonuçlar karşılaştırılmaktadır. Elde edilen bu güç bilgisi AWGN bloğunda giriş sinyal gücü (Input signal power) parametresi olarak kullanılmaktadır.

Tablo 3.2 Modüle edilmiş darbesel sinyalin gücünün hesaplanması

1.Yöntem:



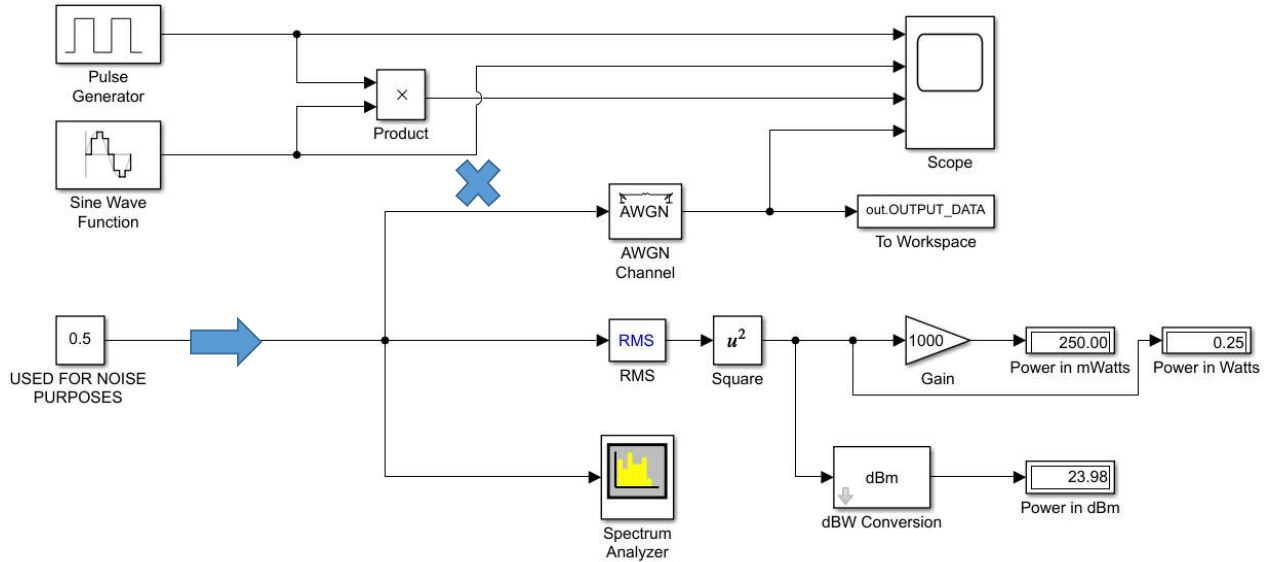
2.Yöntem:



Tablo 3.1’de belirtildiği gibi AWGN bloğundaki SEED değeri değiştirilerek farklı gürültüler oluşturulabilmektedir ve böylelikle farklı gürültüye sahip darbesel sinyaller elde edilebilmektedir. Şekil 3.1’de paylaşılan MATLAB/Simulink modeli ile modüle edilmiş ve gürültü eklenmiş darbesel sinyaller örneklenerek MATLAB/Workspace’de değişken olarak kaydedilmektedir. Örnekleme sayısı kadar elde edilen bu değişkenler, eğitim ve test verilerini oluşturmaktadır. Birden fazla eğitim/test verisi ise eğitim/test veri setlerini oluşturmaktadır. Bu veri setleri UKSB sinir ağı eğitmek ve UKSB sinir ağı ile SYAO yöntemini test etmek için kullanılmaktadır.

UKSB yapay sinir ağı eğitim veri setlerini oluşturabilmek için öncelikle 5 dB, 0 dB ve -5 dB SGO seviyelerinde 100’er adet farklı SEED değerleri ile farklı gürültülere sahip darbesel sinyaller oluşturulmuştur ve bu sinyaller örneklenerek veriler elde edilmiştir. Bu veriler modüle edilmiş darbesel sinyal ve gürültü içermektedir.

Daha sonra Şekil 3.4’de görüldüğü gibi modüle edilmiş darbesel sinyal bağlantısı sistemden kesilmiştir ve sadece 0.5 değerinde sabit bir sinyal (0.25 W güç elde etmek için) uygulanmıştır. Benzer şekilde 100’er adet farklı SEED değerleri ile farklı gürültülere sahip sadece gürültü içeren sinyaller oluşturulmuştur ve bu sinyaller örneklenerek veriler elde edilmiştir. Bu veriler sadece gürültü içermektedir.



Şekil 3.4 Sadece gürültü sinyali elde etmek için oluşturulan MATLAB/Simulink Modeli

Sonuç olarak Tablo 3.3'te gösterildiği gibi UKSB sinir ağını eğitmek için 5 dB, 0 dB ve -5 dB SGO seviyelerinde 200'er adet veri içeren veri setleri oluşturulmuştur. Bu veri setlerindeki verilerin 100 tanesi darbesel sinyal içerirken (sınıflandırma sonucu '1') diğer 100 tanesi sadece gürültü (sınıflandırma sonucu '0') içermektedir. Her bir veri 400.000 adet örneklemeden oluşmaktadır. Örneklenen sinyalin toplam süresi 400 us'dir (Darbe Genişliği 20 us). Bu örnekleme sayısının değeri, sinyalin toplam süresi boyunca 1 GSPS ile örneklenmesiyle elde edilir.

Benzer şekilde Şekil 3.1'da ve Şekil 3.4'da belirtilen modeller kullanılarak, Tablo 3.3'te gösterildiği gibi 5 dB, 0 dB ve -5 dB SGO seviyelerinde farklı SEED değerleri ile 100'er adet veri içeren test amaçlı veri setleri de oluşturulmuştur. UKSB sinir ağı test çalışması için elde edilen bu veri setlerindeki veriler de 400.000 adet örneklemeden oluşmaktadır.

Ayrıca Şekil 3.1'da belirtilen Simulink modeli ile SYAO yönteminin FPGA uygulamasında kullanılmak üzere Tablo 3.3'te gösterildiği gibi 10 dB, 5 dB, 0 dB ve -5 dB SGO seviyelerinde darbesel sinyaller oluşturulmuş ve 100 MHz örnekleme hızı ile örneklenerek SYAO yöntemi için test veri setleri oluşturulmuştur. UKSB sinir ağı çalışmasına ek olarak SYAO yöntemi için 10 dB SGO seviyesinde de test veri seti oluşturulmuştur. Bunun sebebi SYAO yöntemi ile düşük SGO seviyelerine karşılık yüksek SGO seviyelerinde başarılı sonuçlar elde edilebileceğini göstermektir.

Tablo 3.3 MATLAB/Simulink Modelleri ile Oluşturulan Eğitim ve Test Veri Setleri

<u>Sinyal Gürültü Oranı Seviyesi</u>	<u>UKSB Sinir Ağı Eğitim Veri Sayısı</u>			<u>UKSB Sinir Ağı Test Veri Sayısı</u>			<u>SYAO Yöntemi Test Veri Sayısı (Darbesel Sinyal Mevcut)</u>
	Darbesel Sinyal Mevcut	Darbesel Sinyal Mevcut Değil	Toplam Veri Sayısı	Darbesel Sinyal Mevcut	Darbesel Sinyal Mevcut Değil	Toplam Veri Sayısı	
-5 dB	100	100	200	50	50	100	50
0 dB	100	100	200	50	50	100	50
5 dB	100	100	200	50	50	100	50
10 dB	-			-			50

MATLAB/Simulink modeli ile elde edilen bu test verileri SYAO yönteminin FPGA uygulamasında kullanılmak üzere 3.3 eşitliği (3.2 eşitliğinden elde edilmiştir) yardımıyla -/+ 5V, 16-bit ADC çıkış formatına dönüştürülmüştür.

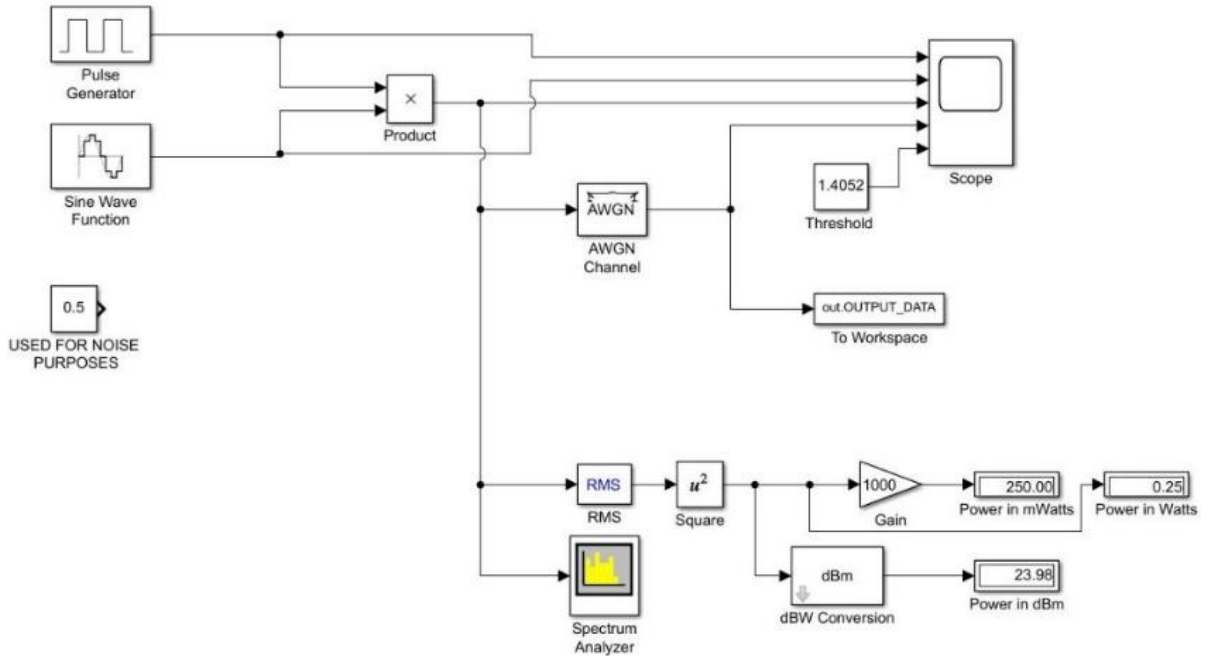
Baz alınan bu ADC’de -5 V, 0 V ve 5 V analog gerilim değerleri onaltılık sistemde sırasıyla “0000”, ”8000” ve “FFFF” dijital değerleri ile ifade edilmektedir. Her bir bit 0.000153 V ($10V/2^{16}$) temsil etmektedir. ADC transfer fonksiyonu 3.2 eşitliği ile gösterilmiştir.

$$Gerilim(V) = \frac{10 V * ADC \text{ çıkış değeri}}{65536} - 5 V \quad (3.2)$$

$$ADC \text{ Çıkış Değeri} = \frac{(Matlab \text{ ile örneklenen gerilim değeri (V)} + 5 V) * 65536}{10 V} \quad (3.3)$$

3.2. MATLAB ile SYAO Darbesel Sinyal Tespit Yöntemi Çalışmaları

Düşük SGO seviyelerinde sabit eşik değerli SYAO yöntemi ile darbesel sinyallerin tespit edilebilmesi amacıyla MATLAB ile Şekil 3.5’da gösterildiği gibi bir çalışma gerçekleştirilmiştir. Bu çalışmada bir MATLAB/Simulink modeli kullanılmıştır. Çeşitli SGO seviyelerinde (10 dB, 5 dB, 0 dB ve -5 dB SGO) darbesel sinyal mevcut değilken sadece gürültü sinyalinden örnekler toplanmıştır. Bu çalışmada 1024 adet örnek toplanmıştır.



Şekil 3.5 Eşik değeri yöntemi için hazırlanan MATLAB/Simulink modeli

Test edilecek SGO seviyelerinde toplanan bu gürültü örneklerinin ortalama değeri ve standart sapma değeri çevrindiği (offline) olarak hesaplanmıştır. Hesaplanan ortalama ve standart sapma değerleri toplanarak Şekil 3.6’de gösterildiği gibi *eşik değerleri* elde edilmiştir.

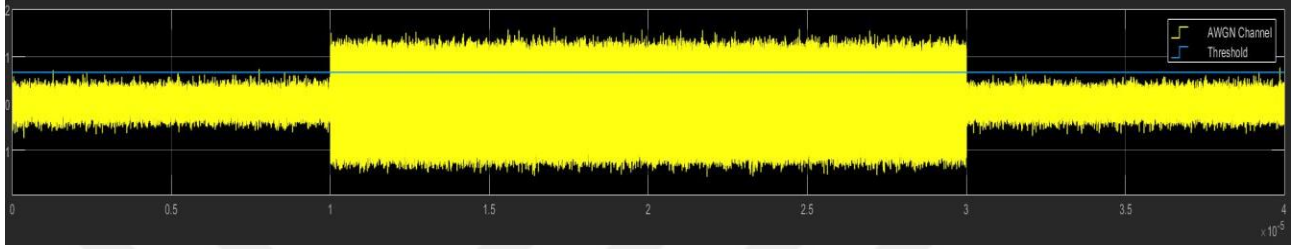
	A	B	C	D	E
1	ÖRNEKLENEN GÜRÜLTÜ VERİLERİ				
2	0.481351088				
3	1.66723983				
4	0.221183652				
5	0.002807337				
6	-0.79015039				
7	0.176030687				
8	0.603601726				
9	0.229676948				
10	0.087745598				
11	0.301988489				
12	1.446421777				
13	0.52674658		ORTALAMA	0.5297	
14	0.214365045		Standart Sapma	0.8755	
15	0.560124836		EŞİK DEĞERİ	1.4052	
16	1.370473695				
17	0.329036623				
18	0.47928089				
19	1.200586713				
20	-0.88020936				
21	1.113973215				
22	0.318379135				
23	1.11884408				
24	2.646828101				
25	-0.13290789				
26	1.722459075				
27	-0.00748072				
28	0.800682619				
29	0.403821461				
30	1.619427642				
31	-0.30260917				
32	-0.25375571				
33	0.76078882				
34	0.609423801				
35	1.246409509				
36	0.52176988				
37	0.481926759				
38	0.927137089				
39	0.870949983				
40	-0.29604648				

Şekil 3.6 Eşik değeri yöntemi ile sinyal tespiti için eşik değerinin hesaplanması örneği

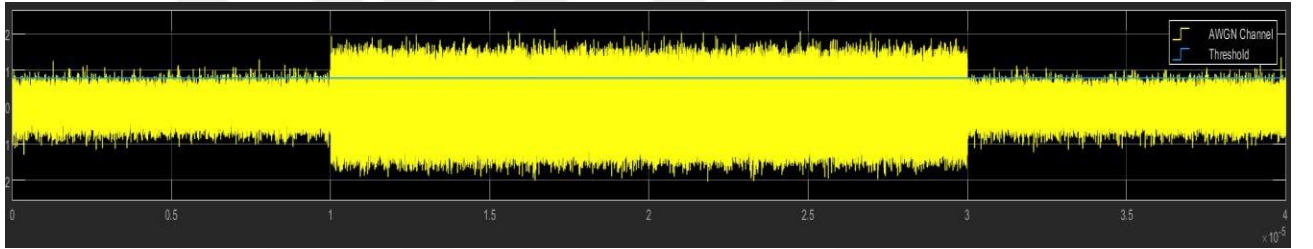
Test edilecek her SGO seviyesi (10 dB, 5 dB, 0 dB ve -5 dB) için elde edilen bu eşik değerleri aynı SGO seviyelerinde darbesel sinyal içeren test verilerine uygulanmıştır. Bu çalışmada beklenen sonuç eşik değerinin üzerinde kalan noktalarda ‘sinyal mevcut’ tespitinin yapılmasıdır. Şekil 3.7, Şekil 3.8, Şekil 3.9 ve Şekil 3.10’da çeşitli örnekler paylaşılmıştır. Şekil 3.7 ve Şekil 3.8’de SGO seviyesi 10 dB ve 5 dB iken bu yöntemin başarılı sonuçlar verdiği gözlemlenmiştir. Ancak Şekil 3.9 ve Şekil 3.10’da görüldüğü üzere SGO seviyesi 0 dB ve -5 dB iken bu yöntemin başarılı olmadığı gözlemlenmiştir.

10 dB ve 5 dB SGO seviyelerinde çok büyük çoğunlukla darbenin olduğu noktalarda eşik değeri üzerine çıktığı ve geri kalan noktalarda eşik değeri altında kaldığı için darbesel sinyal tespiti kolaylıkla yapılabilir.

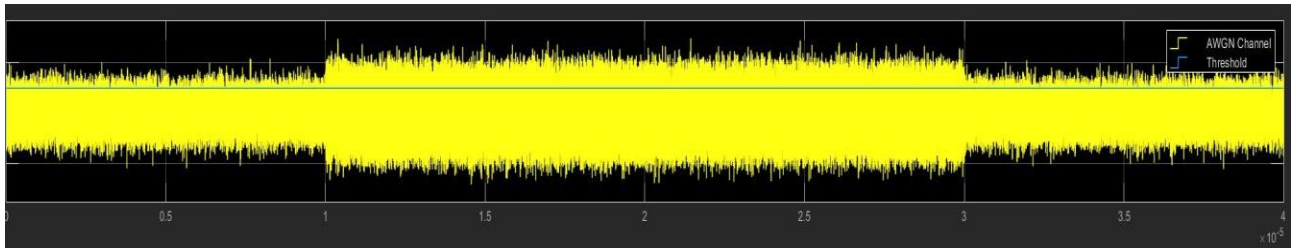
0 dB ve -5 dB seviyesinde sadece darbenin olduğu noktalarda değil geri kalan noktalarda da eşik değerinin üzerine çıktığı için darbesel sinyal tespiti için başarılı bir sonuç elde edilememektedir.



