

**FPGA TABANLI ALGILAYICI DOĐRUSALLAŐTIRMA
UYGULAMASI**

FPGA BASED SENSOR LINEARIZATION APPLICATION

ABDÜLGAFUR MÜJDECİ

Başkent Üniversitesi
Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin
ELEKTRİK-ELEKTRONİK Mühendisliği Anabilim Dalı İçin Öngördüğü
YÜKSEK LİSANS TEZİ
olarak hazırlanmıştır.

2013

“FPGA TABANLI ALGILAYICI DOĞRUSALLAŞTIRMA UYGULAMASI” başlıklı bu çalışma, jürimiz tarafından, 08/02/2013 tarihinde, **ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI 'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Başkan :
Yrd. Doç. Dr. Derya YILMAZ

Üye (Danışman) :
Doç. Dr. Hamit ERDEM

Üye :
Yrd. Doç. Dr. Metin YILDIZ

ONAY

21/02/2013

Prof. Dr. Emin AKATA
Fen Bilimleri Enstitüsü Müdürü

ÖZ

FPGA TABANLI ALGILAYICI DOĞRUSALLAŞTIRMA UYGULAMASI

Abdülğafur MÜJDECİ

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Algılayıcılar, fiziksel büyüklükleri ölçüp, bu büyüklükleri elektriksel devrelerin okuyabileceği voltaj, akım, direnç gibi niceliklere çevirebilen temel ölçüm elemanlarıdır. Algılayıcıların çoğu doğrusal davranış göstermediğinden, alınan değerlerden, algılanan fiziksel değişimi hesaplamak için farklı algoritmalar ve devre elemanları kullanılabilir. Bu tez çalışmasında, doğrusal olmayan bir algılayıcıdan gelen veri kullanılarak, ölçülen fiziksel değeri hesaplamak için Programlanabilir Kazançlı Yükselteç ve Yapay Sinir Ağları yapılarının özellikleri birleştirilmiş, Alan Programlanabilir Kapı Dizileri (İng. Field Programmable Gate Array, FPGA) tabanlı uygulama gerçekleştirilmiştir. Algılayıcı olarak Negatif Sıcaklık Katsayısı (İng. Negative Temperature Coefficient, NTC) tipindeki sıcaklık algılayıcısı kullanılmıştır. Önerilen yöntem, yazılım ve donanım ortamlarında uygulanmış ve elde edilen sonuçlar, diğer yöntemlerin sonuçlarıyla karşılaştırılmıştır. Uygulamanın başarısı, ortalama karesel hata, işlem zamanı, bellek kullanımı ve maliyet kriterlerine göre detaylı olarak tartışılmıştır.

ANAHTAR SÖZCÜKLER: algılayıcı doğrusallaştırma, YSA, PGA, FPGA, NTC, MLP

Danışman: Doç.Dr. Hamit ERDEM, Başkent Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü.

ABSTRACT

FPGA BASED SENSOR LINEARIZATION APPLICATION

Abdülğafur MÜJDECİ

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Sensors are fundamental measurement elements used in wide variety of applications. A sensor is a converter that measures a physical quantity and converts it into a signal which can be read by an electrical instrument. Since many kinds of sensors have non-linear input-output behaviour, to be able to calculate the measured physical quantity by using the sensor data, special electrical circuits and algorithms are required. In the scope of this thesis, the advantages of the Programmable Gain Amplifier (PGA) and Artificial Neural Networks (ANN) are combined and realized experimental applications on an FPGA platform to linearize a non-linear sensor behaviour. The Negative Temperature Coefficient (NTC) type temperature sensor is chosen. The simulation of the suggested linearization method is realized in the software environment and afterwards applied on an FPGA experiment set. The resulting performance of the suggested method is compared with other methods in respect to mean squared error, process speed, memory usage and cost.

KEYWORDS: sensor linearization, ANN, PGA, FPGA, NTC, MLP

Supervisor: Doç.Dr. Hamit ERDEM, Başkent Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü.

İÇİNDEKİLER LİSTESİ

Sayfa

ÖZ	i
ABSTRACT.....	ii
İÇİNDEKİLER LİSTESİ.....	iii
ŞEKİLLER LİSTESİ.....	vi
ÇİZELGELER LİSTESİ	ix
SİMGELER VE KISALTMALAR LİSTESİ.....	x
1. GİRİŞ.....	1
2. ALGILAYICI DOĞRUSALLAŞTIRMA.....	6
2.1. Ters Fonksiyon Alma Yöntemi.....	7
2.2. Başvuru Çizelgesi Yöntemi.....	9
2.3. Kısmi Parçalı Doğrusallaştırma (KPD) Yöntemi	11
2.4. Yapay Sinir Ağlarının Kullanılması Yöntemi	13
2.5. Hibrit Yöntemler.....	14
3. YAPAY SİNİR AĞLARI	15
3.1. Giriş.....	15
3.2. Biyolojik Nöron	15
3.3. Yapay Sinir Ağı.....	17
3.3.1. Yapay nöron modeli	17
3.3.1.1. Doğrusal fonksiyon.....	18
3.3.1.2. Eşik değer fonksiyonu	19
3.3.1.3. Kısmi doğrulaştırılmış eşik değer fonksiyonu	19
3.3.1.4. Lojistik sigmoid fonksiyonu.....	20
3.3.1.5. Tanjant sigmoid fonksiyonu.....	21
3.3.1.6. Gauss fonksiyonu.....	22
3.3.2. Yapay sinir ağlarının yapısı	22
3.3.2.1. YSA'ların genel özellikleri.....	23
3.3.2.2. YSA'ların avantajları.....	24
3.3.2.3. YSA'ların dezavantajları	25
3.3.3. YSA modelleri.....	26
3.3.3.1. Çok katmanlı perseptron (MLP)	26
3.3.4. Geri yayılım (back propagation) algoritması	27
4. PROGRAMLANABİLİR KAZANÇLI YÜKSELTEÇ (PGA).....	32
4.1. Programlanabilir Kazançlı Yükselteçlere Genel Bakış.....	32
4.2. PGA Tasarımı.....	34
4.3. Uygulamada Kullanılan PGA'nın Yapısı	36
5. ALAN PROGRAMLANABİLİR KAPI DİZİLERİ (FPGA)	37
5.1. Programlanabilir Donanımların Tarihsel Gelişimi	37
5.2. FPGA'lerle Tasarıma Genel Bakış	38

5.3.	FPGA'lerin Genel Yapısı	39
5.4.	Tasarım Akışları	41
5.4.1.	Tasarım girdisi.....	41
5.4.2.	Sentez	42
5.4.3.	Yerleştirme ve rotalama.....	42
5.4.4.	Konfigürasyon	42
6.	TERMİSTÖRLER.....	43
6.1.	Termistörlere Genel Bakış.....	43
6.1.1.	Akım-voltaj modu.....	43
6.1.2.	Zaman-akım modu	44
6.1.3.	Sıcaklık-direnç modu.....	45
7.	FPGA TABANLI ALGILAYICI DOĞRUSALLAŞTIRMA UYGULAMALARI. 48	
7.1.	Giriş.....	48
7.2.	Benzetim Çalışmaları	49
7.2.1.	KPD yöntemi kullanılan doğrusallaştırma sisteminin benzetim uygulaması.....	49
7.2.1.1.	KPD yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi.....	50
7.2.2.	PGA+KPD yöntemi kullanılan doğrusallaştırma sisteminin benzetim uygulaması.....	53
7.2.2.1.	PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi.....	53
7.2.3.	YSA yöntemi kullanılan doğrusallaştırma sistemin benzetim uygulaması	56
7.2.3.1.	YSA yöntemi kullanılan sistemdeki ağın eğitilmesi.....	57
7.2.3.2.	YSA yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi.....	60
7.2.4.	PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin benzetim uygulaması.....	61
7.2.4.1.	PGA + YSA yöntemi kullanılan sistemdeki ağın eğitilmesi.....	62
7.2.4.2.	PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi.....	65
7.3.	Donanım Çalışmaları.....	67
7.3.1.	Algılayıcı benzetim devresi.....	67
7.3.2.	Kullanılan PGA'nın özellikleri.....	70
7.3.3.	Kullanılan ADC'nin özellikleri.....	72
7.3.4.	Kullanılan FPGA platformunun özellikleri	73
7.3.5.	KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerinde gerçekleşmesi.....	79

7.3.6. PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım uygulaması	81
7.3.7. YSA yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerinde gerçekleşmesi.....	83
7.3.8. PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerinde gerçekleşmesi.....	84
7.4. Benzetim ve Donanım Uygulamaları Sonuçlarının Değerlendirilmesi	87
8. SONUÇ VE TARTIŞMA.....	91
KAYNAKLAR LİSTESİ	93