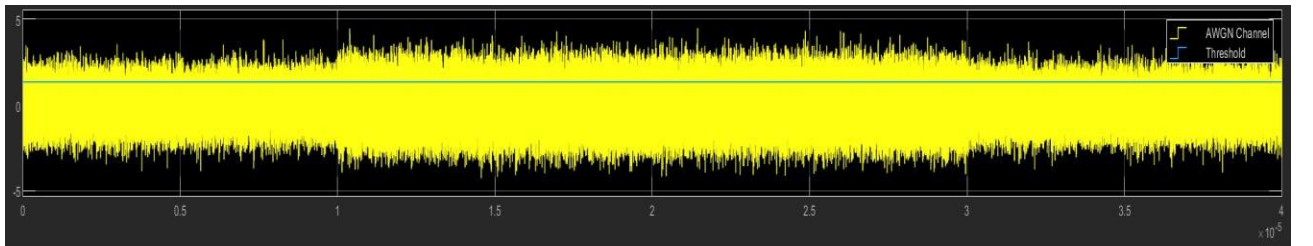
Şekil 3.7 10 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması



Şekil 3.8 5 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması



Şekil 3.9 0 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması



Şekil 3.10 -5 dB SGO seviyesinde eşik değeri yöntemi ile sinyal tespit çalışması

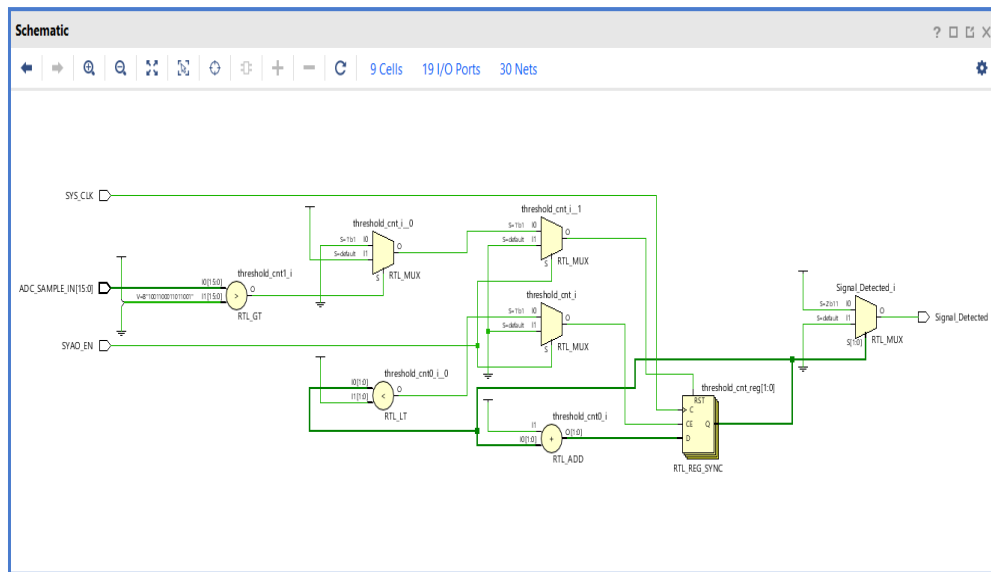
3.3. SYAO Darbesel Sinyal Tespit Yöntemi FPGA Tasarım Çalışması

SYAO yöntemi ile darbesel sinyallerin tespit edilmesi ve UKSB sınır ağıyla performans ölçütleri açısından karşılaştırma yapılabilmesi amacıyla FPGA üzerinde Şekil 3.11’de görüldüğü gibi VHDL donanım tanımla dili kullanılarak bir tasarım gerçekleştirilmiştir. Bölüm 3.2’de belirtildiği gibi çeşitli SGO seviyelerinde sadece gürültü içeren sinyaller üzerinden çevrimdışı olarak eşik değeri seviyeleri elde edilmiştir. MATLAB/Simulink modeli ile çeşitli SGO seviyelerinde 100 MHz örnekleme hızıyla elde edilen test verileri FPGA’de tasarlanan SYAO darbesel sinyal tespit tasarımına uygulanmaktadır.

```
1  -- ERDOGAN BERKAY TEKINCAN --  
2  -- Tez Calismasi 2021 --  
3  -- SYAO Signal Detection --  
4  -----  
5  
6  library IEEE;  
7  use IEEE.STD_LOGIC_1164.ALL;  
8  use IEEE.numeric_std.all;  
9  -----  
10 entity SYAO_Signal_Detection is  
11 port  
12 (   SYS_CLK      : in  std_logic;  
13     SYAO_EN      : in  std_logic;  
14     ADC_SAMPLE_IN : in  std_logic_vector(15 downto 0);  
15     Signal_Detected : out std_logic  
16 );  
17 end SYAO_Signal_Detection;
```

Şekil 3.11 SYAO yönteminin FPGA’de uygulanması

FPGA tabanlı SYAO yöntemi tasarımına uygulanan test verileri eşik değeri ile karşılaştırılmaktadır. Ardışık olarak 3 örneklenen değer eşik değerini geçtiği durumda ‘sinyal mevcut’ tespiti yapılmaktadır. Şekil 3.12’de FPGA üzerinde tasarımı tamamlanan SYAO yöntemi ile darbesel sinyal tespit çalışmasının RTL şeması görülmektedir.



Şekil 3.12 SYAO yönteminin FPGA’de uygulanmasının RTL şeması

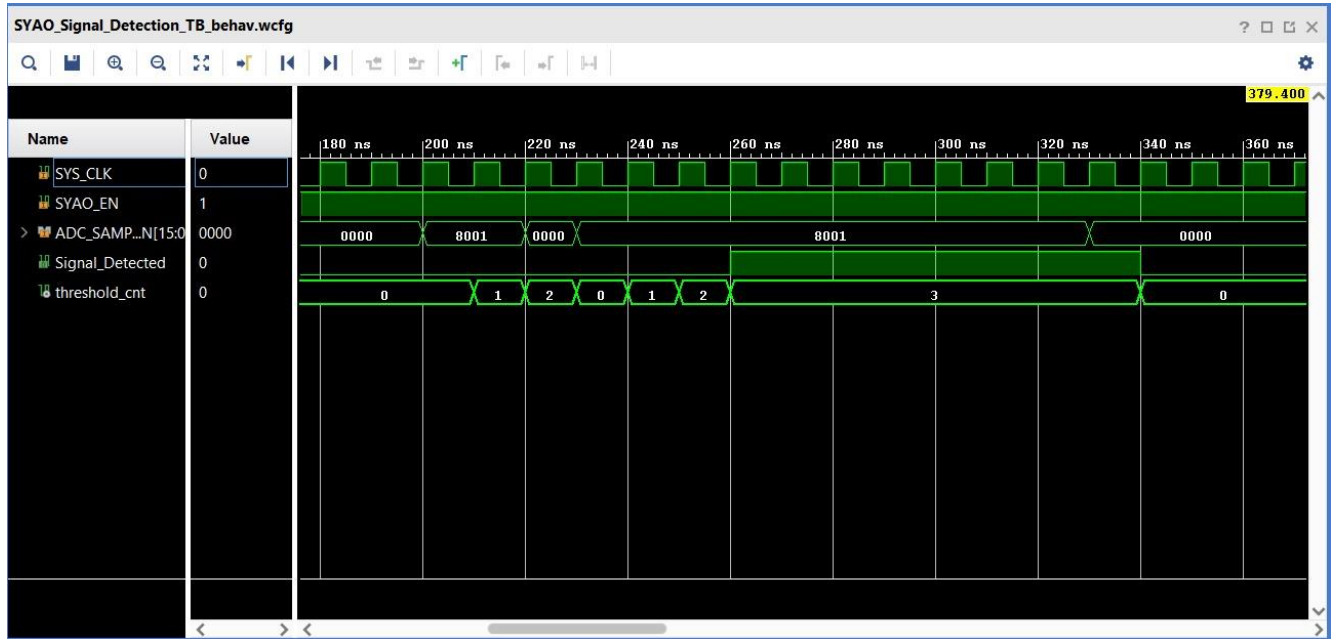
Tasarımı gerçekleştirilen SYAO yöntemi ile sinyal tespit çalışması Xilinx Artix-7 (XC7A100TFGG484-3) FPGA üzerinde sentezlenmiştir ve kaynak tüketimi Tablo 3.4’de paylaşılmıştır. Sistemin maksimum çalışma frekansı 590 MHz’dir.

Tablo 3.4 SYAO yöntemi FPGA kaynak tüketimi

<i>Kaynak Türü</i>	<i>Kullanılabilir Kaynak</i>	<i>Kullanılan Kaynak</i>	<i>Kullanım Yüzdesi</i>
LUT	63400	7	0.01
Flip-flop	126800	2	0.00

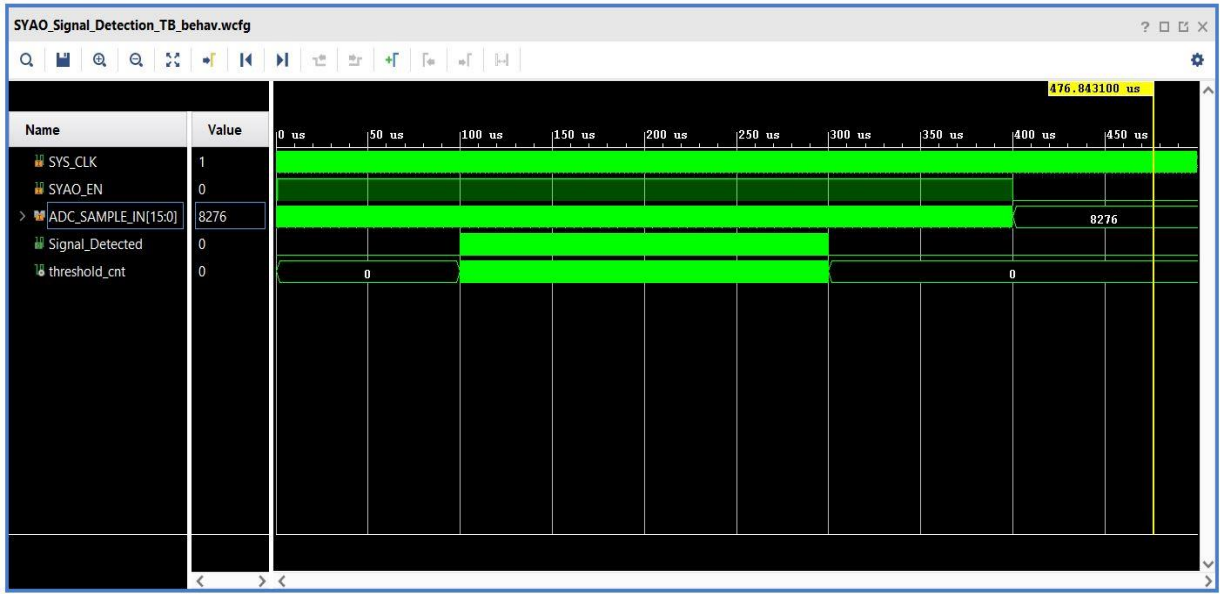
3.4. SYAO Darbesel Sinyal Tespit Yöntemi FPGA Benzetim Çalışması

Tasarımı tamamlanan SYAO yöntemi ile darbesel sinyallerin tespit çalışmasının FPGA benzetimi gerçekleştirilmiştir. Şekil 3.13’de görüldüğü gibi *SYAO_EN* sinyali mantıksal ‘1’ e uyarılarak sistem aktif hale geldikten sonra test sinyali üzerinden örneklenen ardışık 3 değer (*ADC_SAMPLE_IN*) eşik değerinden büyük olduğu durumda *Signal_Detected* mantıksal ‘1’ değerine uyarılarak ‘sinyal mevcut’ tespiti yapılmaktadır. Bu yöntem ile 3 sistem saat işareti (*SYS_CLK*) gecikme ile sinyal tespiti yapılabilmektedir. Şekil 3.13’de görüldüğü gibi örneklenen değer eşik değerinin altında kalır ise *Signal_Detected* sinyali mantıksal ‘0’ değerine geçmektedir.

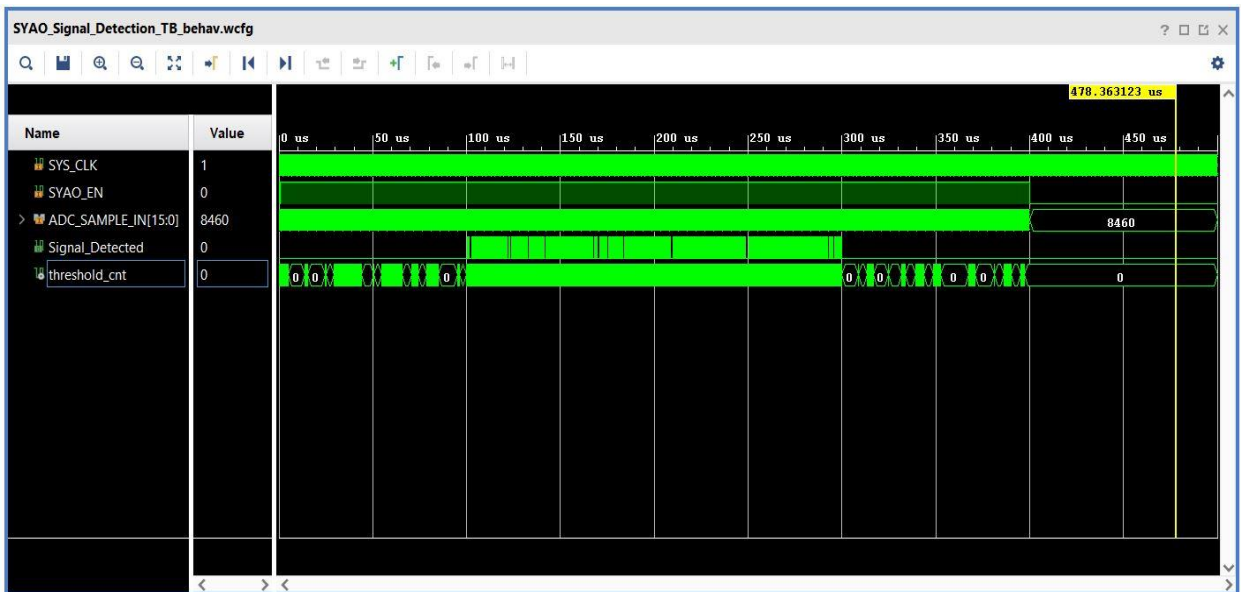


Şekil 3.13 SYAO Yöntemi FPGA Benzetim Çalışması

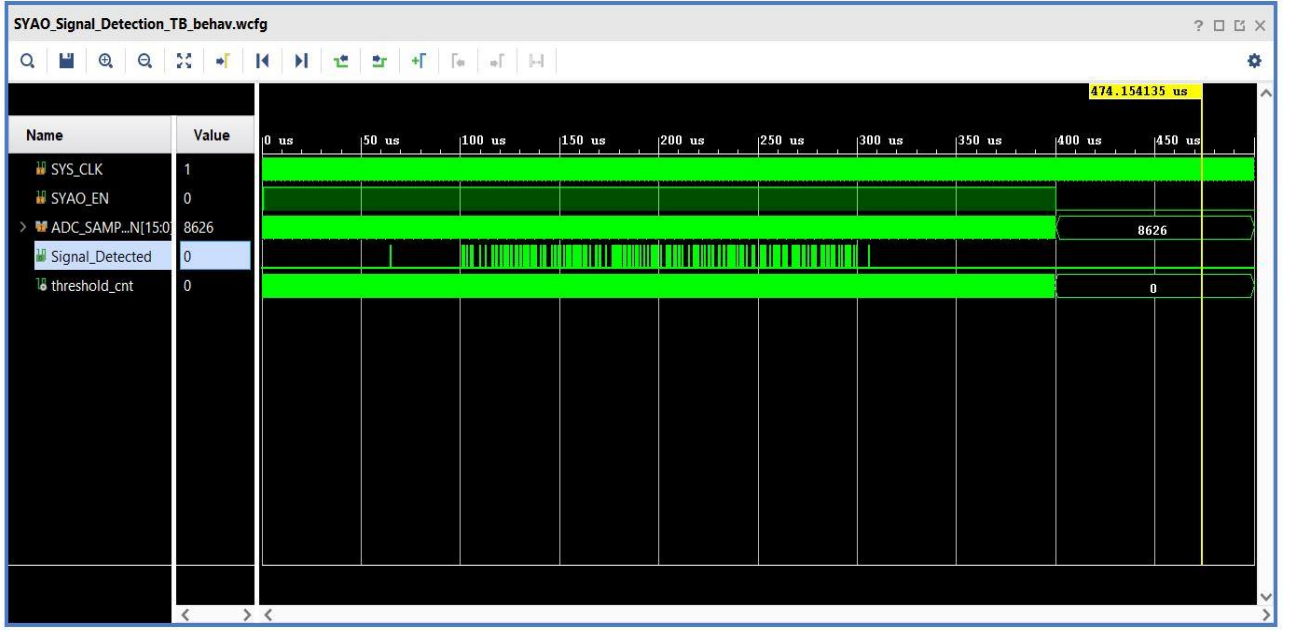
Şekil 3.14, Şekil 3.15, Şekil 3.16 ve Şekil 3.17’de sırasıyla 10 dB, 5dB, 0 dB ve -5 dB SGO seviyelerinde SYAO yöntemi ile darbesel sinyal tespit çalışmalarına yönelik gerçekleştirilen FPGA benzetim çalışmaları görülmektedir. Bölüm 3.2’deki MATLAB/Simulink modeli ile gerçekleştirilen sonuçlara benzer şekilde, SGO seviyesi düştükçe SYAO yönteminin sinyal tespit başarısı düşmektedir. 10 dB ve 5 dB SGO seviyelerindeki sonuçlar kabul edilebilir seviyededir. Ancak 0 dB ve -5 dB SGO seviyelerinde FPGA benzetim çalışmalarında görüldüğü gibi sinyal tespiti kaçırılmakta veya yanlış sinyal tespitleri gerçekleştirilmektedir.