ŞEKİLLER LİSTESİ

Sayfa

Şekil 2.1 Algılayıcı doğrusallaştırma akış diyagramı	6
Şekil 2.2 Ters fonksiyon yönteminin grafiksel ifadesi (Kraska, [4] 'dan değiştirilerek)	8
Şekil 2.3 Başvuru çizelgesi yönteminin akış şeması (Kraska, [4] 'dan değiştirilerek).....	10
Şekil 2.4 BÇ yönteminin işlem akış şeması	11
Şekil 2.5 Transdüser grafiğinde belirlenen kısmi doğrusallaştırılmış parçalar	12
Şekil 2.6 KPD yönteminin işlem akış şeması.....	13
Şekil 3.1 Sinir Hücresi (Nöron) [24]	16
Şekil 3.2 Yapay Nöron Modeli	17
Şekil 3.3 $\alpha = 1$ için doğrusal fonksiyonun grafiği.....	19
Şekil 3.4 $\beta=0$ için eşik değer fonksiyonu.....	19
Şekil 3.5 $\alpha = 0$ ve $\beta=1$ için kısmi doğrusallaştırılmış eşik değer fonksiyonu	20
Şekil 3.6 Lojistik sigmoid fonksiyonunu	21
Şekil 3.7 Tanjant sigmoid fonksiyonu	22
Şekil 3.8 Gauss fonksiyonu	22
Şekil 3.9 İki ara katmanlı bir YSA örneği	23
Şekil 3.10 Örnek bir MLP YSA modeli.....	27
Şekil 3.11 Geri yayılım uygulanan MLP ağı.....	28
Şekil 4.1 Veri toplama sistemlerinde PGA.....	33
Şekil 4.2 PGA'ların genel kullanım mantığı	34
Şekil 4.3 Düşük performanslı bir PGA tasarımı (Jung,[30]'dan değiştirilerek)	35
Şekil 4.4 R_{ON} direncinin etkisini en aza indiren alternatif PGA tasarımı.....	36
Şekil 5.1 Tipik bir mantıksal tasarım devresi[31]	39
Şekil 5.2 FPGA'nın dahili yapısı (Xilinx Inc.).....	40
Şekil 6.1 Akım-voltaj modu NTC karakteristiği eğrisi [33].....	44
Şekil 6.2 Termistörün zaman-akım karakteristiği [33].....	45
Şekil 6.3 10k Ω 'luk bir NTC'nin sıcaklık-direnç eğrisi [33].....	46
Şekil 7.1 BÇ yöntemiyle benzetimi yapılan sistemin gösterimi	49
Şekil 7.2 BÇ yöntemi kullanılan sistemin benzetim sonuçlarında hesaplanan hatanın -55 °C - +150 °C aralığındaki dağılımı	51
Şekil 7.3 KPD yöntemi kullanılan doğrusallaştırma sisteminin benzetimi sırasında kullanılan test akış şeması.....	52
Şekil 7.4 PGA + KPD yöntemi kullanılan sistemin benzetimi sırasında kullanılan test akış diyagramı	54
Şekil 7.5 PGA + KPD tabanlı sistemin benzetim sonucunda görülen hatanın -55 °C - +150°C aralığındaki dağılımı	55
Şekil 7.6 Tanjant sigmoid fonksiyonu ve tanjant sigmoid fonksiyonunu yaklaşımının eğrileri (Çavuşlu, [34]'dan değiştirilerek).....	56
Şekil 7.7 YSA tabanlı algılayıcı doğrusallaştırma sistemi	57
Şekil 7.8 YSA tabanlı sistemde kullanılan ağı yapısı.....	57
Şekil 7.9 Normalizasyon ve de-normalizasyon bağıntıları	58
Şekil 7.10 YSA yöntemi kullanılan doğrusallaştırma sisteminin benzetimi sırasında kullanılan test akış diyagramı	60
Şekil 7.11 YSA yöntemi kullanılan sistemin benzetim sonucunda görülen hatanın - 55 °C – 150°C aralığındaki dağılımı	61

Şekil 7.12 Benzetimi yapılan PGA + YSA yöntemi kullanılan doğrusallaştırma sistemi	62
Şekil 7.13 Transdüserin voltaj-sıcaklık eğrisi ile 8 bölgede PGA ile yükseltilmiş voltaj-sıcaklık eğrisi.....	63
Şekil 7.14 PGA'lı YSA uygulamasında kullanılan MLP yapısı	63
Şekil 7.15 PGA + YSA yöntemi kullanılan sistemin benzetimi sırasında kullanılan test akış diyagramı	66
Şekil 7.16 PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin sonuçlarında görülen hatanın -55 °C – 150°C aralığındaki dağılımı	67
Şekil 7.17 Donanımda gerçekleştirilen algılayıcı doğrusallaştırma uygulaması akış şeması	67
Şekil 7.18 -55°C - +150°C sıcaklık aralığında algılayıcı voltajını ($V_{\text{Algılayıcı}}$) 2V – 0V aralığına haritalayan devre	68
Şekil 7.19 $R = 43K$ için Sıcaklık - $V_{\text{Algılayıcı}}$ eğrisi	68
Şekil 7.20 Uygulamada kullanılan PGA'nın şematiği.....	70
Şekil 7.21 ADS8342 ADC entegresinin blok diyagramı	72
Şekil 7.22 ADS8342 zamanlama diyagramı	73
Şekil 7.23 DE0 geliştirme kartının üstten görünümü [32].....	74
Şekil 7.24 Donanımın Blok Diyagramı	75
Şekil 7.25 Deney ortamı	76
Şekil 7.26 Kullanılan donanıma yakından bakış	76
Şekil 7.27 DAC ile üretilen algılayıcı voltajı	77
Şekil 7.28 DAC ile üretilen algılayıcı voltajının PGA ile yükseltilmesi sonucu oluşan sinyal	78
Şekil 7.29 Donanım üzerinde gerçekleştirilen KPD yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı	79
Şekil 7.30 KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerindeki testinin gösterimi.....	80
Şekil 7.31 Donanım üzerinde gerçekleştirilen PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı	81
Şekil 7.32 PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerindeki testinin akış diyagramı.....	82
Şekil 7.33 YSA yöntemi kullanılan sistemin donanım üzerindeki testinin akış diyagramı.....	83
Şekil 7.34 Donanım üzerinde gerçekleştirilen YSA yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı	84
Şekil 7.35 Donanım üzerinde gerçekleştirilen PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı	85
Şekil 7.36 PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerindeki testinin akış diyagramı.....	86
Şekil 7.37 Benzetim sonucunda elde edilen ortalama karesel hatalar.....	88
Şekil 7.38 Donanımsal uygulama sonucunda elde edilen ortalama karesel hatalar	89
Şekil 7.39 Donanımsal uygulama sonucunda elde edilen ortalama işlem süreleri	89

Şekil 7.40 Donanımsal uygulama sonucunda elde edilen bellek kullanım miktarları	90
---	----

ÇİZELGELER LİSTESİ

	<u>Sayfa</u>
Çizelge 3-1 Biyolojik ve Yapay Sinir Hücreleri arasındaki analogi	17
Çizelge 7-1 Başvuru çizelgesi kayıtlarından örnekler	50
Çizelge 7-2 Eğitim verisi örnekleri	58
Çizelge 7-3 Beş farklı eğitimde elde edilen ortalama karesel hatalar (okh) ve eğitim süreleri.....	59
Çizelge 7-4 Eğitim 4 sonucunda elde edilen ağırlık ve eşik değerleri.....	59
Çizelge 7-5 Seçilen eğitimlere ait ortalama karesel hata ve eğitim süreleri.....	64
Çizelge 7-6 Eğitim sonunda elde edilen ağırlık ve eşik değerleri.....	64
Çizelge 7-7 PGA kazançlarına ait gerilim aralıkları	78

SİMGELER VE KISALTMALAR LİSTESİ

ADC	Analog to Digital Converter (Analogdan sayısala dönüştürücü)
AGC	Automatic Gain Control (Otomatik kazanç kontrolü)
BÇ	Başvuru Çizelgesi
BP	Back Propagation (Geri yayılım)
DAC	Digital to Analog Converter (Sayısalda analoga dönüştürücü)
DC	Direct Current (Doğru Akım)
FPGA	Field Programmable Gate Array (Alan programlanabilir kapı dizileri)
MLP	Multi Layer Perceptron (Çok katlı perseptron)
MSE	Mean Squared Error (Ortalama karesel hata)
NTC	Negative Temperature Coefficient (Negatif sıcaklık katsayısı)
OKH	Ortalama Karesel Hata
PGA	Programmable Gain Amplifier (Programlanabilir kazançlı yükselteç)
RAM	Random Access Memory (Rasgele erişilebilir bellek)
RTD	Resistance Temperature Detector (Dirençsel sıcaklık algılayıcı)
SNR	Signal to Noise Ratio (Sinyal gürültü oranı)
SPI	Serial Peripheral Interface (Seri çevresel arayüz)
YSA	Yapay Sinir Ağı

1. GİRİŞ

Günümüzde algılayıcılar, denetim, ölçüm, biyomedikal, otomasyon gibi pek çok endüstriyel alanda yaygın olarak kullanılmaktadır. Bir sistem olarak algılayıcılar, birinci veya ikinci dereceden modellenebilmektedir. Her algılayıcının statik ve dinamik tepkileri farklı olabilmektedir. Verilen tepkinin dışında, algılayıcılar, giriş-çıkış bilgisi açısından doğrusal veya doğrusal olmayan davranışlar gösterebilmektedir. Yaygın kullanılan algılayıcıların pek çoğunun ölçtükleri fiziksel büyüklük ile ürettikleri elektriksel çıktı arasında doğrusal olmayan bir ilişki bulunmaktadır [1].

Algılayıcıdan alınan akım veya gerilim bilgisini analog ya da sayısal sistemler kullanarak değerlendirmek ve bu sayede ölçülen fiziksel büyüklüğe ulaşmak için, algılayıcının giriş ve çıkışının matematiksel açıdan ilişkilendirilebilmesi, yani algılayıcının modellenebilmesi gerekmektedir. Ancak, bu modellemenin her zaman kolay olmadığı; modelleme yapılabilse dahi bu modelin donanım üzerinde gerçekleştirilmesinin kaynak ve zaman darlığına yol açabildiği ifade edilmektedir [2]. Bu sebeple algılayıcı giriş-çıkışı arasındaki ilişkilendirmeyi yapmak üzere pek çok yöntem geliştirilmiştir. Bu yöntemlere algılayıcı doğrusallaştırma yöntemleri adı verilmektedir. Özellikle sayısal sistemlerde algılayıcı doğrusallaştırma yöntemlerinin çeşitliliği artmaktadır [3]. Yaygın kullanılan doğrusallaştırma yöntemlerinden bazıları aşağıda verilmiştir;

- Ters fonksiyon alma
- Başvuru çizelgesi
- Kısmi parçalı doğrusallaştırma
- Programlanabilir kazançlı yükselteç (İng. Programmable Gain Amplifier, PGA) ile birlikte kısmi parçalı doğrusallaştırma
- Yapay sinir ağları (YSA)

Ters fonksiyon alma yöntemi analog sistemlerde uygulanan bir yöntemdir. Daha çok üstel karakteristik gösteren algılayıcıların davranışlarının opamp tabanlı logaritmik yükselteçler kullanılarak doğrusallaştırılması ilkesine dayanır. Yalnızca analog devre elemanlarıyla gerçekleştirilebilen, belleğe ihtiyaç duymayan ve oldukça

hızlı bir sistemdir. Ancak algılayıcının kullanılabilirdiği ölçüm aralığı ve ölçüm hassasiyeti kullanılan logaritmik yükseltecin algılayıcı karakteristiği ile uyumuyla sınırlanmaktadır. Ayrıca doğrusallaştırma devresinin farklı algılayıcı tiplerine uyarlanması güç olmakta, bu da yöntemin esnekliğini azaltmaktadır [4].

Başvuru Çizelgesi (BÇ) yöntemi, kullanılan en eski ve en hızlı doğrusallaştırma yöntemlerindedir [5,6]. Bu yöntemde, algılayıcının belirli noktalarda ürettiği değerlere karşılık gelen fiziksel büyüklükler bir çizelgede tutulur. Algılayıcıdan okunan değer ile çizelgedeki değerler sırayla karşılaştırılarak okunan değere karşılık gelen en uygun fiziksel büyüklük saptanır [7,8]. Bu yöntemin dezavantajı, her bir örnek nokta için tutulan giriş-çıkış kaydının bellekte yer kaplaması ve yöntemin toplam bellek ihtiyacının fazla olması; bunun yanı sıra artan ölçüm hassasiyetinin doğrudan bellek kullanımını arttırmasıdır [2].

Kısmi Parçalı Doğrusallaştırma (KPD) yöntemi, BÇ yönteminin geliştirilmiş halidir. Bu yöntemde de belirli noktalara ait giriş-çıkış verileri bir çizelge içinde saklanır. Ölçüm sırasında öncelikle, algılayıcıdan okunan verinin çizelgede tutulan hangi iki nokta arasına denk geldiği saptanır. Ardından bu iki nokta arasındaki doğru denklemi kullanılarak okunan ara değere karşılık gelen fiziksel büyüklük hesaplanır. Bu yöntemde tutulan kayıtların aralıklarının BÇ yöntemindeki kadar sık olması gerekmediğinden, ihtiyaç duyulan bellek miktarı azalmaktadır [9].

2004 sonrası teknolojik gelişmelerle birlikte, PGA entegre devreleri üretilmeye başlanmıştır. Bu entegrelerin en büyük avantajı, sağladıkları kazançların sayısal devrelerle ayarlanabilmesidir. 2005'de yapılan KPD tabanlı çalışmada [10] algılayıcı olarak NTC tipindeki termistör kullanılmış, algılayıcı gerilimi belirli bir eşik değerinin altına düştüğünde PGA yardımıyla yükseltmiş ve bu sayede özellikle yüksek sıcaklık ölçümlerindeki doğruluk önemli oranda arttırılmış; ancak bellek kullanımı kısmi parçalı doğrusallaştırma yönteminin doğası gereği azaltılamamıştır. PGA, sadece KPD yöntemiyle değil, BÇ yöntemiyle de kullanılabilir [2].

YSA'lar sinyal ve görüntü işleme, örüntü tanıma [11], denetim sistemleri [12] ve ölçme uygulamaları [13] gibi pek çok alanda yaygın olarak kullanılmaktadır. Yeterli katman ve nöron sayısına sahip bir YSA'nın doğrusal olmayan bir fonksiyonun

kestirimi için kullanılabilirdiği bilinmektedir [2]. Bu sebeple YSA'lar algılayıcıların doğrusal olmayan girdi-çıkı ilişkilerini çözümlenmekte kullanılmaktadır. YSA'lar analog devrelerle gerçekleştirilebildiği gibi [14], sayısal denetleyici tabanlı sistemlerle de gerçekleştirilebilmektedir [15,16]. Ancak yapılan çalışmalara [14,15,16] bakıldığında algılayıcının ölçüm aralığının yalnızca dar bir bölümünün kullanıldığı, alt ve üst sıcaklık limitlerine yakın bölgelerde ölçüm hatalarının özellikle BÇ yöntemine kıyasla oldukça yüksek olduğu, hatanın azaltılması için ise sayısal sistemlerde gerçekleştirilmesi zorlaşan daha fazla katman ve nöron sayısına sahip ağ yapılarının kullanımına gerek duyulduğu görülmektedir.

Bu çalışmada ise, algılayıcı devrenin ürettiği sinyal PGA yardımıyla belirli bir eşik seviyesi üzerinde tutulmuş, her bir kazanç bölgesine ait farklı YSA parametreleri kullanılarak YSA tabanlı kısmi doğrusallaştırma yapılmıştır. Bu hibrit yöntemle birlikte, doğrusallaştırma uygulamalarının başarı ölçütü olan ortalama karesel hata, çevrim hızı ve bellek kullanımı değerlerinin iyileştirilmesi amaçlanmıştır. Uygulanan PGA ve YSA'dan oluşan bu yöntem ile başvuru çizelgesi, PGA'lı başvuru çizelgesi ve yalnızca YSA yöntemleri hem yazılım hem de donanım ortamlarında uygulanmış, elde edilen sonuçlar ortalama karesel hata, hız ve bellek kullanımı yönlerinden karşılaştırılmıştır.

YSA'nın paralel çalışma prensibini donanım üzerinde gerçekleştirmek ve önerilen yöntemin hızını arttırabilmek için özelleşmiş sayısal devrelere ihtiyaç duyulmuştur. Günümüz teknolojisinde bu tür devrelerin gerçekleştirilebildiği en uygun platform Alan Programlanabilir Kapı Dizileri (FPGA)'dir. Bu sebeple yöntemin donanımsal uygulaması FPGA deney setinde gerçekleştirilmiştir.

Önerilen uygulamanın hızını etkileyen önemli unsurlardan bir diğeri ise PGA'nın kazancının değişim hızıdır. Piyasada bulunabilen hazır PGA entegreleri genellikle yavaş olan SPI (Serial Programming Interface), I²C (Inter-Integrated Circuit) gibi seri arayüzlerle kontrol edilebilmektedir. Önerilen yöntemin hızını arttırmak amacıyla PGA yapısı ayrık elemanlarla tasarlanmıştır. Tasarlanan bu yapı paralel arayüzle kontrol edilebilmekte ve bu sayede kazanç ayarlama işlemleri hazır entegrelere nazaran çok daha hızlı bir biçimde gerçekleştirilebilmektedir.