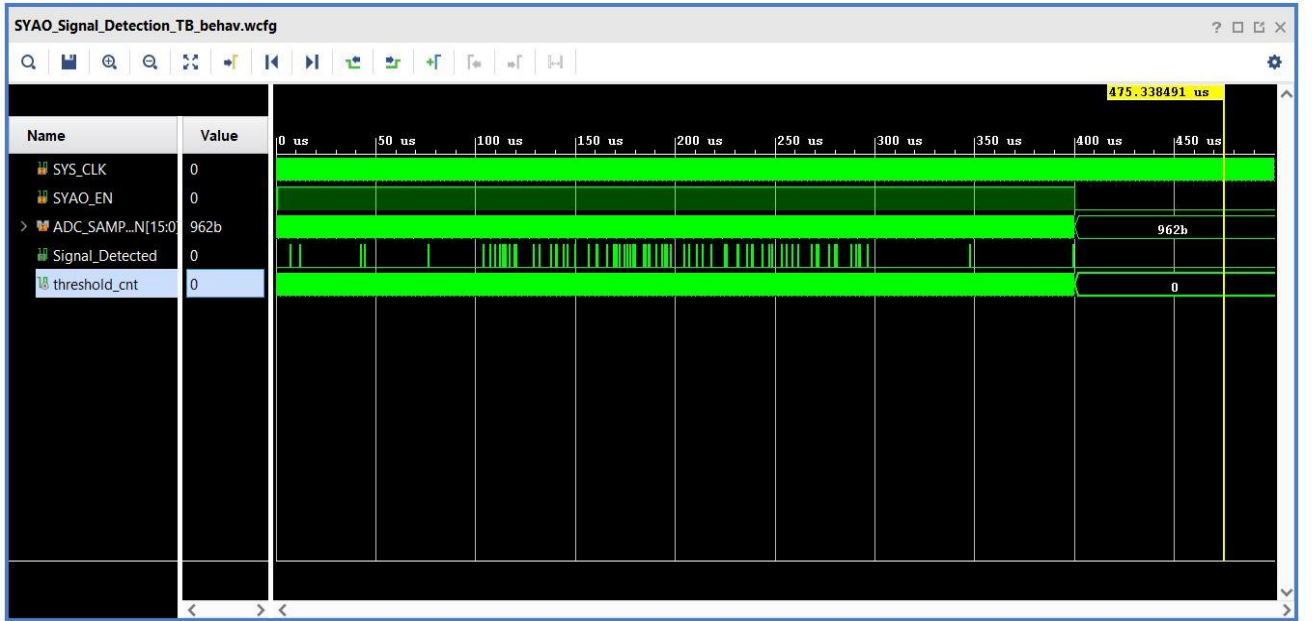
Şekil 3.14 10 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması



Şekil 3.15 5 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması



Şekil 3.16 0 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması

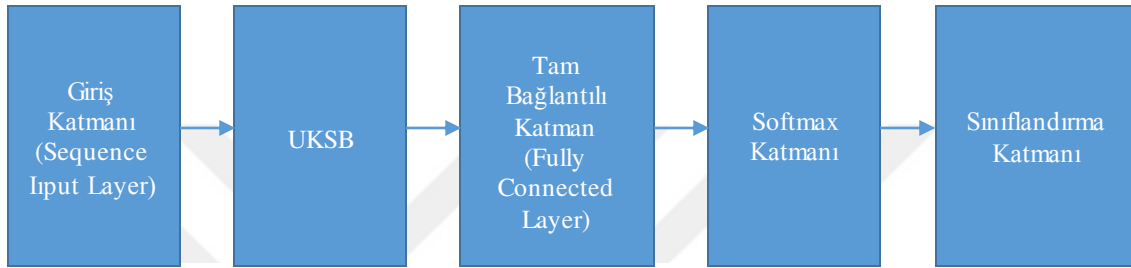


Şekil 3.17 -5 dB SGO seviyesinde SYAO yöntemi FPGA Benzetim Çalışması

SYAO yöntemi ile yüksek SGO seviyelerinde 3 sistem saat işareti (*SYS_CLK*) gibi çok düşük bir gecikme ve kaynak kullanımı ile darbesel sinyal tespiti yapılabilmesine karşılık, düşük SGO seviyelerinde aynı başarı gözlemlenmemektedir. Bu tez çalışmasında düşük SGO seviyelerinde darbesel sinyal tespitine yönelik bir UKSB yapay sinir ağı çalışması önerilmiştir.

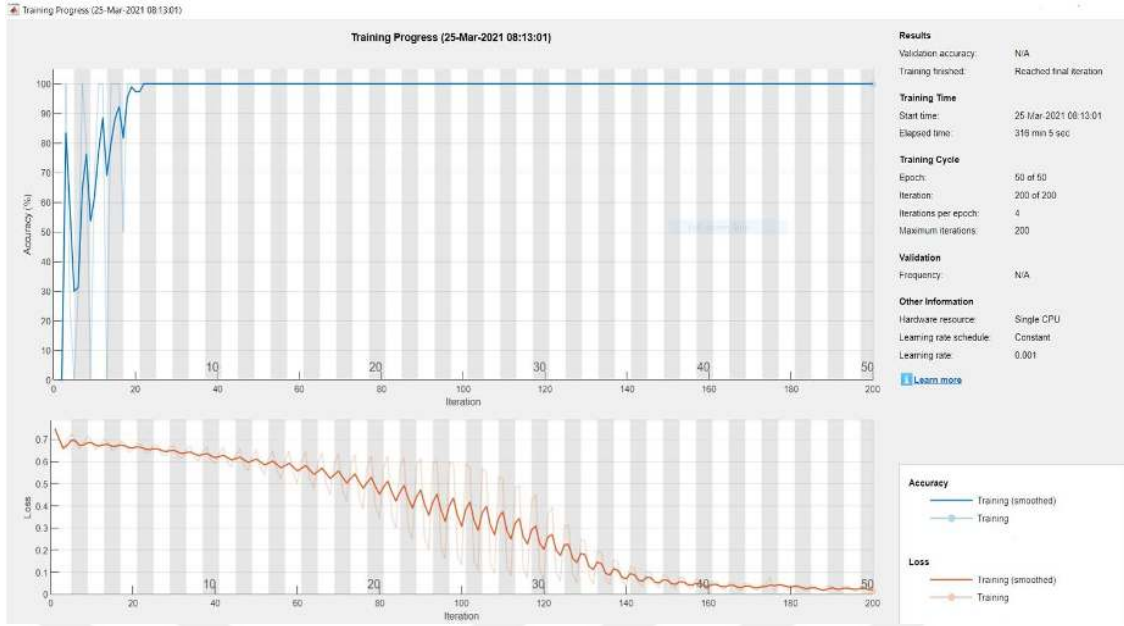
3.5. Uzun Kısa-Süreli Bellek Sinir Ağı MATLAB Çalışmaları

Bölüm 3.1’de belirtildiği şekilde eğitim ve test veri setleri elde edilmiştir. FPGA çalışmasını gerçekleştirmeden önce MATLAB UKSB yapay sinir ağı çalışması gerçekleştirilmiştir. MATLAB UKSB çalışmalarında [29] çalışması örnek alınmıştır. Şekil 3.18’deki blok diyagram, sınıflandırma işlemi için bir UKSB sinir ağının mimarisini göstermektedir. Bu sinir ağı, bir giriş katmanı (input layer), bir UKSB katmanı, bir tam bağlantılı katman (Fully connected layer), bir Softmax katmanı ve bir sınıflandırma katmanı içerir [30].



Şekil 3.18 MATLAB UKSB sinir ağı mimarisi [30]

100 MHz örnekleme hızı ile elde edilen eğitim veri setleri ile yapılan UKSB sinir ağı eğitim çalışmaları, bu tez çalışmasında kullanılan UKSB mimarisi için yeterli doğruluk değerini sağlamamaktadır. Bu yüzden Bölüm 3.1’de belirtildiği gibi 1GSPS örnekleme hızı ile elde edilen eğitim veri setleri kullanılarak 5 dB, 0 dB ve -5 dB SGO seviyeleri için MATLAB UKSB yapay sinir ağı ayrı ayrı eğitilmiştir. Şekil 3.19’de -5 dB SGO seviyesinde eğitim süresince doğruluk (accuracy) ve kayıp (loss) fonksiyonu değerleri gözlemlenmiştir. Eğitim sonunda MATLAB UKSB yapay sinir ağı doğruluk değeri %100 olurken kayıp değeri de 0’a yakınsamıştır. Intel Core i7-7700HQ CPU, 16 GB RAM, 64-bit Windows 10 ortam koşullarında 0.001 öğrenme oranı ve 200 iterasyon ile eğitim süresi 316 dakika 5 saniye sürmüştür.



Şekil 3.19 MATLAB UKSB yapay sinir ağı eğitim ekranı

Başarıyla eğitilen MATLAB UKSB sinir ağlarına Bölüm 3.1’de elde edilen 5 dB, 0 dB ve -5 dB SGO seviyelerinde 100’er adet test verisi içeren veri setleri uygulanmıştır. MATLAB UKSB sinir ağı 5 dB ve 0 dB SGO seviyelerinde **%100** doğruluk değeri, -5 dB SGO seviyesinde ise **%98** doğruluk değeri ile sınıflandırma işlemini gerçekleştirmiştir. Karışıklık Matrisleri (Confusion Matrix) sırasıyla Şekil 3.21, Şekil 3.22 ve Şekil 3.23’de paylaşılmıştır. Sınıflandırma işlemini başarıyla tamamlayan MATLAB UKSB sinir ağlarından Şekil 3.20’de gösterildiği elde edilen katsayılar (Giriş Ağırlıkları (Input Weights), Tekrarlayan Ağırlıklar (Recurrent Weights) ve Bias değerleri) UKSB sinir ağının FPGA’deki tasarım aşamalarında kullanılmaktadır.

LSTM_NETWORK.Layers(2, 1)	
Property	Value
Name	'lstm'
InputSize	1
NumHiddenUnits	1
OutputMode	'last'
StateActivationFunction	'tanh'
GateActivationFunction	'sigmoid'
InputWeights	[-0.4463;0.5582;0.4253;-0.5548]
InputWeightsInitializer	'glorot'
InputWeightsLearnRateFactor	1
InputWeightsL2Factor	1
RecurrentWeights	[-0.6090;-0.1062;0.9397;-0.4810]
RecurrentWeightsInitializer	'orthogonal'
RecurrentWeightsLearnRateFactor	1
RecurrentWeightsL2Factor	1
Bias	[0.2212;1.2243;-0.0481;0.2424]
BiasInitializer	'unit-forget-gate'
BiasLearnRateFactor	1
BiasL2Factor	0
HiddenState	0
CellState	0
NumInputs	1
InputNames	1x1 cell
NumOutputs	1
OutputNames	1x1 cell

Şekil 3.20 -5 dB SGO için eğitilen MATLAB UKSB sinir ağının ağırlık ve bias değerleri

Karışıklık Matrisi (Confusion Matrix)			
	Sınıflandırma Sonucu		
		Sınıflandırma Sonucu '1'	Sınıflandırma Sonucu '0'
Gerçek Değer	Gerçek Değer '1'	50	0
	Gerçek Değer '0'	0	50
	Doğruluk (Accuracy)	1	
	Kesinlik (Precision)	1	
	Duyarlılık (Recall)	1	
	F1 Puanı (F1 Score)	1	

Şekil 3.21 MATLAB UKSB'de 5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi

Karışıklık Matrisi (Confusion Matrix)			
	Sınıflandırma Sonucu		
		Sınıflandırma Sonucu '1'	Sınıflandırma Sonucu '0'
Gerçek Değer	Gerçek Değer '1'	50	0
	Gerçek Değer '0'	0	50
	Doğruluk (Accuracy)	1	
	Kesinlik (Precision)	1	
	Duyarlılık (Recall)	1	
	F1 Puanı (F1 Score)	1	

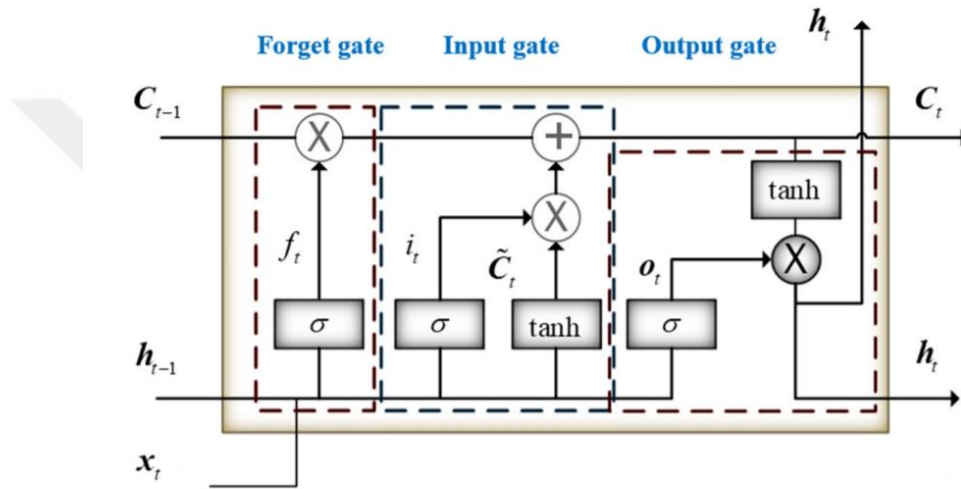
Şekil 3.22 MATLAB UKSB'de 0 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi

Karışıklık Matrisi (Confusion Matrix)			
	Sınıflandırma Sonucu		
		Sınıflandırma Sonucu '1'	Sınıflandırma Sonucu '0'
Gerçek Değer	Gerçek Değer '1'	50	0
	Gerçek Değer '0'	2	48
	Doğruluk (Accuracy)	0.98	
	Kesinlik (Precision)	0.9615	
	Duyarlılık (Recall)	1	
	F1 Puanı (F1 Score)	0.9804	

Şekil 3.23 MATLAB UKSB'de -5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi

3.6. Uzun Kısa-Süreli Bellek Sinir Ağı FPGA Tasarım Çalışmaları

Bölüm 3.5’de belirtildiği gibi MATLAB UKSB sinir ağı çalışmaları tamamlandıktan sonra UKSB sinir ağının FPGA’deki tasarım çalışmalarına başlanmıştır. FPGA üzerinde UKSB sinir ağı uygulamasını gerçekleştirmek için öncelikle bir adet UKSB hücresi tasarlanmıştır. Tasarım çalışmalarında Xilinx FPGA ve VHDL donanım tanımlama dili kullanılmıştır. Ayrıca tasarım aşamalarında detayları ilerleyen sayfalarda paylaşılan Xilinx IP (Intellectual Property, Fikri Mülkiyet) çekirdekleri de kullanılmıştır. Tipik bir UKSB hücresi Şekil 3.24’de gösterilmiştir.



Şekil 3.24 Tipik bir UKSB hücre yapısı [31]

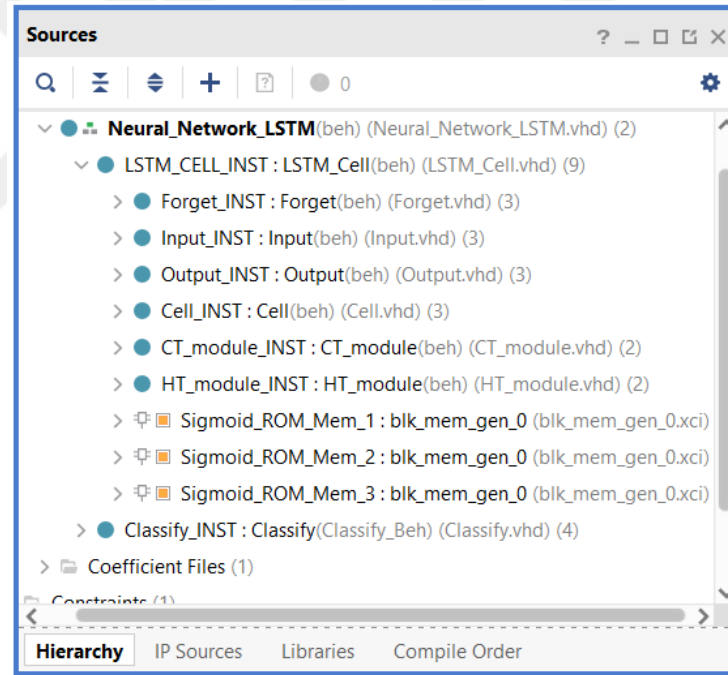
Tipik bir UKSB hücresine ait eşitlikler Tablo 3.5’te gösterilmiştir.

Tablo 3.5 Tipik bir UKSB hücresi eşitlikleri

Unutma Kapısı (Forget Gate) Aktivasyon Fonksiyonu	$f_t = \sigma_g (W_f \mathbf{x}_t + R_f \mathbf{h}_{t-1} + b_f)$
Giriş Kapısı (Input Gate) Aktivasyon Fonksiyonu	$i_t = \sigma_g (W_i \mathbf{x}_t + R_i \mathbf{h}_{t-1} + b_i)$
Çıkış Kapısı (Output Gate) Aktivasyon Fonksiyonu	$o_t = \sigma_g (W_o \mathbf{x}_t + R_o \mathbf{h}_{t-1} + b_o)$
Hücre Giriş (Cell Input) Aktivasyon Fonksiyonu	$\tilde{c}_t = \sigma_c (W_c \mathbf{x}_t + R_c \mathbf{h}_{t-1} + b_c)$
Hücre Durumu (Cell State) Fonksiyonu	$\mathbf{c}_t = f_t \odot \mathbf{c}_{t-1} + i_t \odot \tilde{c}_t$
Gizli Durum (Hidden State) Fonksiyonu	$\mathbf{h}_t = o_t \odot \sigma_c (\mathbf{c}_t)$

Giriş Ağırlıkları (Input Weights)	W
Tekrarlayan Ağırlıklar (Recurrent Weights)	R
Bias	b
Hadamard Çarpımı (Vektörlerin Eleman Bazında Çarpımı)	$\odot$
Sigmoid Fonksiyonu	σ_g
Hiperbolik Tanjant Fonksiyonu	σ_c
UKSB Giriş Vektörü	$\mathbf{x}_f$

FPGA ile UKSB hücresi oluşturmak için yapılan çalışmalar aşağıda paylaşılmıştır. Şekil 3.25’de görüldüğü gibi FPGA projesinde Tablo 3.5’te belirtilen eşitlikleri uygulamak için modüler bir tasarım yapılmıştır.



Şekil 3.25 FPGA UKSB Hiyerarşi

Forget_INST modülü, Tablo 3.5’te belirtilen Unutma Kapısı (Forget Gate) Aktivasyon Fonksiyonu işlemlerini gerçekleştirmektedir.

Input_INST modülü, Tablo 3.5’te belirtilen Giriş Kapısı (Input Gate) Aktivasyon Fonksiyonu işlemlerini gerçekleştirmektedir.