PGA ve YSA yöntemlerinden oluşan hibrit yöntemin başarısını değerlendirmek üzere doğrusal olmayan bir algılayıcının kullanımı gerekliliği doğmuştur. Bu amaçla endüstriyel uygulamalarda sıklıkla kullanılan NTC tipindeki sıcaklık algılayıcısı kullanılmıştır. Bu algılayıcının direnci sıcaklığa bağlı olarak değişmektedir. Algılayıcının direnci genellikle gerilime çevrilerek okunmaktadır. Uygulama kapsamında NTC sıcaklığının geniş bir aralıkta kontrollü olarak değiştirilemeyeceği ve sabitlenemeyeceği öngörülerek ölçüm için gerekli gerilim değerleri doğrusallaştırma sistemlerine Sayısaldan Analoga Çevirici (Digital to Analog Converter, DAC) entegre kullanılarak sağlanmıştır.

Bölüm 2'de algılayıcı doğrusallaştırma kavramına, analog ve sayısal sistemlerde yaygın kullanılan algılayıcı doğrusallaştırma yöntemlerine yer verilmiştir.

Bölüm 3'de algılayıcı doğrusallaştırma yöntemlerinden olan ve diğer pek çok alanda da yaygın kullanım alanı bulan YSA anlatılmıştır. Ayrıca bu bölümde sık kullanılan ağlardan olan Çok Katlı Perseptron (İng. Multi-Layer Perceptron, MLP) ve YSA'ların eğitiminde sıklıkla tercih edilen Geri Yayılım (İng. Back Propagation, BP) algoritmasının ayrıntıları anlatılmıştır.

Bölüm 4'de uygulamanın önemli bir bileşeni olan Programlanabilir Kazanç Yükseltici (PGA)'nin kullanım amacına, yapısına, tasarımında karşılaşılan belli başlı problemlere değinilmiştir.

Bölüm 5'de FPGA'lerin genel yapısı ve FPGA ile gerçekleştirilen tasarım aşamaları anlatılmıştır.

Bölüm 6'da, uygulamanın merkezinde yer alan sıcaklık algılayıcısı NTC'nin genel özelliklerine ve kullanım alanlarına yer verilmiştir.

Bölüm 7'de YSA ve KPD tabanlı algılayıcı doğrusallaştırma uygulamalarının benzetimleri hem PGA kullanılmadan hem de PGA kullanılarak yapılmış ve sonuçları paylaşılmıştır. Benzetimi yapılan bu yöntemlerin donanımsal uygulamaları FPGA deney setinde gerçekleştirilmiş ve sonuçlara yer verilmiştir. Elde edilen benzetim ve deney sonuçları kullanılarak dört yöntem hata, hız ve bellek kullanımı yönünden karşılaştırılmıştır.

Bölüm 8’de ise önerilen yöntem olan PGA+YSA yönteminin performansı YSA, KPD, PGA+KPD yöntemlerinin performanslarıyla karşılaştırılmıştır. Yorum ve öneriler sunulmuştur.

2. ALGILAYICI DOĞRUSALLAŞTIRMA

Günümüzde, pek çok alanda geniş bir kullanım yelpazesine sahip olan algılayıcıların birçoğunun ölçtükleri fiziksel büyüklük ile ürettikleri elektriksel form arasında doğrusal olmayan bir ilişki bulunur. Geleneksel yöntemler kullanıldığında, bir algılayıcıdan alınan sinyal kullanılarak ölçülen büyüklüğü hesaplayabilmek için algılayıcının giriş-çıkış modelinin matematiksel olarak bilinmesi gerekir. Ancak bu matematiksel modellemenin gerçekleştirilmesinin zor olduğu; gerçekleştirildiğinde ise kaynak ve zaman darlığına neden olduğu ifade edilmektedir [2]. Bu tip doğrusal olmayan algılayıcıların kullanımını kolaylaştırmak amacıyla algılayıcı doğrusallaştırma adı verilen yöntemler kullanılmaktadır.

Doğrusal olmayan bir algılayıcının doğrusal çalışma bölgesini genişletmek üzere, algılayıcı çıkışına Şekil 2.1'de gösterilen bir doğrusallaştırma bloğu yerleştirilir. Böylece arzu edilen doğrusal karakteristik elde edilir. Doğrusallaştırma işleminin analog devrelerle doğrudan sinyal koşullandırma şeklinde uygulanabileceği gibi, sayısal devrelerle de yazılımsal bir takım algoritmalar kullanılarak gerçekleştirilebileceği ifade edilmektedir [17].



Şekil 2.1 Algılayıcı doğrusallaştırma akış diyagramı

Analog ve sayısal sistemlerde yaygın kullanılan doğrusallaştırma yöntemleri aşağıda verilmiştir.

- Analog devrelerle gerçekleştirilebilen yöntemler
 - Ters fonksiyon alma (opamp tabanlı logaritmik yükselteç)
- Sayısal devrelerle gerçekleştirilebilen yöntemler
 - Başvuru çizelgesi (BÇ)
 - PGA kullanımı ile BÇ
 - Kısmi parçalı doğrusallaştırma
- Hem analog hem de sayısal devrelerle gerçekleştirilebilen yöntemler

- Yapay sinir ağıları (YSA)
- PGA kullanımı ile YSA

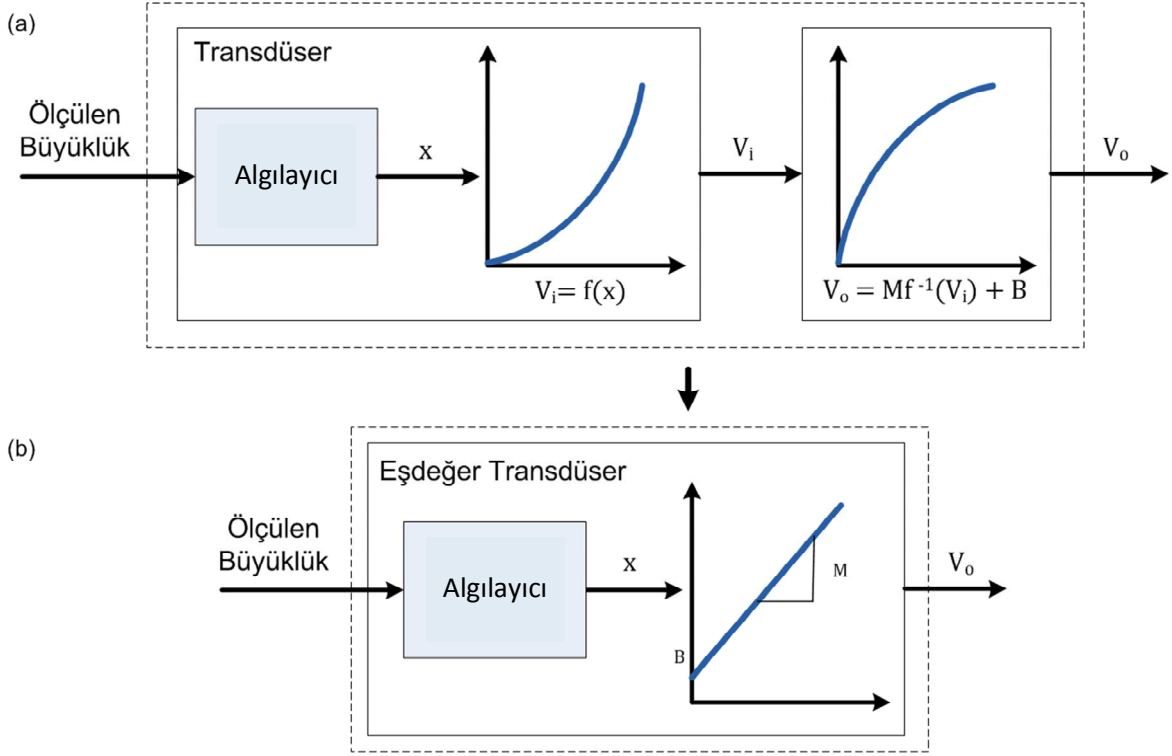
2.1. Ters Fonksiyon Alma Yöntemi

Analog sistemlerde algılayıcı doğrusallaştırmada yaygın kullanılan tekniklerden birinin, transdüser (algılayıcı ve algılayıcı verisini elektriksel sinyale çeviren devre bütünü) transfer fonksiyonunun tersinin analog devrelerle gerçekleştirilip kullanılması olduğu bilinmektedir. Bu yöntemde, öncelikle uygulamaya özgü kabul edilebilir hata payı belirlenir. Transdüserin transfer fonksiyonu, belirlenen bu hata payı göz önünde bulundurularak çıkarılır. Ardından bu fonksiyonun tersini gerçekleyen bir analog devre tasarlanır. Yöntemin daha sonraki aşamasında ise, transdüserin çıkışı, tasarlanan devreye girdi olarak aktarılır. Bu yolla, devrenin çıkışı ile ölçülen büyüklük arasında doğrusal bir ilişki yakalanmış olur. Örneğin, fonksiyonu logaritmik olan bir transdüserin doğrusallaştırılması için gereken ters fonksiyonun üstel karakterde olması gerektiği bilinmektedir. Ters fonksiyon kullanımı yöntemini anlatan grafiksel ifade Şekil 2.2 (a)'da yer almaktadır.