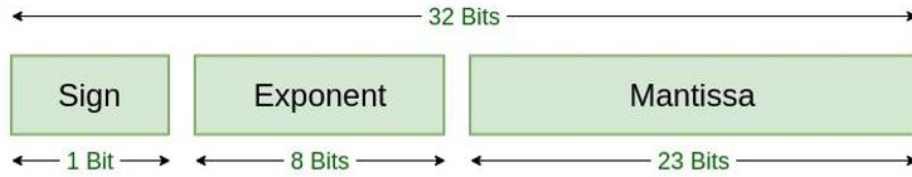
Output_INST modülü, Tablo 3.5'te belirtilen Çıkış Kapısı (Output Gate) Aktivasyon Fonksiyonu işlemlerini gerçekleştirmektedir.

Cell_INST modülü, Tablo 3.5'te belirtilen Hücre Giriş (Cell Input) Aktivasyon Fonksiyonu işlemlerini gerçekleştirmektedir.

CT_module_INST modülü, Tablo 3.5'te belirtilen Hücre Durumu (Cell State) Fonksiyonu işlemlerini gerçekleştirmektedir.

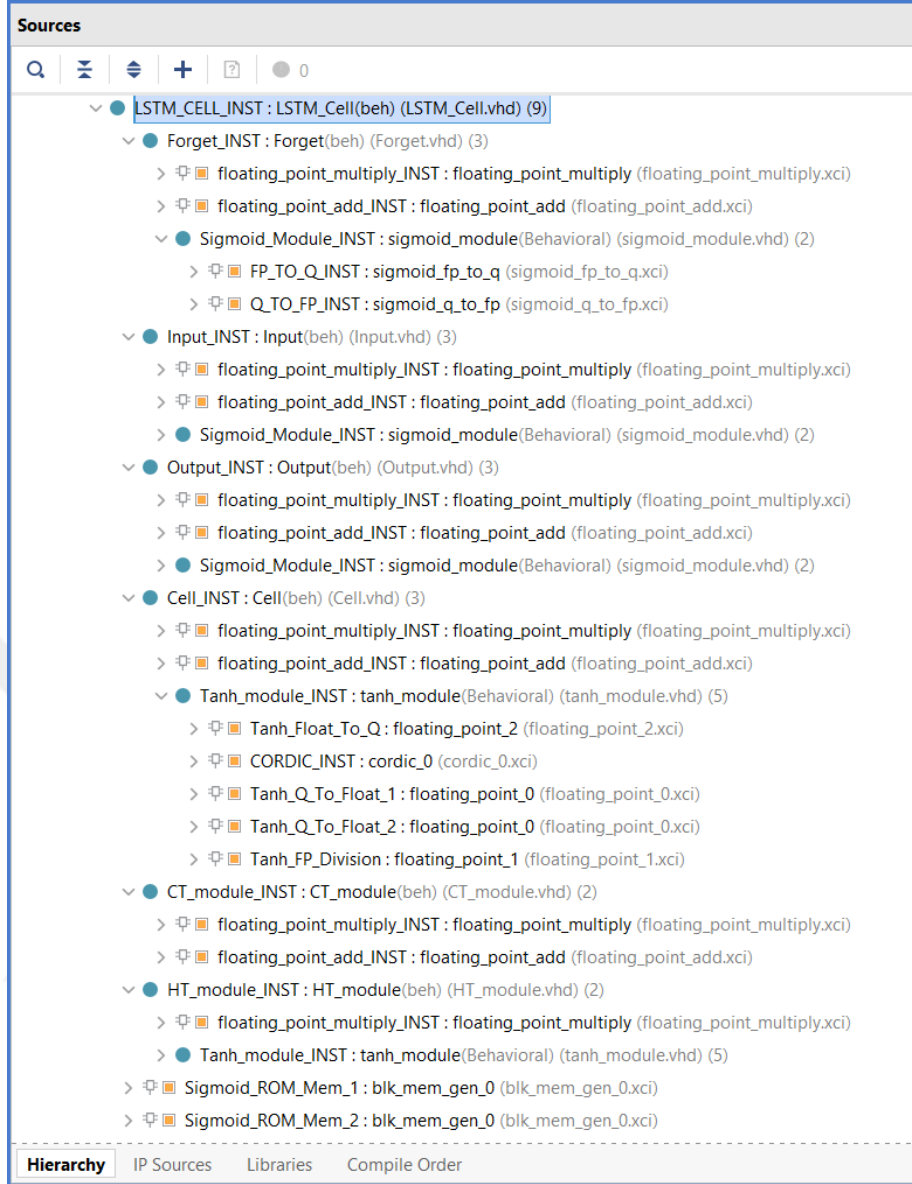
HT_module_INST modülü, Tablo 3.5'te belirtilen Gizli Durum (Hidden State) Fonksiyonu işlemlerini gerçekleştirmektedir.

Yukarıda belirtilen UKSB eşitliklerinde gerçekleştirilen tüm matematiksel işlemler hassas sonuçlar gerektirdiği için Şekil 3.26'da yapısı gösterilen IEEE-754 Tek Duyarlıklı Kayan Nokta (Single Precision Floating Point) sayılar ile gerçekleştirilmektedir.



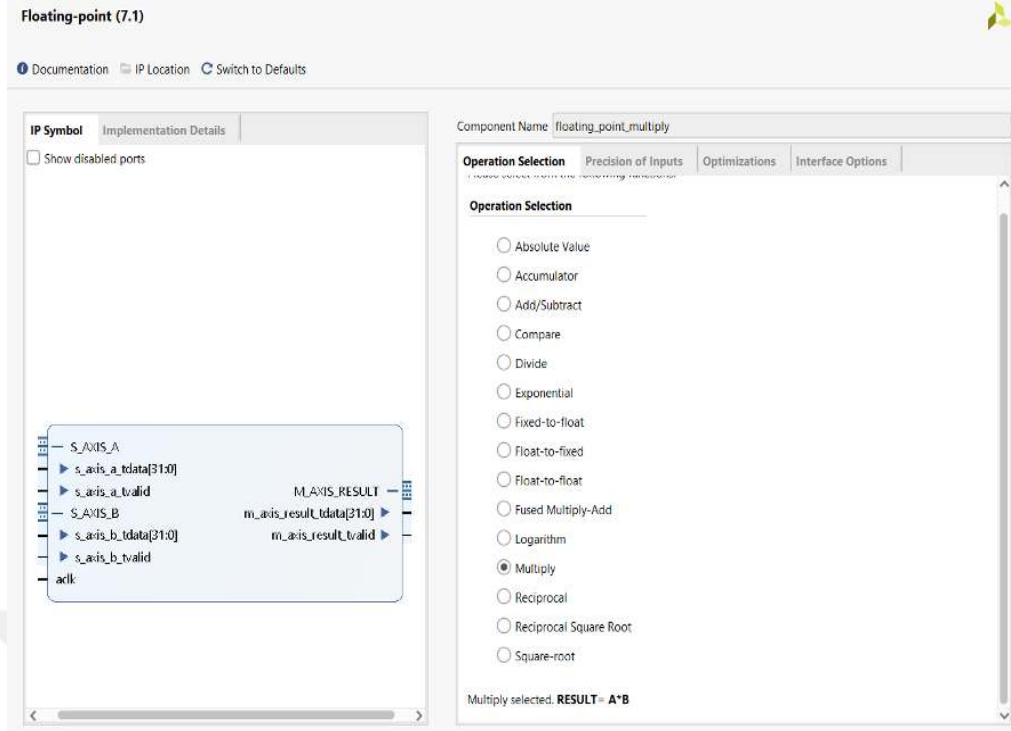
Şekil 3.26 IEEE-754 Tek Duyarlıklı Kayan Nokta (Single Precision Floating Point)

FPGA'de tabanlı bir UKSB hücresi oluşturmak için tasarlanan alt modüllerin detaylı hiyerarşisi Şekil 3.27'de gösterilmiştir. Bu projede çeşitli IP çekirdekleri kullanılmaktadır.



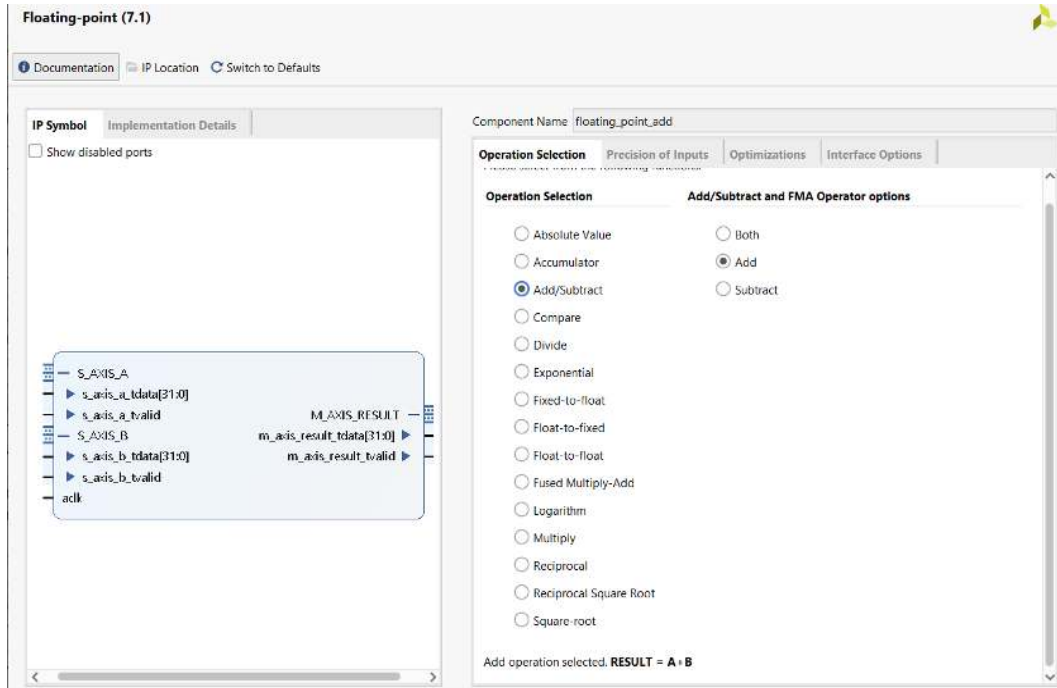
Şekil 3.27 FPGA UKSB Detaylı Hiyerarşi

Forget_INST modülünde, Input_INST modülünde, Output_INST modülünde, Cell_INST modülünde, CT_module_INST modülünde, HT_module_INST modülünde ve Classify_INST modülünde Şekil 3.28’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği çarpma işlemi için kullanılmaktadır. Giriş ve çıkış sayıları IEEE-754 formatındadır.



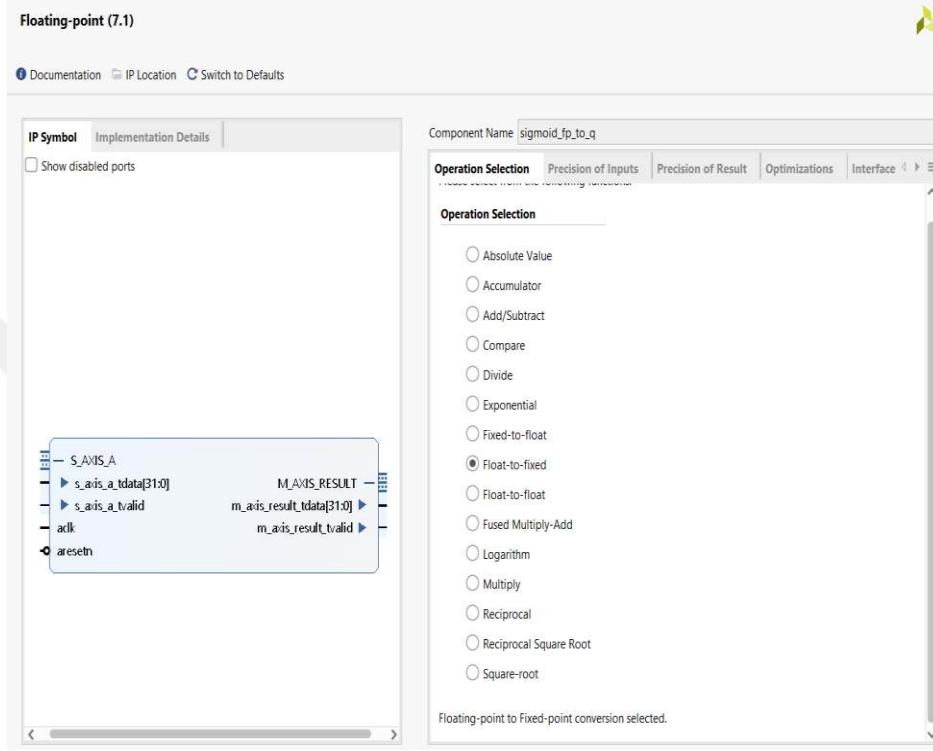
Şekil 3.28 Çarpma işlemi için kullanılan IP çekirdeği

Forget_INST modülünde, Input_INST modülünde, Output_INST modülünde, Cell_INST modülünde, CT_module_INST modülünde ve Classify_INST modülünde Şekil 3.29’da detayları paylaşılan Floating-Point (7.1) IP çekirdeği toplama işlemi için kullanılmaktadır. Giriş ve çıkış sayıları IEEE-754 formatındadır.

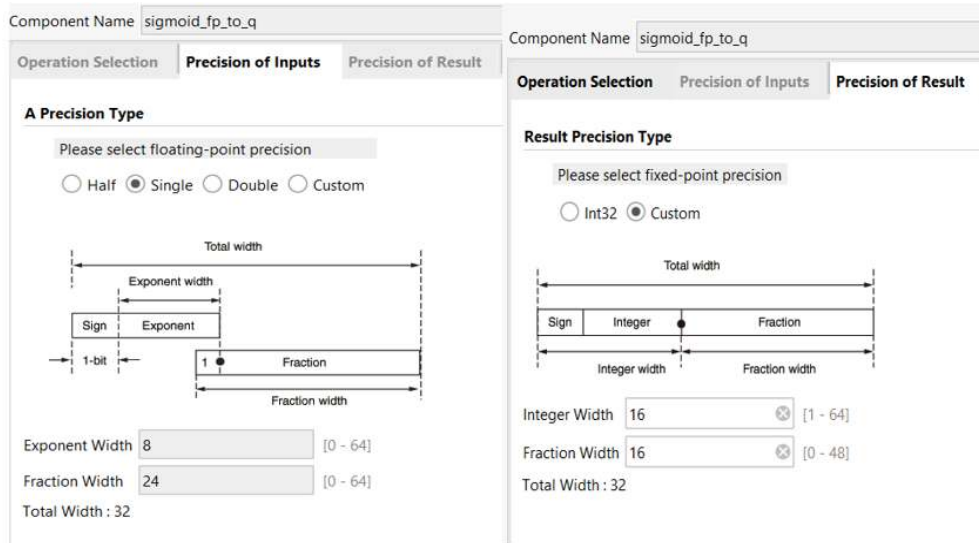


Şekil 3.29 Toplama işlemi için kullanılan IP çekirdeği

Forget_INST modülünde, Input_INST modülünde, Output_INST modülünde ve Classify_INST modülünde Şekil 3.30 ve Şekil 3.31’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği Kayan Nokta Gösterimi (Floating Point) - Sabit Nokta Gösterimi (Fixed Point) dönüşümü için kullanılmaktadır. Bu dönüşüm sigmoid fonksiyonu hesaplamalarında kullanılmaktadır.

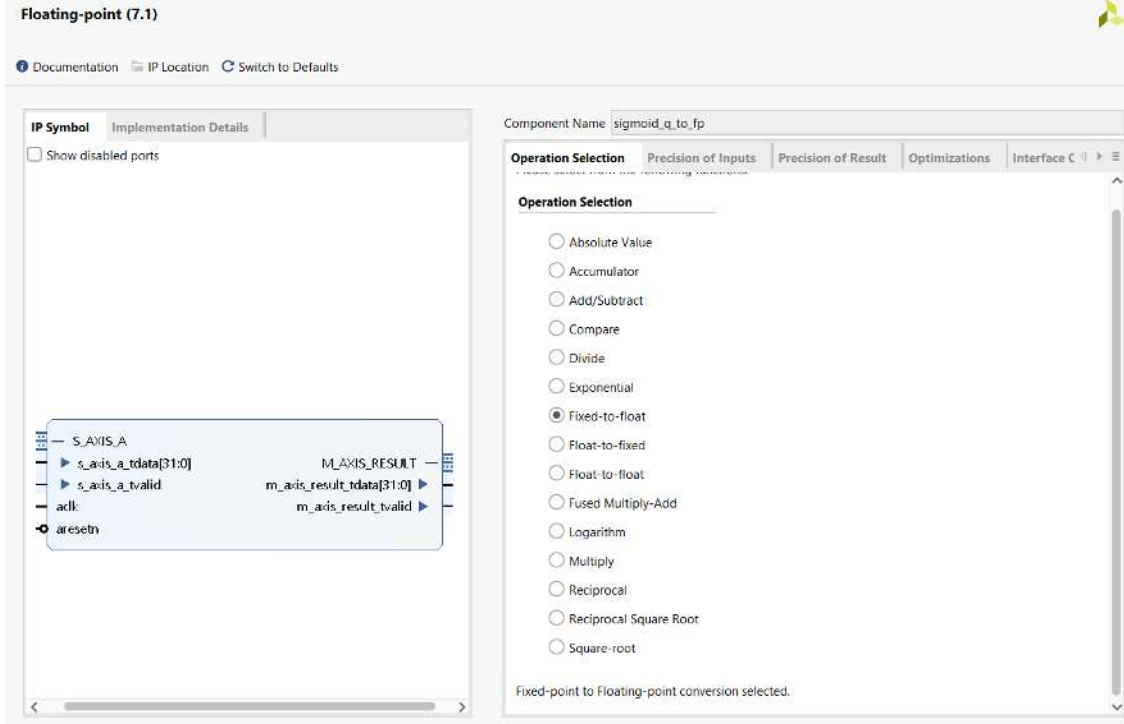


Şekil 3.30 Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği

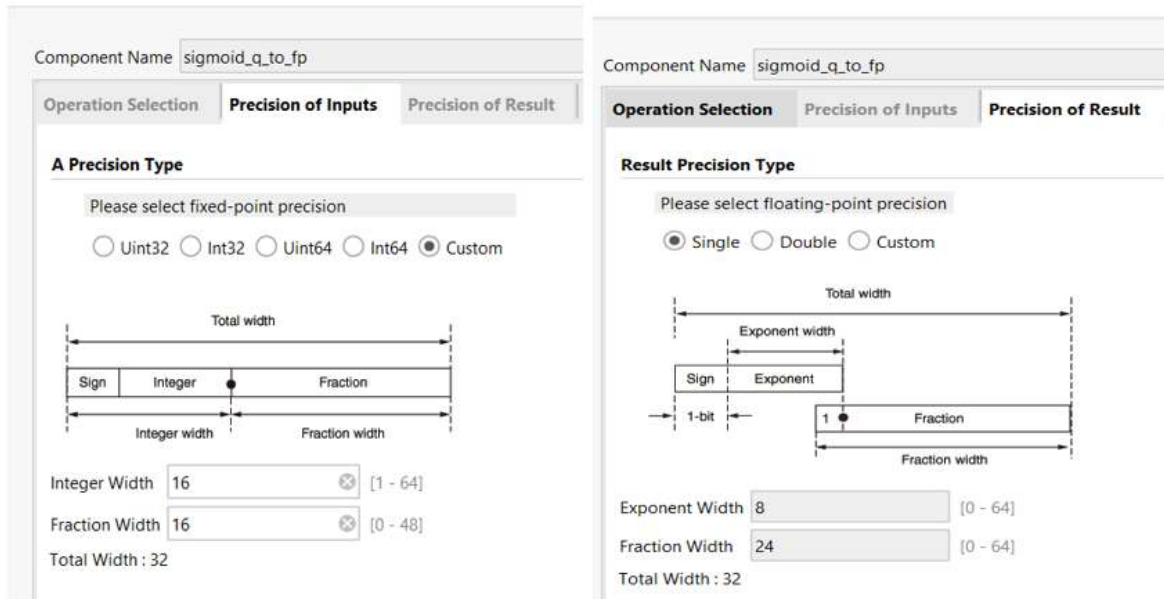


Şekil 3.31 Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları

Forget_INST modülünde, Input_INST modülünde, Output_INST modülünde ve Classify_INST modülünde Şekil 3.32 ve Şekil 3.33’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği Sabit Nokta Gösterimi (Fixed Point) - Kayan Nokta Gösterimi (Floating Point) dönüşümü için kullanılmaktadır. Bu dönüşüm sigmoid fonksiyonu hesaplamalarında kullanılmaktadır.

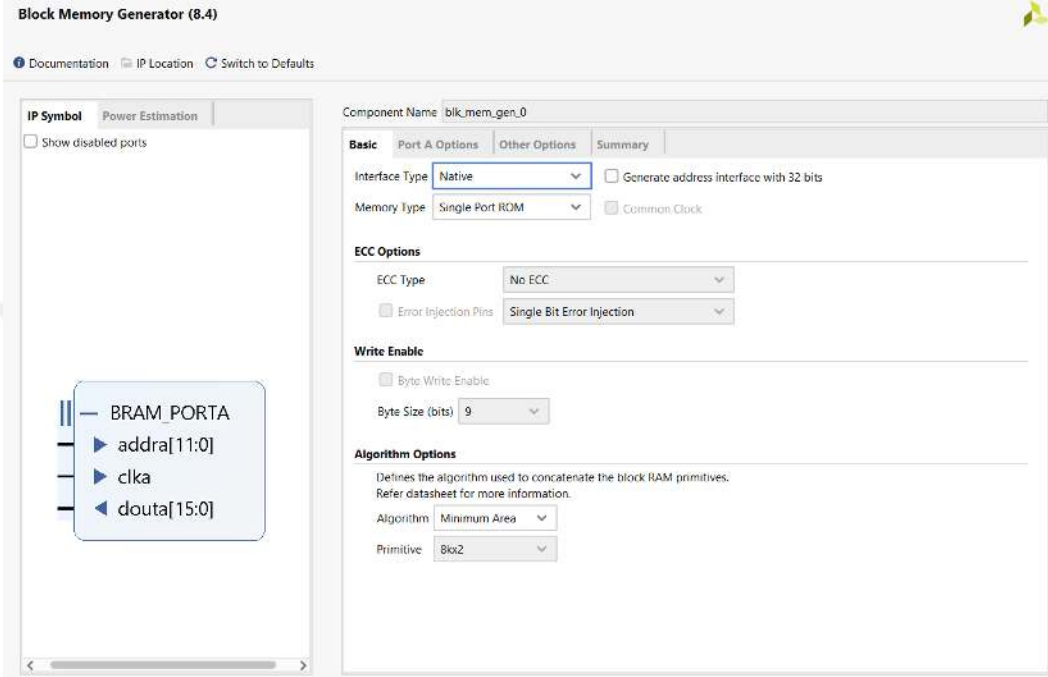


Şekil 3.32 Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği

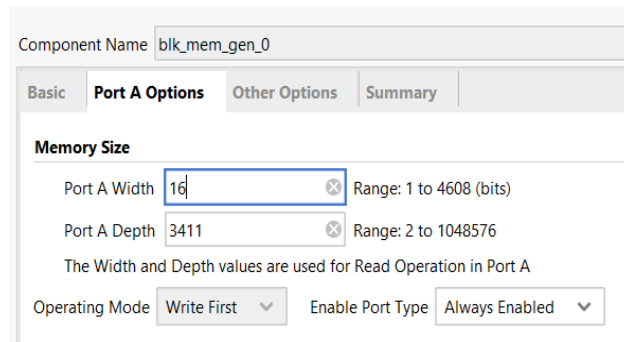


Şekil 3.33 Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları

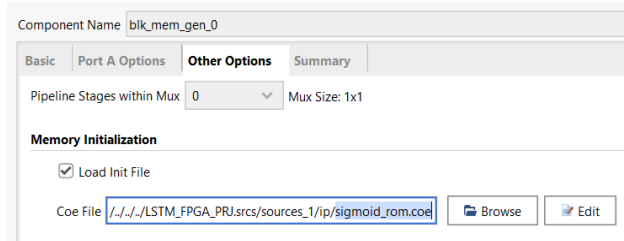
Forget_INST modülünde, Input_INST modülünde, Output_INST modülünde ve Classify_INST modülünde Şekil 3.34, Şekil 3.35 ve Şekil 3.36’de detayları paylaşılan Block Memory Generator (8.4) IP çekirdeği, sigmoid fonksiyonu hesaplamalarında LUT (Lookup-Table) amacıyla kullanılmaktadır. Sigmoid fonksiyonu değerleri bu bellek alanında saklanmaktadır.



Şekil 3.34 Block Memory Generator IP çekirdeği

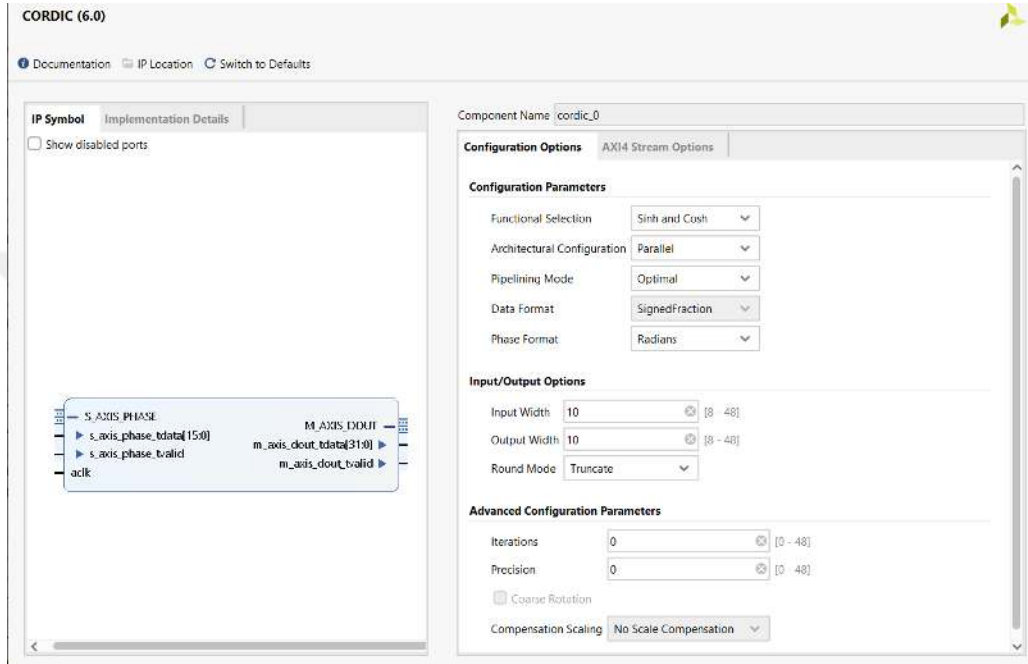


Şekil 3.35 Block Memory Generator IP çekirdeği parametreleri

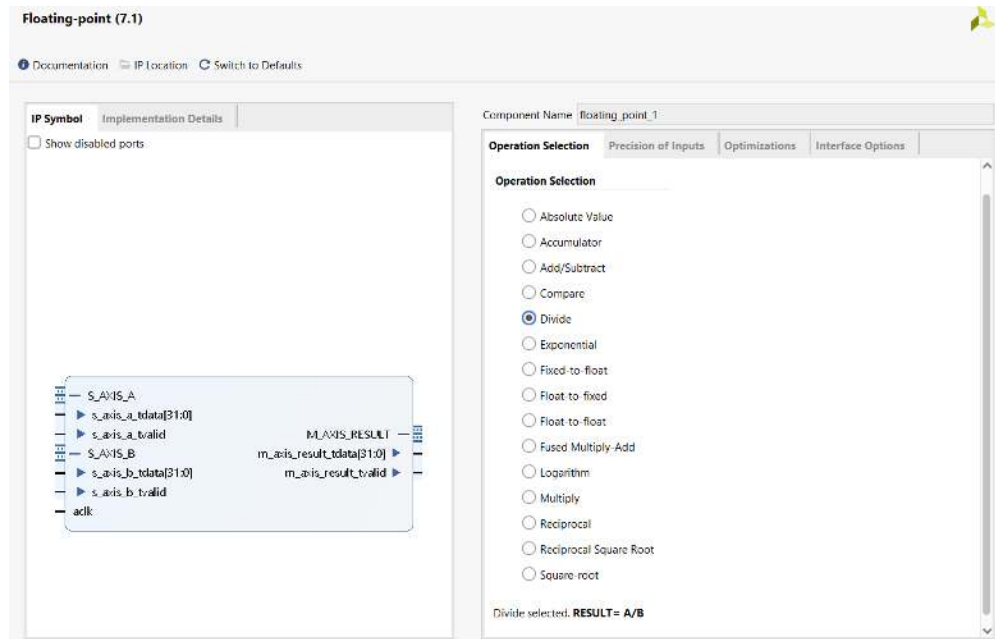


Şekil 3.36 Sigmoid fonksiyonu değerlerinin IP çekirdeğine yüklenmesi

Cell_INST modülünde ve HT_module_INST modülünde Şekil 3.37’da detayları paylaşılan CORDIC (6.0) IP çekirdeği, hiperbolik tanjant fonksiyonu hesaplamalarında kullanılmaktadır. Bu IP çekirdeği verilen giriş değerinin hiperbolik sinüs ve hiperbolik kosinüs değerlerini vermektedir. Şekil 3.38’da detayları paylaşılan Floating-Point (7.1) IP çekirdeği yardımıyla hiperbolik sinüs değeri hiperbolik kosinüs değerine bölünerek hiperbolik tanjant değeri elde edilmektedir.

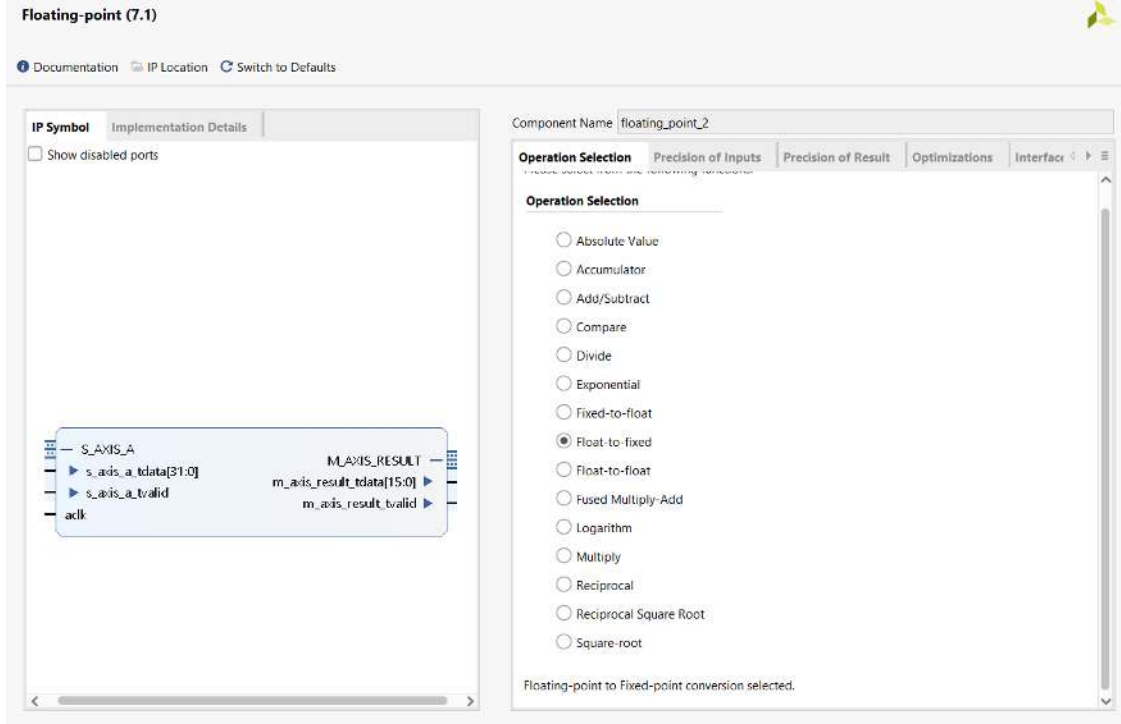


Şekil 3.37 CORDIC IP çekirdeği

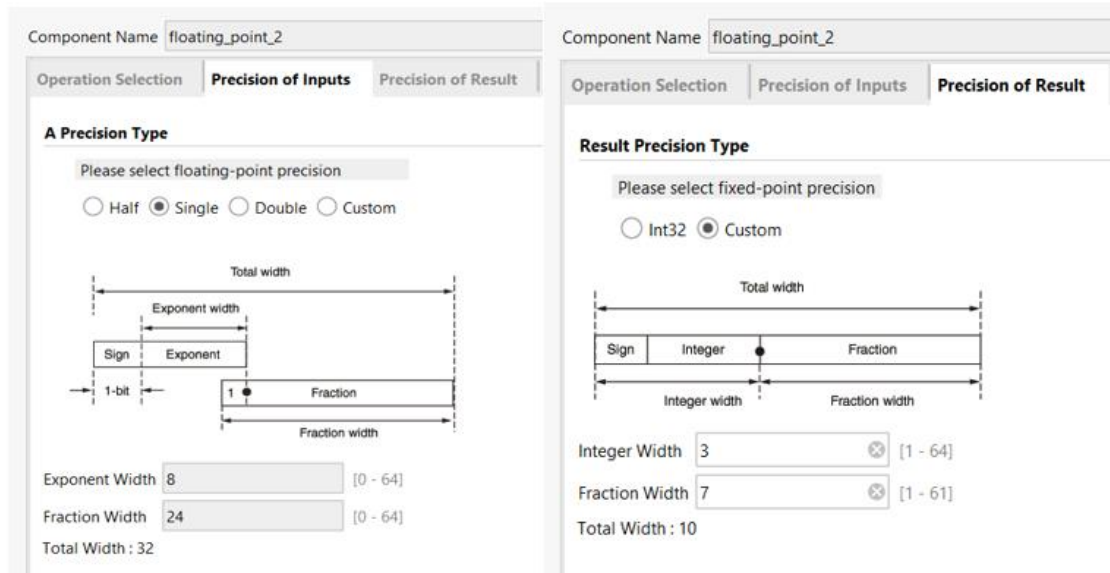


Şekil 3.38 Bölme işlemi için kullanılan IP çekirdeği

Cell_INST modülünde ve HT_module_INST modülünde Şekil 3.39’de ve Şekil 3.40’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği Kayan Nokta Gösterimi (Floating Point) - Sabit Nokta Gösterimi (Fixed Point) dönüşümü için kullanılmaktadır. Bu dönüşüm hiperbolik tanjant fonksiyonu hesaplamalarında kullanılmaktadır.