Şekil 2.2 (a)'da görüldüğü gibi, algılayıcıdan alınan x verisi transdüserin elektriksel sinyal üretici devresine girdi oluşturmakta ve transdüser çıkışı V_i elde edilmektedir. V_i sinyali tasarlanan analog devreye aktarılmaktadır ve bu devrenin çıkışında denklem (2-1)'de verilen transfer fonksiyonuna uygun olarak V_o üretilmektedir [4].

$$V_o = Mf^{-1}(V_i) + B \quad (2-1)$$

Denklem (2-1)'de yer alan “ M ” doğrunun eğimini gösteren katsayıdır. Bu katsayı genellikle 1 olarak seçilmektedir. Denklemde yer alan “ B ” sabiti ise sistem çıkışının minimum değerini ifade etmektedir. Ölçülen büyüklük ile sistemin çıkışı arasındaki ilişki Şekil 2.2 (b)'de verilmiştir.



Şekil 2.2 Ters fonksiyon yönteminin grafiksel ifadesi (Kraska, [4] 'dan değiştirilerek)

Tersini alma yönteminin, anlaşılma kolaylığı ve uygulama adımlarının netliği itibariyle kullanışlı görülebildiği ancak bu yöntemin bir takım problemleri de beraberinde getirdiği belirtilmektedir [[17,4],[18]. Örneğin tasarlanan devrenin sıcaklık ve gürültüye karşı dayanıklı olması gerekmekte, bu nedenle de bir seri önlemlerin alınması gerektiği bilinmektedir. Bu durum ise tasarımın karmaşıklığını arttırmaktadır. Ayrıca tasarlanan devre algılayıcıya özgü olduğundan esnekliği ve kullanım alanı kısıtlıdır. Bu da her yeni algılayıcıyla gerçekleştirilen uygulamada yeni bir tasarım demektir. Bu durum tasarımcı için zaman alıcı ve maliyetli olabilmektedir [[17,4],[18].

Ters fonksiyon alma yönteminde sistemin transfer fonksiyonunun tersinin gerçekleşmesinin her zaman kolay olmadığı bilinmektedir. Bu nedenle, çözünürlük ya da belirli bölgelerdeki ölçümlerin sınırlandırılması gerekebilmektedir. Örneğin bir algılayıcının yalnızca belirli bir bölgesi için fonksiyon gerçekleştirilmesi yapılır ve sadece bu bölge için hedeflenen hata değerinde ölçümler alınabilir; ya da hata payı yükseltilerek daha geniş bir bölgede, ancak yükseltilmiş hata payı toleransı ile ölçüm gerçekleştirmek mümkün olabilir. Hassas ölçüm gerektiren uygulamalarda bu

yöntemin belirtilen kısıtlarından ötürü tercih edilmeyebildiği ifade edilmektedir [[17,4].

Ters fonksiyon alma yönteminin kullanımı amaca yönelik olarak farklılık göstermektedir. Hassas çalışmalarda bu yöntem tercih edilmezken, sayısal yöntemlerle gerçekleştiren uygulamalara nazaran daha hızlı olduğundan, hızın ön planda olduğu çalışmalarda bu yöntem tercih edilmektedir [[17,4].

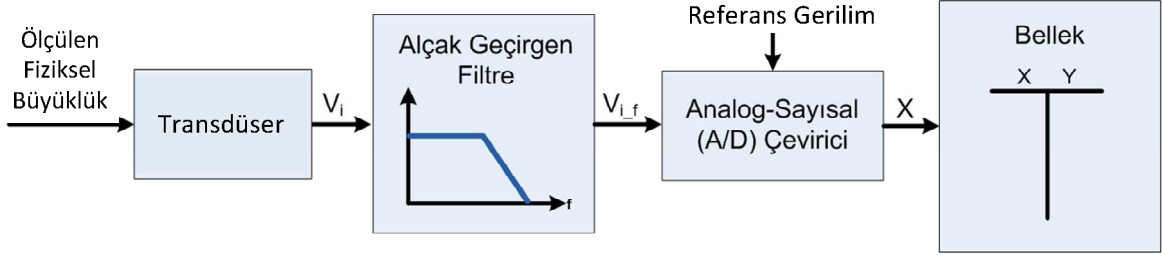
2.2. Başvuru Çizelgesi Yöntemi

Doğrusallaştırma işleminin sayısal devrelerle de gerçekleştirilebildiği ve bunun için yaygın olarak kullanılan sayısal yöntemlerden birinin de başvuru çizelgesi yöntemi (İng. look-up table) olduğu bilinmektedir [[2].

Bu yöntemde; transdüserden alınan analog sinyal öncelikle alçak geçiren filtreden geçirilmektedir (Şekil 2.3). Bu işlem ile, sinyal üzerine binen ve ölçümde ani değişimlere yol açabilecek gürültüler filtrelenmiş olur. Ardından filtrelenmiş sinyal, analogtan sayısala çevirici (İng. Analog to Digital Converter, ADC) devre vasıtasıyla sayısal veriye dönüştürülmektedir [4].

ADC'ler belirli bir çözünürlükte sayısal çıkış üretmektedirler. Örneğin, 8 bitlik bir ADC'nin, 0 ile 256 aralığında sayısal değerler ürettiği bilinmektedir. Burada 0, referans voltajının eksi kutbunu, 256 referans voltajının artı kutbunu, arada kalan değerler ise tam aralıkla doğrusal orantılı bir çıkışı ifade etmektedir. Örneğin, 2.5V'luk bir referans gerilimiyle çalışan ADC'nin 0 değerini üretmesi girişin 0V, 256 değerini üretmesi girişin 2.5V, 128 değerini üretmesi için ise girişin 1.25V olduğunu göstermektedir.

Kontrolcü belleğinde ADC'den alınabilecek her bir sayısal değere karşılık gelen ölçüm büyüklüğünün çizelgesi tutulmaktadır. Örneğin devrede 8 bitlik bir ADC kullanılıyorsa toplamda $2^8 = 256$ satırdan oluşan bir çizelge hazırlanmış olmalıdır. Alınan sayısal veri çizelgedeki değerlerle tek tek karşılaştırılmaktadır. Çizelgede ilgili satır bulunduğu girişin hangi çıkışa karşılık geldiği belirlenmiş olur. BÇ yönteminin işlem akış şeması Şekil 2.4'de verilmektedir.



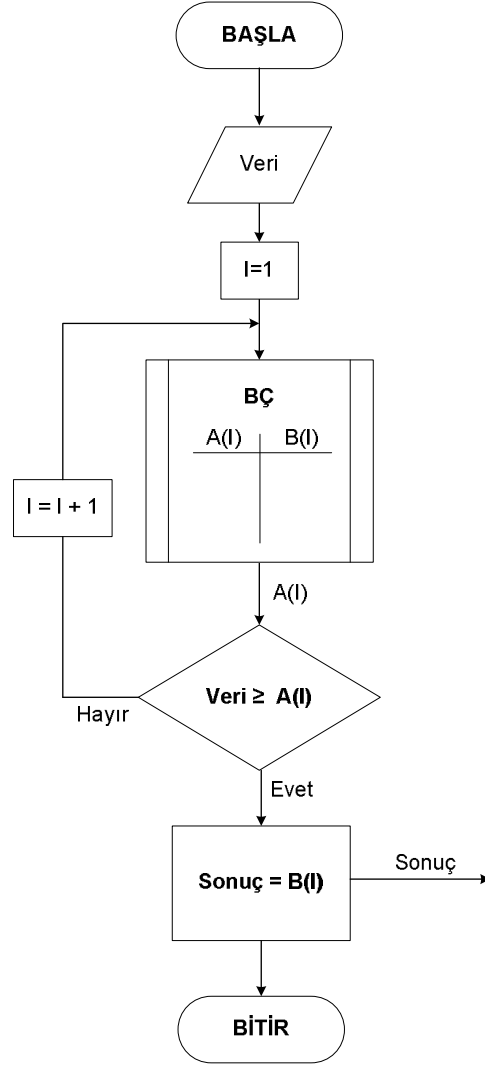
Şekil 2.3 Başvuru çizelgesi yönteminin akış şeması (Kraska, [4] 'dan değiştirilerek)

Başvuru çizelgesi yönteminde çözünürlük arttırılmak isteniyorsa çizelgenin boyutunu da arttırmak gerekmektedir. Örneğin 16 bitlik bir ADC kullanıldığında hazırlanması gereken çizelge $2^{16} \times 2$ 'lik bir matris halini almaktadır. Burada her bir sayısal girdi verisi 16 bitten (2 byte) oluşmaktadır. Çıkış verisinin de 16 bitten oluşan gerçek değerler olduğu düşünülürse toplamda $(2^{16} \times 2 \times 16)$ bit ~ 262 Kbyte'lık bir belleğe ihtiyaç duyulmaktadır.

Çizelge yönteminde çözünürlüğü arttırmanın bir dezavantajı da artan çözünürlükle birlikte kıyaslama zamanının artmasıdır. Bu yöntemde girişin hangi çıkışı ifade ettiği çizelgedeki değerlere tek tek bakılarak bulunmaya çalışılır. Artan çizelge boyutu da karşılaştırma işleminin daha uzun zaman alması anlamına gelmektedir. Bu da uygulamanın hızını düşüren bir faktördür. İstenilen hızı sağlamak için daha yüksek hızda çalışan kontrolcüler kullanmak gerekir ki bu da uygulamanın maliyetinin artmasına yol açmaktadır.

Çizelge yönteminin kullanıldığı uygulamalarda genellikle mikrodenetleyiciler kullanılmaktadır. Akışta verilen ADC ve bellek yapıları mikrodenetleyicilerde tümleşik halde olabilmektedir. Ancak mikrodenetleyicilerin belleği kısıtlıdır. Bu yüzden yüksek çözünürlük istendiğinde daha geniş kapasiteli mikrodenetleyici kullanımı ihtiyacı doğmakta ve bu da yüksek hacimli üretimlerde maliyetin önemli ölçüde artmasına yol açmaktadır.

Çizelge yönteminin en önemli avantajı hızlı çalışabilmesi ve uygulamasının kolay olmasıdır.

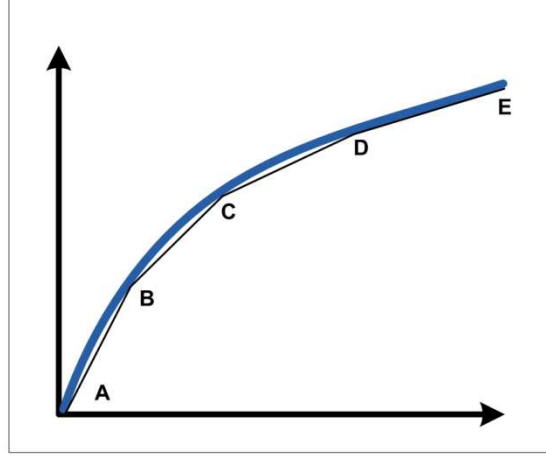


Şekil 2.4 BÇ yönteminin işlem akış şeması

2.3. Kısmi Parçalı Doğrusallaştırma (KPD) Yöntemi

Sayısal sistemlerde yaygın kullanılan bir diğer doğrusallaştırma yöntemi KPD yöntemidir. KPD yöntemi, BÇ yönteminin geliştirilmiş halidir. Bu yöntemde transdüser çıkışı grafiğinde doğrusala yakın aralıklar belirlenmekte ve bu aralıkların her birinin doğrusal olduğu varsayılmaktadır. Belirlenen bu aralıkların başlangıç ve bitiş noktalarına ait giriş-çıkış verileri bir çizelge içinde saklanır. Ölçüm sırasında öncelikle, algılayıcıdan okunan verinin çizelgede tutulan hangi iki nokta arasında denk geldiği saptanır. Ardından bu iki nokta arasındaki doğru denklemi kullanılarak okunan ara değere karşılık gelen fiziksel büyüklük hesaplanır. Bu yöntemde tutulan kayıtların aralıklarının BÇ yöntemindeki kadar sık olması gerekmediğinden, ihtiyaç duyulan bellek miktarı azalmaktadır [9].

Örneğin Şekil 2.5'de verilen transdüser grafiği AB, BC, CD ve DE olmak üzere dört doğrusal bölgeye ayrılmıştır [[2,4],[17],[19].

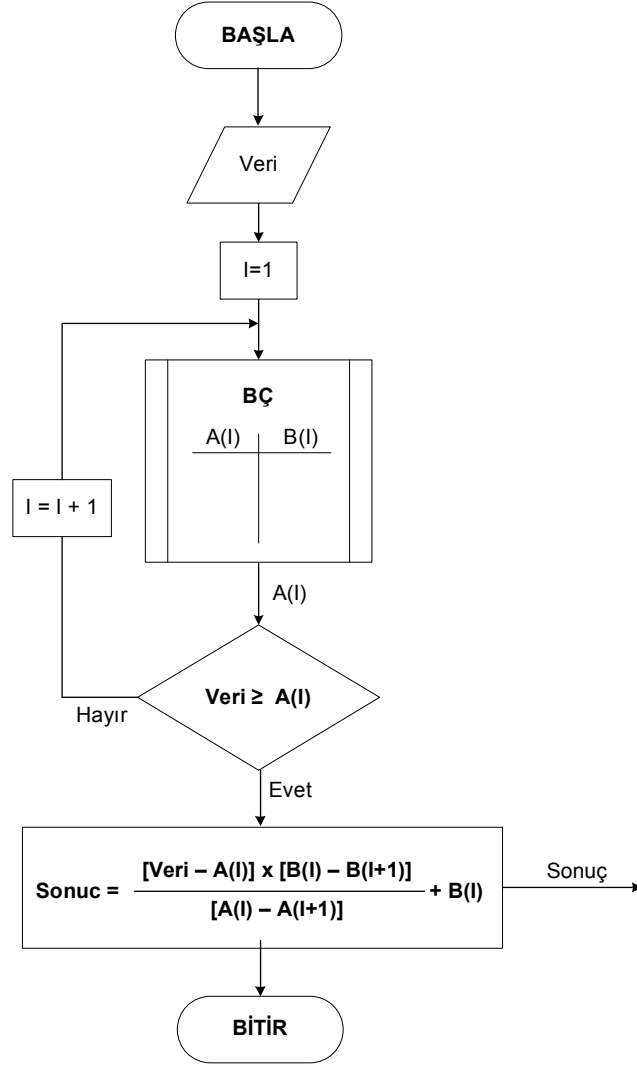


Şekil 2.5 Transdüser grafiğinde belirlenen kısmi doğrusallaştırılmış parçalar

Bu yöntemin, transdüserin transfer fonksiyonunun matematiksel olarak modellenemediği ya da matematiksel modelin fazla karmaşık olduğu durumlarda kullanışlı olduğu bilinmektedir. Kısmi parçalı doğrusallaştırma yönteminde, her bir aralığın kazancı birbirinden bağımsız olarak belirlenebilmektedir.

Algılayıcı karakteristiği zamanla değişen, ya da karakteristiğinde farklılıklar görülebilen aynı tipte algılayıcıların kullanıldığı uygulamalarda kısmi doğrusallaştırılan aralıkların, ihtiyaç duyuldukça yeniden belirlenmesi gerekmektedir [4].

KPD yönteminin işlem akış şeması Şekil 2.6'da verilmiştir.



Şekil 2.6 KPD yönteminin işlem akış şeması

2.4. Yapay Sinir Ağlarının Kullanılması Yöntemi

YSA'lar sinyal ve görüntü işleme, örüntü tanıma [11], denetim sistemleri [12] ve ölçme uygulamaları [13] gibi pek çok alanda yaygın olarak kullanılmaktadır. Yeterli katman ve nöron sayısına sahip bir YSA'nın doğrusal olmayan bir fonksiyonun kestirimi için kullanılabilirdiği bilinmektedir [2]. Bu sebeple YSA'lar algılayıcıların doğrusal olmayan girdi-çıkı ilişkilerini çözümlenmekte kullanılmaktadır. YSA'lar analog devrelerle gerçekleştirilebildiği gibi [14], sayısal denetleyici tabanlı sistemlerle de gerçekleştirilebilmektedir [15,16]. YSA'lar ile ilgili ayrıntılı bilgi 3. Bölümde verilmiştir.

2.5. Hibrit Yöntemler

Algılayıcı çıkışının geniş bir aralıkta doğrusal olmayan dağılımı sonucu, bazı ölçüm aralıkları dar bir çıktı kümesi içine sıkışmaktadır. Bu durum, özellikle ADC çözünürlüğünün yetersiz, gürültü oranının yüksek olduğu uygulamalarda düşük gerilim aralığındaki ölçümlerde hata oranını yükseltmektedir.

2004 sonrası teknolojik gelişmelerle birlikte, PGA entegre devreleri üretilmeye başlanmıştır. Bu entegrelerin en büyük avantajı, sağladıkları kazançların sayısal devrelerle ayarlanabilmesidir. PGA'nın sağladığı ayarlanabilir kazanç kabiliyeti sayesinde algılayıcıdan gelen sinyaller istenilen aralıklarda yükseltilmekte ve ADC ile daha sağlıklı okumalar yapılabilmektedir [10]. Okunan sinyalin iyileştirilmesinin ardından önceki bölümlerde anlatılan analog ya da sayısal doğrusallaştırma yöntemleri kullanılabilir. PGA, sıklıkla başvuru çizelgesi ve kısmi parçalı doğrusallaştırma yöntemleriyle kullanılmaktadır [10].