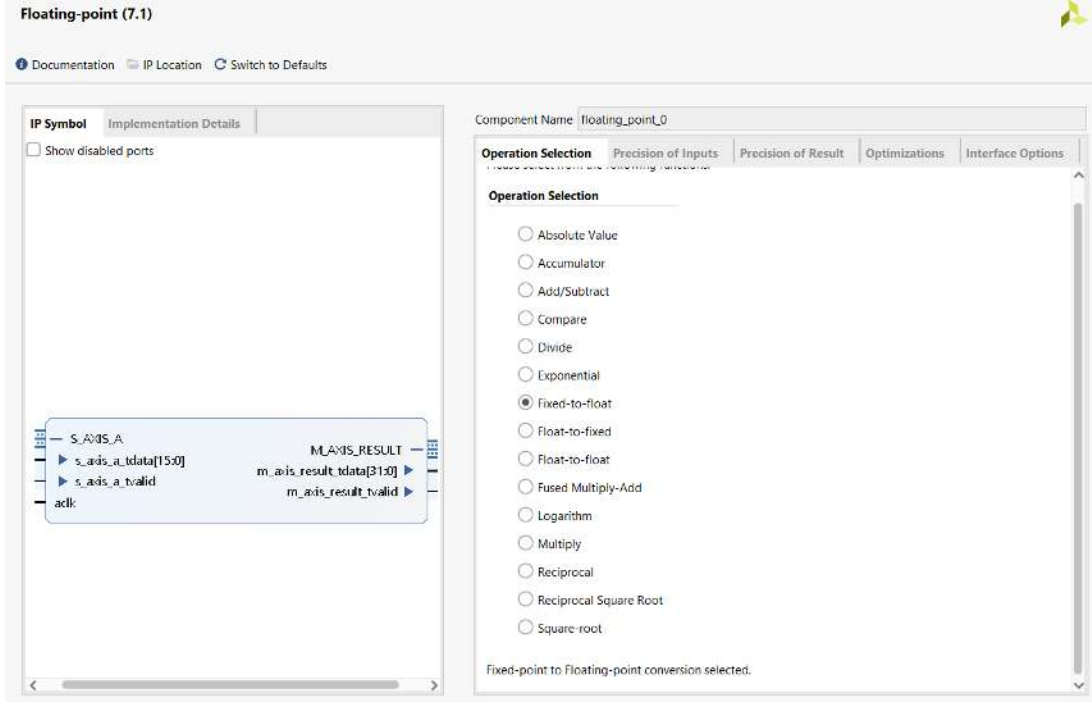


Şekil 3.39 Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği

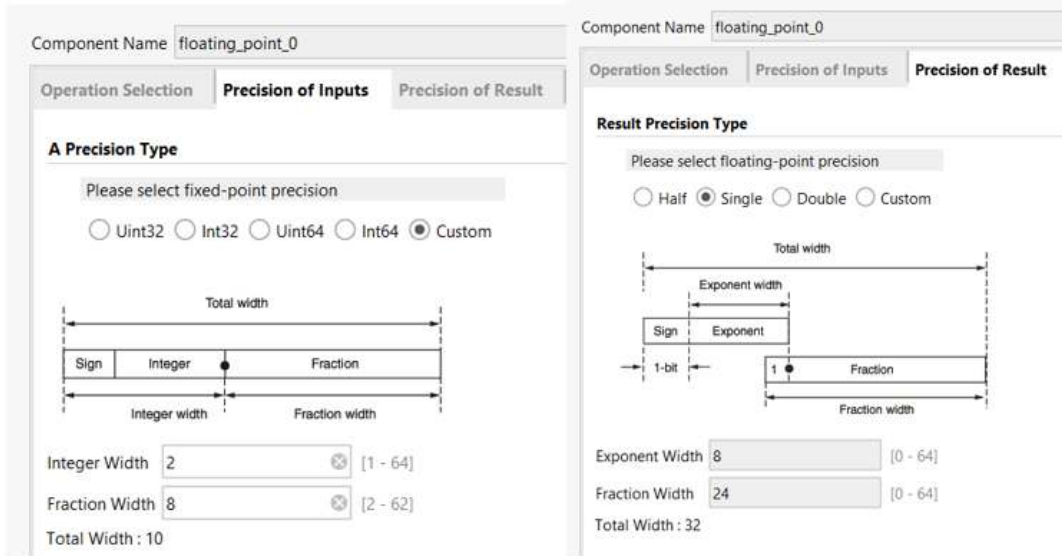


Şekil 3.40 Kayan Nokta Gösterimi - Sabit Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları

Cell_INST modülünde ve HT_module_INST modülünde Şekil 3.41’de ve Şekil 3.42’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği Sabit Nokta Gösterimi (Fixed Point) - Kayan Nokta Gösterimi (Floating Point) dönüşümü için kullanılmaktadır. Bu dönüşüm hiperbolik tanjant fonksiyonu hesaplamalarında kullanılmaktadır.

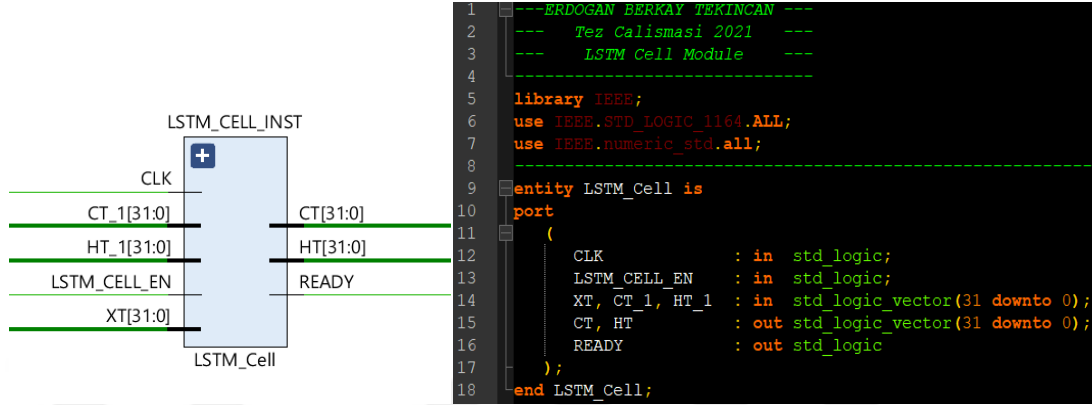


Şekil 3.41 Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeği



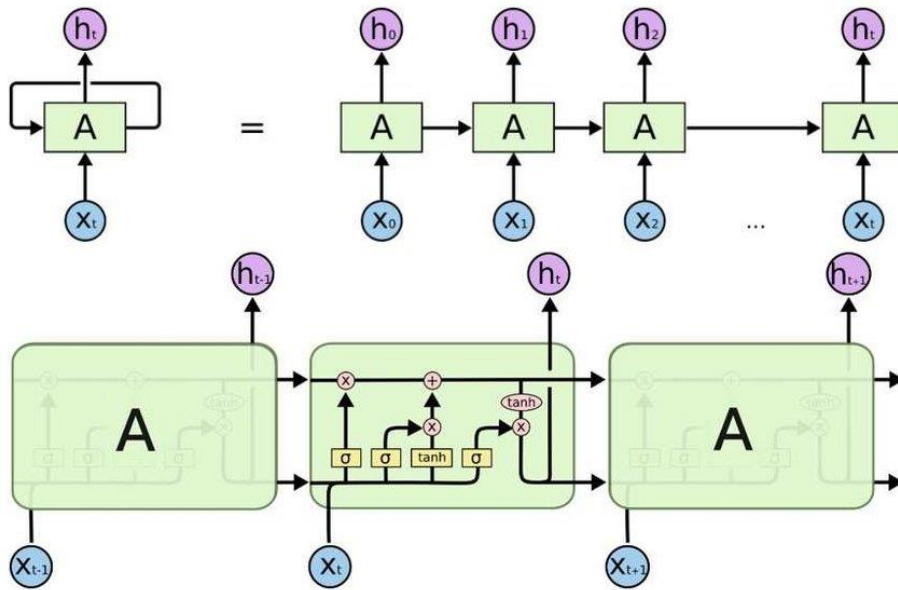
Şekil 3.42 Sabit Nokta Gösterimi - Kayan Nokta Gösterimi Dönüşümü için kullanılan IP Çekirdeğinin giriş ve çıkışları

FPGA tabanlı bir UKSB sinir ağı hücresi oluşturmak için tasarlanan alt modüller ve bu alt modüllerde kullanılan IP çekirdekleri açıklanmıştır. Bu IP çekirdeklerine ek olarak VHDL donanım tanımla dili kullanılarak bir UKSB hücresinin tasarımı tamamlanmıştır ve Şekil 3.43’de gösterilmiştir.



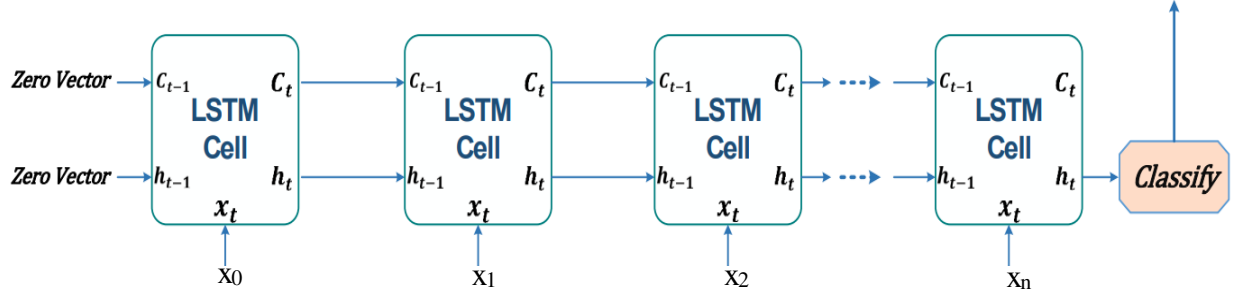
Şekil 3.43 FPGA’de tasarımı tamamlanan bir UKSB hücresi

UKSB yapay sinir ağı, bir tekrarlayan sinir ağı modelidir. TSA mimarisinin tekrarlayan (yinelenen) olarak adlandırılmasının sebebi, bir giriş dizisinin her ögesi için aynı görevi önceki çıktılara bağlı olarak yerine getirmesidir. Tüm tekrarlayan sinir ağıları gibi UKSB sinir ağı da Şekil 3.44’da gösterildiği gibi tekrar eden bir sinir ağı zinciri biçimindedir.



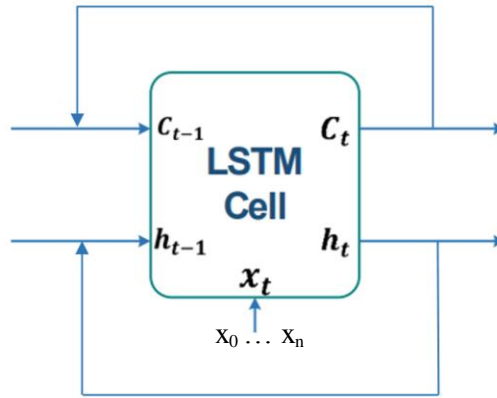
Şekil 3.44 UKSB Tekrarlayan Sinir Ağı Yapısı [11]

Sınıflandırma işleminin başarıyla tamamlanabilmesi için tasarlanan UKSB hücresi kendisini tekrarlayarak Şekil 3.45’de gösterildiği gibi bir sinir ağı zincirine dönüştürülmüştür. En son katman ise sınıflandırma katmanıdır.



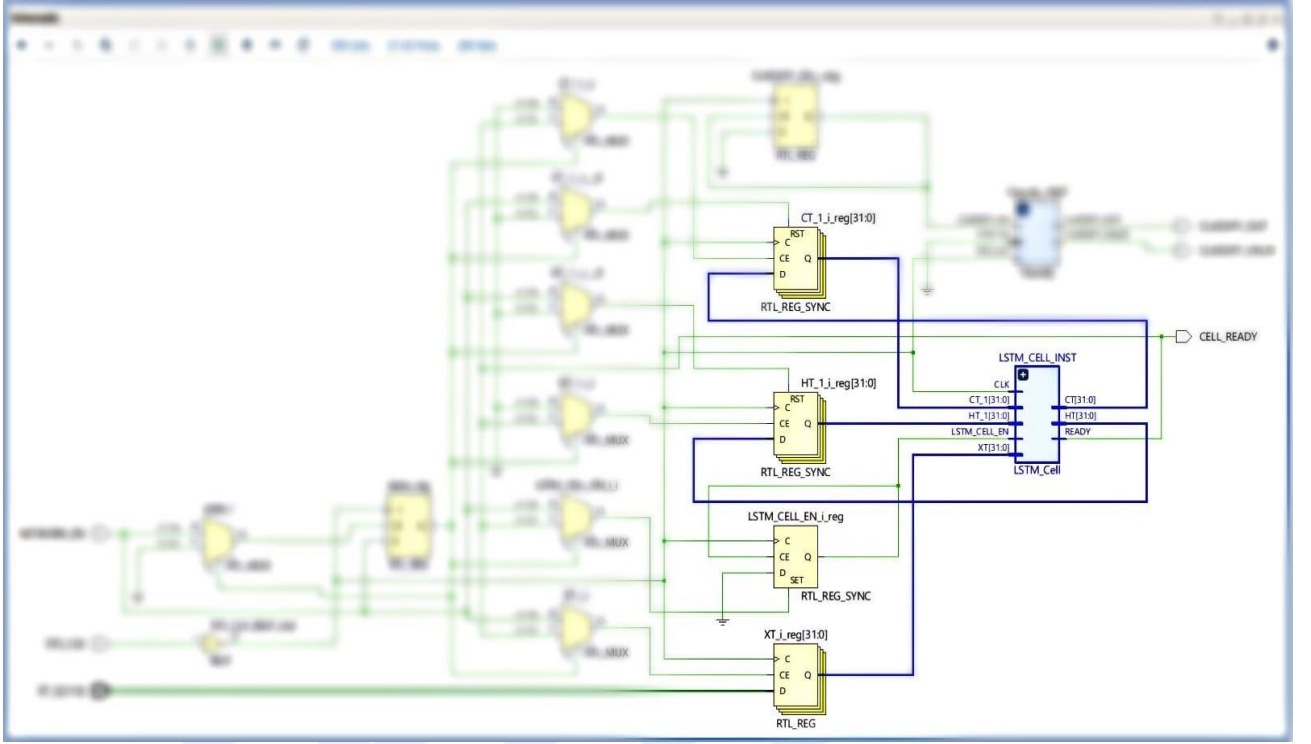
Şekil 3.45 UKSB hücrelerinin zincir yapısı

Şekil 3.45’de görüldüğü gibi bir önceki UKSB hücresine ait c_t ve h_t çıktıları bir sonraki UKSB hücresinin c_{t-1} ve h_{t-1} girişlerine bağlanmaktadır ve her seferinde yeni bir giriş değeri x_t ile Tablo 3.5’te belirtilen eşitlikleri tekrar tekrar gerçekleştirmektedir. Bu nedenle bu yapı Şekil 3.46’deki gibi ifade edilebilir.



Şekil 3.46 UKSB hücresinin kendini tekrarlayan zincir yapısı

Şekil 3.46’deki tasarım Şekil 3.47 ve Şekil 3.48’de gösterildiği gibi FPGA’de başarıyla uygulanmıştır. Şekil 3.46’da gösterildiği gibi bir adet UKSB hücresinde Tablo 3.5’te belirtilen eşitlikler sonucu elde edilen c_t ve h_t çıktıları tekrar kendi UKSB hücresinin c_{t-1} ve h_{t-1} girişlerine bağlanmaktadır. Ayrıca Şekil 3.45’de belirtildiği gibi ilk değer olarak sıfır değerleri c_{t-1} ’e ve h_{t-1} ’e yüklenmektedir. Bölüm 3.1 de belirtildiği gibi bu tez çalışmasında oluşturulan test veri setlerindeki veriler 400.000 adet örnekleme değerinden oluşmaktadır. Böylelikle bu işlem her bir test verisi için 400.000 defa kendini tekrar eden bir sinir ağı yapısıdır. En son giriş verisi $x_{400.000}$ ile işlem yapıldıktan sonra elde edilen h_t çıktısı sınıflandırma (classify) modülüne girmektedir.



Şekil 3.47 UKSB hücresinin kendini tekrarlayan zincir yapısının FPGA uygulaması

```

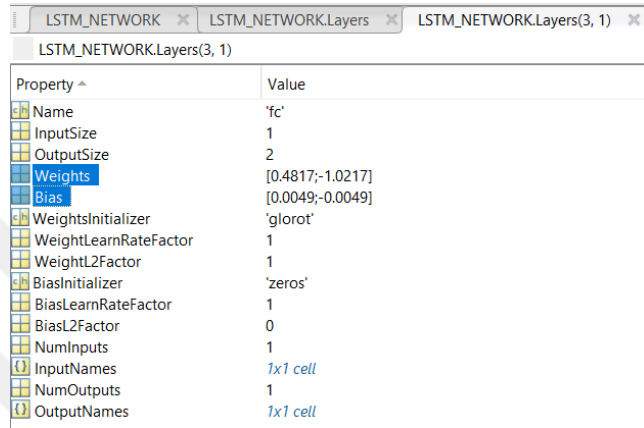
1  --- ERDOĞAN BERKAY TEKİNCAN ---
2  --- Tez Çalışması 2021 ---
3  --- LSTM Neural Network Module ---
4  -----
5  library IEEE;
6  use IEEE.STD_LOGIC_1164.ALL;
7  use IEEE.numeric_std.all;
8  -----
9  entity Neural_Network_LSTM is
10 generic
11 (
12     LSTM_CELL_N      : integer := 400000
13 );
14 port
15 (
16     SYS_CLK           : in  std_logic;
17     NETWORK_EN        : in  std_logic;
18     XT_S              : in  std_logic_vector(31 downto 0);
19     CELL_READY        : out std_logic;
20     CLASSIFY_OUT       : out std_logic;
21     CLASSIFY_VALID    : out std_logic;
22 );
23 end Neural_Network_LSTM;

```

Şekil 3.48 FPGA'de tasarımı tamamlanan UKSB sinir ağı

Test veri setindeki verilerin örnekleme sayısı kadar UKSB hücresi kendisini tekrarladıktan sonra elde edilen h_t çıktısı Şekil 3.45'de görüldüğü üzere bir sınıflandırma (classify) modülüne girmektedir. FPGA tasarımının son aşaması sınıflandırma (classify) katmanıdır.

Elde edilen son h_t çıktısı kullanılarak ve uygun işlemler sonrası sınıflandırma modülü, bu veri setinde darbesel sinyal mevcut (mantıksal '1') veya bu veri seti sadece gürültü içeriyor (mantıksal '0') sonucunu üretmelidir. FPGA üzerinde tasarlanan sınıflandırma modülü MATLAB üzerinde gerçekleştirilen UKSB sinir ağı tasarımı temel almaktadır ve başarılı sonuçlar elde edilebilmesi için gerekli olan ağırlık (weights) ve bias değerleri Şekil 3.49'de gösterildiği gibi Bölüm 3.5'de gerçekleştirilen MATLAB UKSB sinir ağı çalışması sonucu eğitilen UKSB sinir ağından elde edilmiştir.