Bu tez kapsamında yapılan çalışmada uygulanan PGA'nın YSA ile kullanılmasıyla birlikte oluşan yöntem de hibrit yöntemler arasında yerini almaktadır.

3. YAPAY SINİR AĞLARI

3.1. Giriş

YSA'ların biyolojik sinir sistemlerinden, özellikle insan beyninden esinlenen bir yapay zeka uygulaması olduğu bilinmektedir. YSA'lar; birbirine bağlı çok sayıda bilgi işleme elemanını (nöronları) bir bütünlük içinde çalıştırarak özel bir probleme çözüm üretmeyi amaçlamaktadır. YSA'lar eldeki örnek veriler (girdilere karşı çıktılar) ile eğitilebilmekte, eğitim sırasında karşılaşmadıkları bir girdiyle karşılaştıklarında ise düşük hatayla sonuç üretebilmektedirler. Bu da YSA'ların genelleme özelliğiyle mümkün olmaktadır. YSA'ların bu esnek ve güçlü özellikleri yapay zeka konusunda çalışan araştırmacıların yoğun ilgisini çekmektedir. Her ne kadar günümüzde insan beyni gibi karmaşık bir yapının modellenmesi henüz YSA'lar ile mümkün olmasa da, bu ağların beynin kimyasal işlem hızına karşı sunduğu elektriksel işlem hızı YSA'ları özelleşmiş uygulamalarda hesaplama ve sonuç üretmede biyolojik sistemlerden daha üstün kılmaktadır [[20],[21].

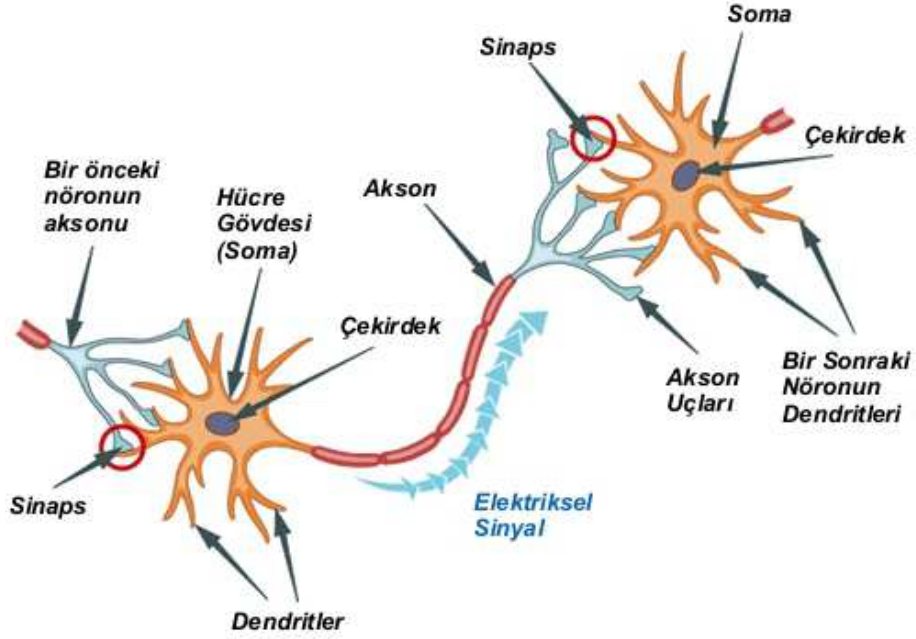
3.2. Biyolojik Nöron

Sinir hücreleri (nöronlar) sinir sisteminin temel işlem elemanlarıdır. Nöronların çoğu üç bölümden oluşmaktadır; çevreden gelen uyarıları alan, çok sayıda ve kısa olan özelleşmiş uzantılar (dendrit), hücreye gelen uyarıları başka bir hücreye taşıyan, tek ve uzun olan yapılar (akson-eksen) ve hücre gövdesidir (perikaryon-soma; peri-çevre, karyon-nukleus). Akson, perikaryondan uzak bölgelerde genellikle dallanmakta ve uçlarda düğme şeklinde genişlemiş bölgeleri oluşturmaktadır. Bu bölgeler, diğer sinir hücreleri ya da sinir hücresi olmayan başka hücrelerle etkileştikleri, bilgiyi aktardıkları sinaps bölgeleridir [[22].

Sinaps bölgeleri iletişim kanalı olarak vazife görmekte ve bir nöronun diğer bir nörona sinyal taşıma görevini üstlenmektedirler. Sinyal taşıma ise elektrik yüklü iyonlar vasıtasıyla olmaktadır. Kararlı bir nöronun hücre zarında görülen gerilim -85mV seviyelerindedir. Dendritlerden yeterli miktarda sinyal alınması ve hücre zarında görülen gerilimin -40mV'luk eşik değerini aşması durumunda hücrenin kararlılık hali bozulmakta ve hücre zarını depolarize edecek yönde stoplazmaya

akım aşılanmaktadır. Böylece hücre aktif moda geçmekte ve akson üzerinden diğer nöronlara iletilmek üzere bir elektriksel darbe üretilmektedir [[23].

Sinaps bölgelerine ulaşan sinyallerin karşı tarafa taşınması için sinaptik eşik değerinin aşılması gerekmektedir. Bu nedenle sinapslerin eşik değerleri bir nöronun diğer bir nöron üzerindeki etkisinde etken rol oynamaktadır. Öğrenme süreci de sinapslerin eşik değerlerinin değiştirilmesiyle gerçekleşmektedir [[21].



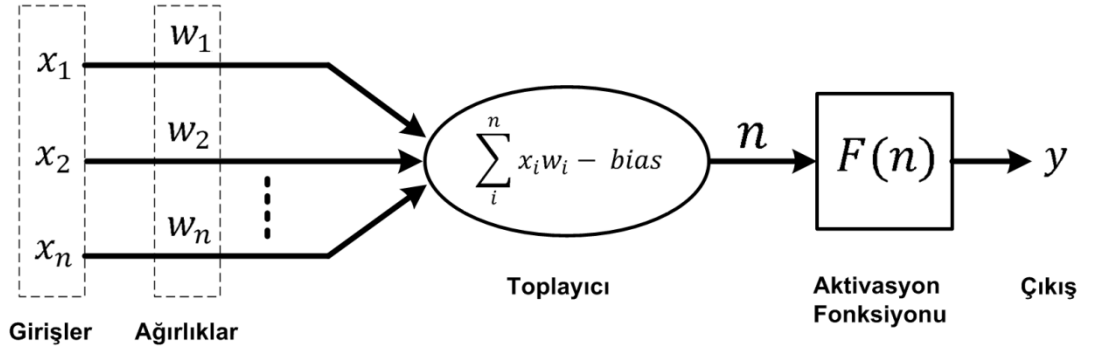
Şekil 3.1 Sinir Hücresi (Nöron) [[24]

İnsan beyninde yaklaşık 10^{10} adet nöron ve nöronları birbirine bağlayan 6×10^{13} 'ten fazla ara bağlantı olduğu ifade edilmektedir. Sayısal elektronik devrelerde saniyede milyonlarca işlem yapılabilirken biyolojik bir nöronun tepki süresi yaklaşık 1-2 ms'dir. Bu da biyolojik nöronların nispeten yavaş çalıştığı kanısını doğurur. Ancak beynin karmaşık, doğrusal olmayan ve paralel dağılmış yapısı öğrenme, birleştirme, uyarılma ve genelleme yeteneklerinin üst düzeyde performans göstermesini sağlamaktadır [[25].

3.3. Yapay Sinir Ağı

3.3.1. Yapay nöron modeli

Yapay nöron modeli beş kısımdan oluşmaktadır; girişler, ağırlıklar, toplama fonksiyonu, aktivasyon fonksiyonu ve çıkış. Yapay nöron modeli Şekil 3.2'de gösterilmiştir.



Şekil 3.2 Yapay Nöron Modeli

Yapay nöronlar biyolojik nöronların matematiksel benzetimleri olarak tanımlanmaktadır. Biyolojik sinir ağları ile YSA'lar karşılaştırıldığında biyolojik nöronlar ile yapay nöronlar, sinyal toplayıcı olarak görev yapan dendrit ile toplama fonksiyonu, aktivasyonun gerçekleştiği hücre gövdesi ile transfer fonksiyonu, hücrenin çıkış sinyalinin taşındığı aksonlar ile yapay nöron çıkışı, sinaps boşluklarının eşik değerleri ile yapay nöron giriş ağırlıkları eşleştirilebilir [26].

Çizelge 3-1 Biyolojik ve Yapay Sinir Hücreleri arasındaki analogi

Biyolojik Sinir Hücresi	Yapay Sinir Hücresi
Soma	Nöron
Dendrit	Toplama Fonksiyonu
Hücre Gövdesi	Transfer Fonksiyonu
Aksonlar	Nöron Çıkışı
Sinapslar	Ağırlıklar

Şekil 3.2'de gösterilen nöron birden fazla girişe ve tek çıkışa sahiptir. Her bir giriş bağlandığı hattın ağırlığıyla (W_i) çarpılmaktadır. Bu ağırlıklar, girişin nöronu ne

kadar etkileyeceğini belirlemektedirler. Tüm çarpım sonuçları nöronda toplanmakta, nöronun eşik değeri toplamdan çıkartıldıktan sonra elde edilen sonuç aktivasyon fonksiyonuna sokulmaktadır. Aktivasyon fonksiyonunun sonucu nöronun çıktısını oluşturmaktadır.