Property	Value
Name	'fc'
InputSize	1
OutputSize	2
Weights	[0.4817;-1.0217]
Bias	[0.0049;-0.0049]
WeightsInitializer	'glorot'
WeightLearnRateFactor	1
WeightL2Factor	1
BiasInitializer	'zeros'
BiasLearnRateFactor	1
BiasL2Factor	0
NumInputs	1
InputNames	1x1 cell
NumOutputs	1
OutputNames	1x1 cell

Şekil 3.49 MATLAB UKSB sınıflandırma katmanı ağırlık ve bias değerleri

Bu tez çalışmasında 2 adet sınıflandırma olduğundan (bu test verisi darbesel sinyal içeriyor veya bu test verisinde sadece gürültü sinyali mevcut) 2 adet ağırlık ve 2 adet bias değeri vardır. Son verinin işlenmesi ile elde edilen h_t çıktısı bu modülde aşağıdaki eşitliklere uygulanır.

$$\text{Sınıflandırma sonucu (1)} = \text{sigmoid} (h_t * W_1 + b_1) \quad (3.4)$$

$$\text{Sınıflandırma sonucu (2)} = \text{sigmoid} (h_t * W_2 + b_2) \quad (3.5)$$

Elde edilen sonuçlara göre;

Sınıflandırma sonucu (2) $\geq$ Sınıflandırma sonucu (1) ise bu test verisi sinyal içeriyor bilgisi üretilir.

Sınıflandırma sonucu (2) $<$ Sınıflandırma sonucu (1) ise bu test verisi sadece gürültü sinyali içeriyor bilgisi üretilir.

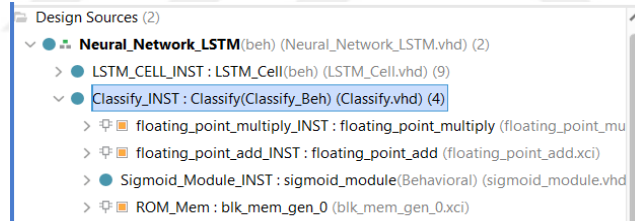
```

1  --- ERDOGAN BERKAY TEKINCAN ---
2  --- Tez Calismasi 2021 ---
3  --- Classify Module ---
4  ---
5  library IEEE;
6  use IEEE.STD_LOGIC_1164.ALL;
7  use IEEE.NUMERIC_STD.ALL;
8  ---
9  entity Classify is
10 port
11 (
12     SYS_CLK      : in std_logic;
13     CLASSIFY_EN   : in std_logic;
14     HT           : in std_logic_vector (31 downto 0);
15     CLASSIFY_OUT  : out std_logic;
16     CLASSIFY_VALID : out std_logic
17 );
18 end Classify;

```

Şekil 3.50 FPGA’de tasarımı tamamlanan sınıflandırma modülü

Şekil 3.50 ve Şekil 3.51’de görüldüğü gibi FPGA üzerinde tasarımı tamamlanan Classify_INST modülü yukarıda belirtilen işlemleri gerçekleştirmektedir. Çarpma ve toplama işlemi için daha önce Şekil 3.28’de ve Şekil 3.29’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği kullanılmaktadır. Sigmoid fonksiyonu hesaplamaları için ise daha önce Şekil 3.30, Şekil 3.32 ve Şekil 3.34’de detayları paylaşılan Floating-Point (7.1) IP çekirdeği ve Block Memory Generator (8.4) IP çekirdeği kullanılmaktadır. Sınıflandırma işlemi iki adet sınıf üzerinde gerçekleştirildiğinden dolayı bu modüle çoklu sınıflandırma problemleri için kullanılan Softmax aktivasyon fonksiyonu yerine sigmoid aktivasyon fonksiyonu kullanılmıştır.



Şekil 3.51 FPGA’de sınıflandırma modülünün hiyerarşik yapısı

Tasarımı gerçekleştirilen UKSB sinir ağı Xilinx Artix-7 (XC7A100TFGG484-3) FPGA üzerinde sentezlenmiştir ve kaynak tüketimi Tablo 3.6’de paylaşılmıştır. Sistemin çalışma frekansı 25 MHz’dir.

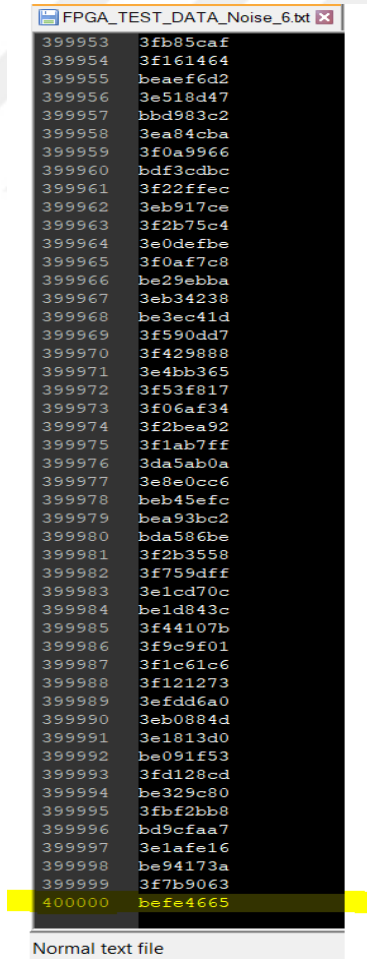
Tablo 3.6 UKSB sinir ağının FPGA kaynak tüketimi

Kaynak Türü	Kullanılabilir Kaynak	Kullanılan Kaynak	Kullanım Yüzdesi %
LUT	63400	11108	17.52
LUTRAM	19000	4	0.02
Flip-flop	126800	4160	3.28
BRAM	135	8	5.93
DSP	240	24	10

3.7. Uzun Kısa-Süreli Bellek Sinir Ağı FPGA Benzetim Çalışmaları

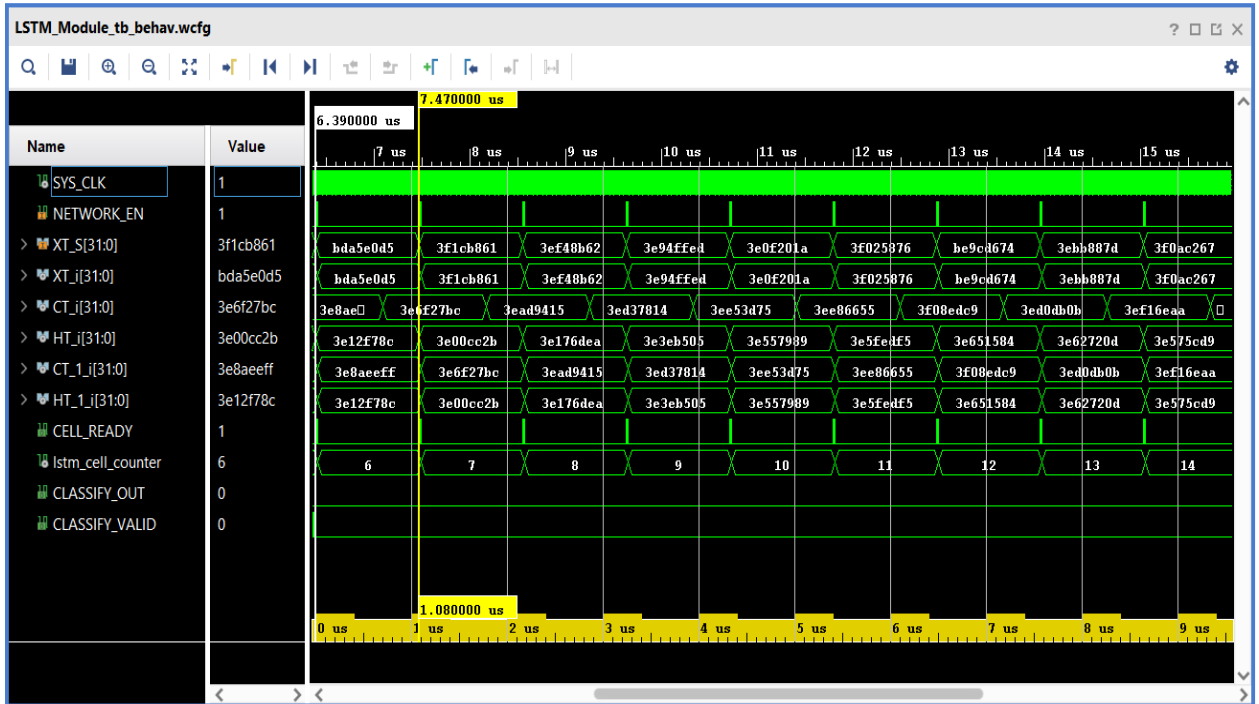
Tasarım çalışmaları tamamlandıktan sonra FPGA benzetim çalışmaları gerçekleştirilmiştir. Bölüm 3.1’de eğitim ve test veri setlerinin nasıl oluşturulduğu ifade edilmiştir. Bu tez çalışması kapsamında 5 dB, 0 dB ve -5 dB SGO seviyelerinde FPGA benzetim çalışmaları gerçekleştirilmiştir ve detaylı sonuçlar Bölüm 0’de paylaşılmıştır. Bu bölümde -5 dB SGO seviyesinde gerçekleştirilen FPGA benzetim çalışması detaylı olarak açıklanmıştır.

Bölüm 3.1’de ifade edildiği gibi -5 dB SGO seviyesinde 100 adet test verisi içeren veri seti oluşturulmuştur. Bu veri setindeki test verilerinin 50 tanesi darbesel sinyal ve gürültü içerirken kalan 50 tanesi sadece gürültü sinyali içermektedir. Bu test veri seti FPGA’de tasarlanan UKSB sinir ağına uygulanmıştır. MATLAB yardımıyla bu veri setindeki test verilerinin örnekleme değerleri tek tek text dosyasına IEEE-754 formatına uygun olarak yazılmaktadır. Şekil 3.52’de bir test verisi ve örnekleme değerleri görülmektedir.



Şekil 3.52 6 numaralı test verisi

Benzetim çalışmaları Vivado Simulator programında gerçekleştirilmiştir. Benzetim çalışmasını gerçekleştirebilmek için VHDL donanım tanımlama dili ile bir testbench hazırlanmıştır. Bu testbench, test veri setinin içerdiği verileri text dosyasından okuyarak tek tek UKSB sinir ağına uygulamaktadır. Şekil 3.53’de FPGA benzetim çalışmasından bir kesit paylaşılmıştır. *SYS_CLK* (Sistem saat işareti) 25MHz’dir. Veriler text dosyasından okunup *XT_S* sinyaline aktarılmaktadır ve *NETWORK_EN* sinyali uyarılmaktadır. Bu sinyal UKSB hücresini aktif hale getirmektedir ve text dosyasından okunan veri UKSB hücresinin *XT_i* girişine uygulanmaktadır.



Şekil 3.53 UKSB FPGA Benzetim Çalışması

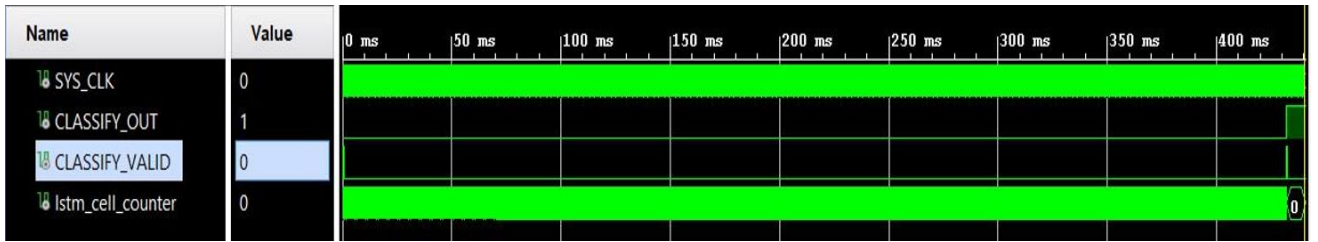
Bölüm 3.6’da anlatıldığı gibi Tablo 3.5’te verilen eşitlikler sonucu *CT_i* ve *HT_i* değerleri hesaplanır ve Şekil 3.53’de görüldüğü üzere ve Şekil 3.46’de açıklandığı gibi *CT_1_i*’e ve *HT_1_i*’e aktarılaraq *CELL_READY* sinyali uyarılır. Bu sinyal uyarıldığında UKSB hücresi hesaplama işlemlerini tamamlanmış ve bir sonraki hücreye aktarılacak bilgileri (*CT* ve *HT*) oluşturmuştur. Bu işlem Şekil 3.53’de görüldüğü üzere 1.08 us sürmektedir. *CELL_READY* sinyali uyarıldıktan sonra testbench yeni *XT_S* verisini UKSB hücresine aktarmaktadır. UKSB hücresi *CT_1_i* ve *HT_1_i* girişleri için bir önceki işlemde hesaplanan verileri kullanmaktadır. Bu şekilde test verisindeki örnekleme sayısı kadar (bu tez çalışmasında 400.000 adet) bu işlem tekrarlanmaktadır. 432 ms sonra 400.000 adet işlem tamamlanır ve hesaplanan son *HT* değeri sınıflandırma (classify) modülüne aktarılır.

Şekil 3.54’de görüldüğü üzere test verisindeki örnekleme sayısı kadar UKSB hücresi işlemleri tekrarladıktan sonra sınıflandırma sonucu elde edilmektedir. *lstm_cell_counter* 400.000 değerine ulaşmıştır ve *CLASSIFY_VALID* sinyali uyarılmıştır. *CLASSIFY_VALID* sinyali uyarıldığı anda *CLASSIFY_OUT* çıkışı mantıksal ‘1’ değerine geçmiştir. Bu, uygulanan veri setinin -5 dB SGO seviyesinde darbesel sinyal içerdiğini ifade etmektedir. Tüm işlemler 432.1 ms’de tamamlanmıştır.



Şekil 3.54 FPGA benzetim çalışması sınıflandırma işlemi

Şekil 3.55, Şekil 3.56, Şekil 3.57 ve Şekil 3.58 gerçekleştirilen bazı benzetim çalışmalarını göstermektedir. Şekil 3.55 ve Şekil 3.57 benzetim sonucuna göre sınıflandırma sonucu (*CLASSIFY_OUT* mantıksal ‘1’) -5 dB SGO seviyesinde darbesel sinyal içerdiğini göstermektedir. Şekil 3.56 ve Şekil 3.58 benzetim sonucuna göre sınıflandırma sonucu ise (*CLASSIFY_OUT* mantıksal ‘0’) -5 dB SGO seviyesinde sadece gürültü sinyali içerdiğini göstermektedir.



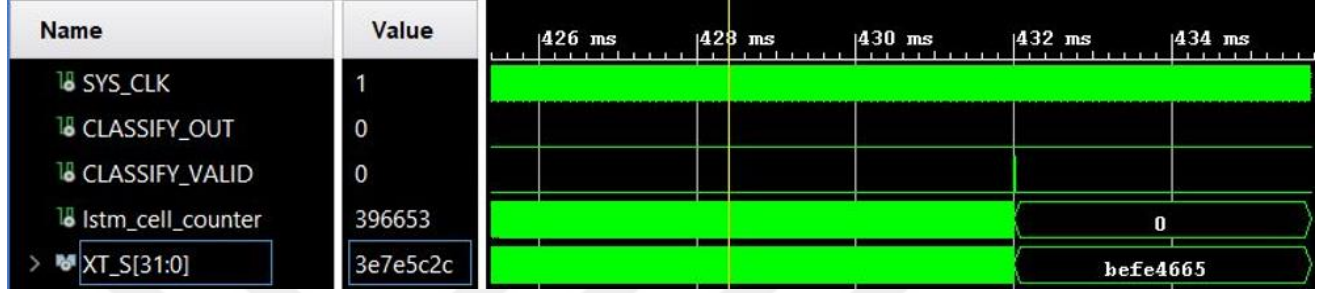
Şekil 3.55 Benzetim çalışması örneği 1 (Darbesel sinyal mevcut)



Şekil 3.56 Benzetim çalışması örneği 2 (Darbesel sinyal mevcut değil)



Şekil 3.57 Benzetim çalışması örnek 3 (Darbesel sinyal mevcut)



Şekil 3.58 Benzetim çalışması örnek 4 (Darbesel sinyal mevcut değil)

Karışıklık Matrisi (Confusion Matrix)			
	Sınıflandırma Sonucu		
		Sınıflandırma Sonucu '1'	Sınıflandırma Sonucu '0'
	Gerçek Değer		
Gerçek Değer	Gerçek Değer '1'	49	1
	Gerçek Değer '0'	0	50
	Doğruluk (Accuracy)	0.99	
	Kesinlik (Precision)	1	
	Duyarlılık (Recall)	0.98	
	F1 Puanı (F1 Score)	0.9899	

Şekil 3.59 FPGA tabanlı UKSB'de 5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi

Karışıklık Matrisi (Confusion Matrix)			
	Sınıflandırma Sonucu		
		Sınıflandırma Sonucu '1'	Sınıflandırma Sonucu '0'
	Gerçek Değer		
Gerçek Değer	Gerçek Değer '1'	48	2
	Gerçek Değer '0'	1	49
	Doğruluk (Accuracy)	0.97	
	Kesinlik (Precision)	0.9796	
	Duyarlılık (Recall)	0.96	
	F1 Puanı (F1 Score)	0.9697	

Şekil 3.60 FPGA tabanlı UKSB'de 0 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi

Karışıklık Matrisi (Confusion Matrix)			
	Sınıflandırma Sonucu		
		Sınıflandırma Sonucu '1'	Sınıflandırma Sonucu '0'
Gerçek Değer	Gerçek Değer '1'	48	2
	Gerçek Değer '0'	4	46
	Doğruluk (Accuracy)	0.94	
	Kesinlik (Precision)	0.9231	
	Duyarlılık (Recall)	0.96	
	F1 Puanı (F1 Score)	0.9412	

Şekil 3.61 FPGA tabanlı UKSB’de -5 dB SGO seviyesinde sınıflandırma için Karışıklık Matrisi

4. SONUÇ VE ÖNERİLER

Elektronik harp destek sistemleri, dost kuvvetleri düşman tehditlerinden korumak için düşman elektromanyetik dalga yayıcılarından gelen tehdit edici sinyalleri tespit ve analiz etmelidir. Çeşitli nedenlerden dolayı hedef sinyallerin tespit edilmesi giderek daha zor hale gelmektedir ve düşük SGO seviyelerinde klasik yöntemler (SYAO yöntemi ile tespit) ile sinyal tespiti yapılamamaktadır.

Bu tez çalışmasında, darbesel sinyal tespit uygulaması için FPGA tabanlı bir Uzun Kısa-Süreli Bellek yapay sinir ağı çözümü önerilmiştir. UKSB sinir ağını eğitmek ve test edebilmek için gerekli olan veri setlerini elde edebilmek amacıyla bir MATLAB/Simulink modeli tasarlanmıştır. Bu MATLAB/Simulink modeli ile çeşitli SGO seviyelerinde darbesel sinyaller elde edilmiştir ve bu sinyaller örneklendirilerek eğitim ve test veri setleri elde edilmiştir. Örneklenen değerler üzerinde herhangi bir işlem gerçekleştirilmeden direkt olarak ham veriler üzerinde işlem yapılmıştır. Eğitim amaçlı elde edilen veri setleri kullanılarak MATLAB UKSB sinir ağı başarıyla eğitilmiştir. MATLAB/Simulink modeli ile elde edilen test veri setleri MATLAB UKSB sinir ağına uygulandığında 5 dB ve 0 dB SGO seviyesinde **%100** doğruluk değeri, -5 dB SGO seviyesinde **%98** doğruluk değeri elde edilmiştir. Karışıklık Matrisleri (Confusion Matrix) sırasıyla Şekil 3.21, Şekil 3.22 ve Şekil 3.23’de paylaşılmıştır.

Bu tez çalışmasında, UKSB sinir ağı FPGA’de tasarlanmıştır ve sentezlenmiştir. FPGA’de UKSB sinir ağı tasarım aşamasında MATLAB UKSB sinir ağı eğitimi sonucu elde edilen katsayılar kullanılmıştır. MATLAB/Simulink modeli ile elde edilen test veri setleri FPGA tabanlı UKSB sinir ağına uygulanmıştır. FPGA benzetim sonuçlarına göre

5 dB SGO seviyesinde **%99** doğruluk değeri, 0 dB SGO seviyesinde **%97** doğruluk değeri ve -5 dB SGO seviyesinde **%94** doğruluk değeri elde edilmiştir. Karışıklık Matrisleri (Confusion Matrix) sırasıyla Şekil 3.59, Şekil 3.60 ve Şekil 3.61’da paylaşılmıştır. Bu UKSB sinir ağı çalışması Xilinx Artix-7 (XC7A100TFGG484-3) FPGA’de sentezlenmiştir ve kaynak tüketimi ve çalışma frekansı gibi performans ölçütleri Bölüm 3.6’da paylaşılmıştır.

Bu tez çalışmasında önerilen FPGA tabanlı UKSB yapay sinir ağını, performans ölçütleri açısından SYAO yöntemi ile karşılaştırmak amacıyla belirlenen sabit eşik değeri ile darbesel sinyal tespit çalışmasına yönelik bir MATLAB/Simulink çalışması ve SYAO yönteminin FPGA tasarım ve benzetim çalışmaları yapılmıştır. SYAO yöntemi ile yüksek SGO seviyelerinde (10 dB ve 5 dB) başarılı sinyal tespitleri yapılabildiği gözlemlenmiştir. Ancak bu yöntem ile 0 dB ve -5 dB SGO seviyelerinde başarılı sonuçlar elde edilememiştir.

SYAO yöntemi ile darbesel sinyal tespit çalışmalarında 0 dB ve -5 dB gibi düşük SGO seviyelerinde başarısız sonuçlar elde edilmesine karşılık bu tez çalışmasında sunulan FPGA tabanlı UKSB sinir ağı, aynı SGO seviyelerinde sırasıyla **%97** ve **%94** doğruluk değeri ile darbesel sinyal tespiti gerçekleştirmiştir. FPGA benzetim sonuçlarına göre -5 dB gibi düşük SGO seviyelerinde SYAO eşik değeri yöntemi ile tespit edilemeyen darbesel sinyallerin önerilen FPGA tabanlı UKSB sinir ağı ile tespit edilebilmesi umut verici bir gelişmedir.

Bu tez çalışmasında elde edilen performans ölçütleri göz önüne alındığında, önerilen FPGA tabanlı UKSB sinir ağının 432 ms gecikme ile darbesel sinyal tespiti gerçekleştirdiği görülmektedir. Yalnızca 3 sistem saat işareti sonra tespit işlemini gerçekleştirebilen SYAO yöntemine göre bu süre oldukça fazladır. Bu tez çalışmasında çeşitli SGO seviyeleri için UKSB hücresi ayrı ayrı eğitilmiştir ve ayrı ayrı FPGA benzetim çalışması ile test edilmiştir. Ayrıca bu tez çalışmasındaki darbesel sinyalin darbe genişliği sabittir. Farklı darbe genişlikleri için farklı UKSB hücrelerinin eğitilmesi ihtiyacı doğmaktadır. Farklı SGO seviyeleri, farklı darbe genişlikleri ve farklı güç seviyeleri gibi sinyal karakteristikleri için çok sayıda UKSB hücresinin eğitilmesi ve FPGA’de uygulanması gerekmektedir ve gelen darbesel sinyalin uygun (SGO seviyesinin kestirilip o SGO seviyesi için eğitilen UKSB hücresine) yönlendirilmesi gerekmektedir. Bu ekstra bir SGO kestirim çalışması gerektirmektedir. Bunun için çok daha detaylı ve ekstra zaman gerektiren çevrimdışı bir yapay sinir ağı eğitim çalışması gerekmektedir ve çok sayıda UKSB hücresinin FPGA’de uygulanması FPGA kaynak problemine yol açabilmektedir.

Bu tez çalışmasında SYAO yöntemi için gerçekleştirilen eşik değeri hesaplama işlemleri çevrimdışı olarak gerçekleştirilmektedir. Ancak belirtilen eşik hesaplama yöntemi FPGA içerisinde de uygulanabilir. Bu, sıcaklık değişimlerinde ya da almaç ayarları değişimleri gibi eşik değerinin tekrar hesaplanması gereken benzeri konularda gerçek zamanlı olarak tekrar hesaplanabilmesine de olanak sağlayarak hiçbir çevrimdışı işlem gerektirmeyerek UKSB sinir ağına göre avantaj sağlamaktadır.

SYAO yöntemi gerçek zamanlı çalışma gereksinimlerine uygun olarak 100 MHz örnekleme hızıyla, ADC ile elde edilen veriler üzerinde çalışabilirken, bu tez çalışmasında önerilen UKSB mimarisi benzer örnekleme hızıyla elde edilen veriler üzerinde istenilen performansı sağlayamamaktadır. Bu sorunu ortadan kaldırmak için 2 çözüm yöntemi önerilmektedir; Örnekleme hızını artırarak daha çok veri elde etmek veya UKSB mimarisini değiştirmek. MATLAB ile gerçekleştirilen çalışmalarda FPGA’de uygulanan UKSB tipini değiştirerek (Çift Yönlü UKSB) veya UKSB hücreleri arasında bilgi aktarımını sağlayan gizli birimlerin sayısını FPGA’de uygulanan UKSB sinir ağına göre 5 kat artırarak 100 MHz örnekleme hızıyla elde edilen veriler ile başarılı sonuçlar elde edilebildiği görülmüştür. Ancak MATLAB üzerinde gerçekleştirilen bu UKSB mimarisi değişikliklerinin yüksek matematiksel işlem gereksinimleri, FPGA kaynak kullanımı ve maksimum çalışma frekansı gibi performans ölçütleri göz önüne alındığında donanım (FPGA) uygulaması için uygun değildir. Bu tez çalışmasında örnekleme hızı 10 kat artırılmıştır ve bu şekilde FPGA üzerinde başarılı sinyal tespit çalışmaları gerçekleştirilmiştir. Ancak bu ADC örnekleme hızında (1 GSPS), bu tez çalışmasında ulaşılan maksimum sistem saat işareti (25 MHz) ile ADC’den verilerin anlık olarak elde edilmesi uygulanabilir değildir. Örnekleme hızının 10 kat artırılması ile bu tez çalışmasında elde edilen 400.000 adet örnekleme değerinin geçici olarak saklanması ve bu tez çalışmasında sunulan UKSB sinir ağına uygulanması gerekmektedir. FPGA içerisindeki geçici hafıza birimlerinin bu boyutta bir veri setini saklayabilmesi çoğu FPGA ailesi için uygulanabilir değildir. Çözüm olarak harici bir bellek ile saklanması önerilebilir ancak bu da yine 25 MHz sistem saat işareti ile harici bellekten verileri okuma sorununa neden olacaktır. Ek olarak gerçek zamanlı radar sinyal tespit uygulamaları için verilerin bir bellek üzerinde saklandıktan sonra işlenmesi uygun değildir. Bu tez çalışmasında önerilen UKSB sinir ağının sinyal tespit süresinin 432 ms olduğu düşünüldüğünde, sinyal tespit işlemleri sırasında gelecek diğer veri setlerinin işlenememesine ve kaçırılmasına sebep olacaktır.

Bu tez çalışmasında önerilen mimarinin gerçek zamanlı uygulamalar için iyileştirilmesi gerektiği sonucuna ulaşılmaktadır. Daha hızlı bir FPGA ailesinin kullanılması, FPGA tasarımında kullanılan IP çekirdeklerinin daha yüksek hızda çalışması için optimize edilmesi ve boru hattı (pipeline) tasarımı gibi değişiklikler ile sistem saat işaretinin iyileştirilebilmesi öngörülmektedir. Daha düşük örnekleme hızı ancak daha yüksek eğitim veri seti sayısı ile UKSB sinir ağının eğitilmesi yüksek eğitim süresi de göz önüne alınarak değerlendirilebilir. Ayrıca mevcut tasarımdaki UKSB hücrelerinin arasındaki bilgi aktarımı sağlayan gizli birimlerin sayısının artırılarak veya literatürdeki çeşitli UKSB mimarilerinin (Çift yönlü UKSB ve Geçitli Tekrarlayan Birim) FPGA’de uygulanabilir hale getirilmesi ile daha düşük örnekleme hızıyla da darbesel sinyal tespiti gerçekleştirilmesi öngörülmektedir.

Mevcut sonuçlar, bu tez çalışmasında önerilen FPGA tabanlı UKSB yapay sinir ağının, teorik çalışmaları destekler nitelikte, bir donanım (FPGA) üzerinde gerçekleştirilerek klasik eşik değeri yönteminin (SYAO yöntemi) başarılı sonuçlar elde edemediği düşük SGO seviyelerinde sinyal tespiti gerçekleştirebildiğini ancak gerçek zamanlı bir elektronik harp sistemi radar almacı gereksinimlerini karşılayamadığını göstermektedir. Gelecekteki çalışmalarımız, bu bölümde önerilen iyileştirmeleri uygulayan yeni çalışmalara yönelik olacaktır. Bu iyileştirmelerin uygulanması ile gelecek çalışmalarda, gerçek zamanlı olarak ve daha düşük SGO seviyelerinde başarılı bir şekilde darbesel sinyallerin tespiti gerçekleştirilebileceği öngörülmektedir.

KAYNAKLAR

- [1] H. Merkezi, “Elektronik Harp Nedir,” SavunmaSanayiST, 02-Apr-2020. [Online]. Available: <https://www.savunmasanayist.com/elektronik-harp-nedir> [Accessed: 15-Dec-2021].
- [2] J.-W. Shin, K.-H. Song, K.-S. Yoon, and H.-N. Kim, “Weak Radar Signal Detection Based on Variable Band Selection,” IEEE Transactions on Aerospace and Electronic Systems, vol. 52, no. 4, pp. 1743–1755, 2016.
- [3] M. Bayram, “Radar Nedir ve Nasıl Çalışır,” Mühendis Beyinler, 09-Oct-2017. [Online]. Available: <https://www.muhendisbeyinler.net/radar-nedir-ve-nasil-calisir> [Accessed: 15-Dec-2021].
- [4] C. Ilioudis, “Introduction to Radar Signal Processing,” [Online]. Available: <https://udrc.eng.ed.ac.uk/sites/udrc.eng.ed.ac.uk/files/attachments/Introduction%20Radar%20signal%20processing.pdf> [Accessed: 15-Dec-2021].
- [5] “Principles of Radar Systems,” 38542-F0, Aug-2015. [Online]. Available: https://labvolt.festo.com/downloads/38542_F0.pdf [Accessed: 15-Dec-2021].
- [6] The Central Flying School (CFS), “AP3456 – 11-2 - Pulse Radar.” [Online]. Available: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/857309/Volume_11_Radar.pdf [Accessed: 15-Dec-2021].
- [7] “Basic Principles of Radar,” Communication System Engineering (Radar Systems). [Online]. Available: http://vssut.ac.in/lecture_notes/lecture1429093042.pdf [Accessed: 16-Dec-2021].
- [8] “Radar Basic Principles,” Khulna University of Engineering & Technology. [Online]. Available: <https://www.kuet.ac.bd/webportal/ppmv2/uploads/1575340548Radar%20Basics.pdf> [Accessed: 16-Dec-2021].
- [9] T. Srinivas and N. Swetha, “Minimum Detectable Signal,” Radar Systems, Malla Reddy College of Engineering & Technology. [Online]. Available: https://mrcet.com/downloads/digital_notes/ECE/IV%20Year/RADAR%20SYSTEMS.pdf [Accessed: 16-Dec-2021].
- [10] F. C. Akyon, M. A. Nuhoglu, Y. K. Alp, and O. Arikan, “Multi-task learning based Joint Pulse Detection and Modulation Classification,” 2019 27th Signal Processing and Communications Applications Conference (SIU), Aug. 2019.
- [11] J. Fu, J. Chu, P. Guo, and Z. Chen, “Condition monitoring of wind turbine gearbox bearing based on deep learning model,” IEEE Access, vol. 7, pp. 57078–57087, 2019.

- [12] S. Wang, P. Lin, R. Hu, H. Wang, J. He, Q. Huang, and S. Chang, "Acceleration of LSTM with structured pruning method on FPGA," *IEEE Access*, vol. 7, pp. 62930–62937, 2019.
- [13] H. Zeng, R. Chen, C. Zhang, and V. Prasanna, "A framework for generating high throughput CNN implementations on FPGAs," *Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pp. 117–126, Feb. 2018.
- [14] M. Rizakis, S. I. Venieris, A. Kouris, and C.-S. Bouganis, "Approximate FPGA-based LSTMs under computation time constraints," *Applied Reconfigurable Computing. Architectures, Tools, and Applications*, pp. 3–15, 2018.
- [15] M. A. Nuhoglu, Y. K. Alp, and F. C. Akyon, "Deep learning for radar signal detection in Electronic Warfare Systems," *2020 IEEE Radar Conference (RadarConf20)*, Sep. 2020.
- [16] L. Wang, J. Tang, and Q. Liao, "A study on radar target detection based on Deep Neural Networks," *IEEE Sensors Letters*, vol. 3, no. 3, pp. 1–4, Mar. 2019.
- [17] A. Sailaja, A. K. Sahoo, G. Panda, and V. Baghel, "A recurrent neural network approach to Pulse Radar Detection," *2009 Annual IEEE India Conference*, 2009.
- [18] K. Bhattacharyya and K. Kumar Sarma, "Automatic Target Recognition (ATR) system using recurrent neural network (RNN) for pulse radar," *International Journal of Computer Applications*, vol. 50, no. 23, pp. 33–39, Jul. 2012.
- [19] H. Nguyen, D. Ngo, and V. Do, "Deep Learning for Radar Pulse Detection," *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods*, pp. 32–39, 2019.
- [20] G.-H. Lee, J. Jo, and C. H. Park, "Jamming prediction for radar signals using machine learning methods," *Security and Communication Networks*, vol. 2020, pp. 1–9, Jan. 2020.
- [21] Sahil Abrol and Rita Mahajan, "Artificial Neural Network Implementation on FPGA Chip," *International Journal of Computer Science and Information Technology Research*, Vol. 3, Issue 1, pp. 11-18, Jan-Mar 2015.
- [22] P. Škoda, T. Lipic, Ā. Srp, B. M. Rogina, K. Skala, and F. Vajda, "Implementation Framework for Artificial Neural Networks on Fpga," *2011 Proceedings of the 34th International Convention MIPRO*, May 2011, pp. 274–278.
- [23] R. Raeisi and A. Kabir, "Implementation of Artificial Neural Network on FPGA," [Online]. Available: <http://ilin.asee.org/Conference2006program/Papers/Raeisi-P59.pdf> [Accessed: 15-Dec-2021].
- [24] A. X. M. Chang, B. Martini, and E. Culurciello, "Recurrent Neural Networks Hardware Implementation on FPGA," *arXiv.org*, 04-Mar-2016. [Online]. Available: <https://arxiv.org/abs/1511.05552> [Accessed: 15-Dec-2021].

- [25] H. Rashid, N. K. Manjunath, H. Paneliya, M. Hosseini, W. D. Hairston, and T. Mohsenin, "A low-power LSTM processor for multi-channel brain EEG Artifact Detection," 2020 21st International Symposium on Quality Electronic Design (ISQED), p. 105, 2020.
- [26] Y. Zhang, C. Wang, L. Gong, Y. Lu, F. Sun, C. Xu, X. Li, and X. Zhou, "A Power-Efficient Accelerator Based on FPGAs for LSTM Network," 2017 IEEE International Conference on Cluster Computing (CLUSTER), pp. 629–630, 2017.
- [27] Z. Sun, Y. Zhu, Y. Zheng, H. Wu, Z. Cao, P. Xiong, J. Hou, T. Huang, and Z. Que, "FPGA acceleration of LSTM based on data for test flight," 2018 IEEE International Conference on Smart Cloud (SmartCloud), 2018.
- [28] Q. Liu, T. Liang, Z. Huang, and V. Dinavahi, "Real-Time FPGA-Based Hardware Neural Network for Fault Detection and Isolation in More Electric Aircraft," IEEE Access, vol. 7, pp. 159831–159841, Nov. 2019.
- [29] "Sequence Classification Using Deep Learning," MathWorks. [Online]. Available: <https://www.mathworks.com/help/deeplearning/ug/classify-sequence-data-using-lstm-networks.html> [Accessed: 16-Dec-2021].
- [30] "Long Short-Term Memory Networks," MathWorks. [Online]. Available: <https://www.mathworks.com/help/deeplearning/ug/long-short-term-memory-networks.html> [Accessed: 16-Dec-2021].
- [31] C. Jiang, Y. Chen, S. Chen, Y. Bo, W. Li, W. Tian, and J. Guo, "A Mixed Deep Recurrent Neural Network for MEMS Gyroscope Noise Suppressing," Electronics, vol. 8, no. 2, p. 181, Feb. 2019.