Ağırlıklar her bir girişin bağlı olduğu hatta özgü matematiksel büyüklüklerdir. Girişler nörona ulaşmadan önce bu ağırlıklar ile çarpılırlar. Dolayısıyla girişin nöronu ne derece etkileyeceği bu ağırlıklar ile belirlenmektedir. YSA'larda öğrenme bu ağırlıkların değiştirilmesiyle gerçekleşmektedir. Ağırlık değeri pozitif ya da negatif olabilir [27].

Tüm girişler ilgili ağırlıklarla çarpıldıktan sonra toplama fonksiyonuna girerek toplanırlar. Bu sayede nörona uygulanan net giriş büyüklüğü hesaplanmış olur.

Eşik değeri (bias) her bir nörona özgü net girdiyi etkileyen değerdir. Bu değer de pozitif ya da negatif olabilir. Eşik değeri toplama fonksiyonunu sonucundan çıkarılır ve aktivasyon fonksiyonuna girdi olacak değer hesaplanmış olur.

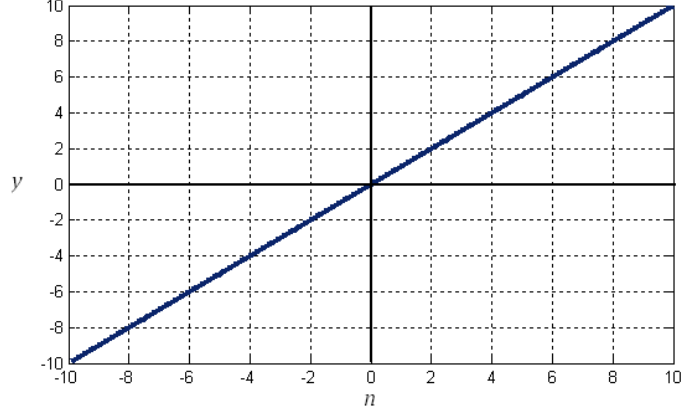
Transfer fonksiyonu, sıkıştırma işlemci veya eşik fonksiyonu olarak da bilinen aktivasyon fonksiyonu, girdi-çıkı ilişkisini modelleyen fonksiyondur. Aktivasyon fonksiyonları türevlenebilir olmalıdır. Sıkça kullanılan aktivasyon fonksiyonları aşağıda açıklanmaya çalışılmıştır [27].

3.3.1.1. Doğrusal fonksiyon

Doğrusal fonksiyonun matematiksel ifadesi (3-1)'de verilmiştir. Toplayıcı çıkışı (n), belirli bir katsayıyla (α) çarpılarak aktivasyon fonksiyonunun çıkışı (y) hesaplanır.

$$y = \alpha n \quad (3-1)$$

$\alpha = 1$ için doğrusal fonksiyonun grafiği Şekil 3.3'de verilmiştir.



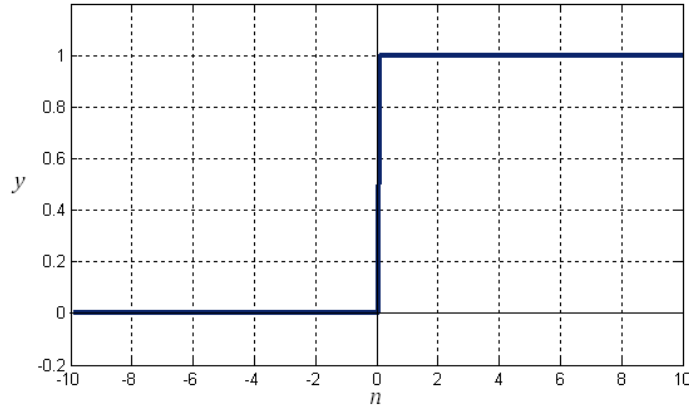
Şekil 3.3 $\alpha = 1$ için doğrusal fonksiyonun grafiği

3.3.1.2. Eşik değer fonksiyonu

Eşik değer (adım) fonksiyonunun matematiksel ifadesi (3-2)'de verilmiştir. Giriş eşik değerinden (β) küçükse sonuç 0; giriş eşik değerine eşit ya da eşik değerinden büyükse sonuç 1 olur.

$$y = \begin{cases} 0, & n < \beta \\ 1, & n \geq \beta \end{cases} \quad (3-2)$$

$\beta = 0$ için eşik değer fonksiyonu grafiği Şekil 3.4'de verilmiştir.



Şekil 3.4 $\beta=0$ için eşik değer fonksiyonu

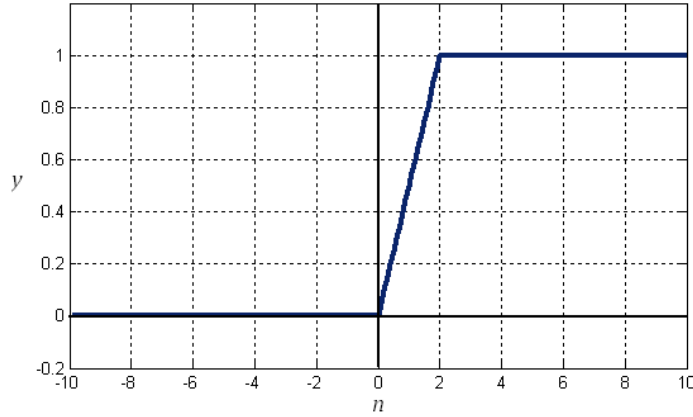
3.3.1.3. Kısmi doğrusallaştırılmış eşik değer fonksiyonu

Kısmi doğrusallaştırılmış eşik değer fonksiyonunun matematiksel ifadesi (3-3)'de yer almaktadır. Çıkış üç aralıkta değer alabilir. Giriş alt eşik değerinden (α) küçükse sonuç 0, üst eşik değerinden (β) büyükse sonuç 1, iki eşik değeri

arasındaysa sonuç bu eşik değerleri arasında tanımlanan doğrusal fonksiyonun sonucudur.

$$y = \begin{cases} 0, n < \alpha \\ n, \alpha \leq n \leq \beta \\ 1, n > \beta \end{cases} \quad (3-3)$$

$\alpha = 0$ ve $\beta=1$ için kısmi doğrusallaştırılmış eşik değer fonksiyonunun grafiği Şekil 3.5'de verilmiştir.



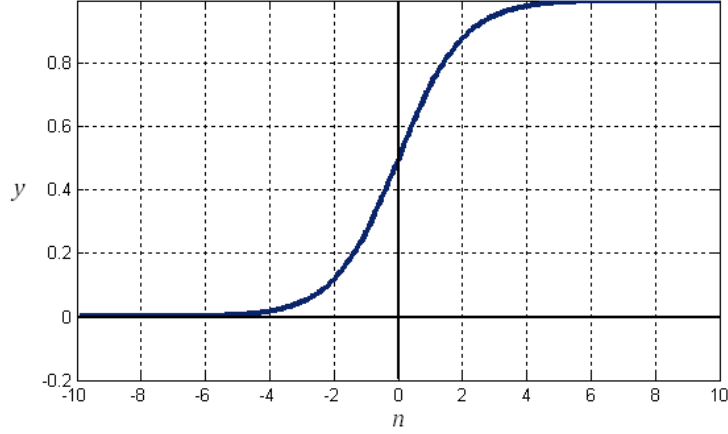
Şekil 3.5 $\alpha = 0$ ve $\beta=1$ için kısmi doğrusallaştırılmış eşik değer fonksiyonu

3.3.1.4. Lojistik sigmoid fonksiyonu

Lojistik sigmoid fonksiyonunun matematiksel ifadesi (3-4)'de verilmiştir. YSA'larda en sık kullanılan aktivasyon fonksiyonudur. Çıkışı 0 ile 1 arasında sınırlandırılmış, türevlenebilen ve sürekli artan bir fonksiyondur. Bu fonksiyon genellikle doğrusal olmayan ilişkilerin çözülmesinde kullanılmaktadır.

$$y = \frac{1}{1 + e^{-n}} \quad (3-4)$$

Lojistik sigmoid fonksiyonunun grafiği Şekil 3.6'da verilmiştir.



Şekil 3.6 Lojistik sigmoid fonksiyonunu

Sigmoid fonksiyonun YSA'larda yaygın kullanılmasının bir sebebi de türevinin (3-5) kolay hesaplanabilmesidir.

$$y' = y \times (1 - y) \quad (3-5)$$

3.3.1.5. Tanjant sigmoid fonksiyonu

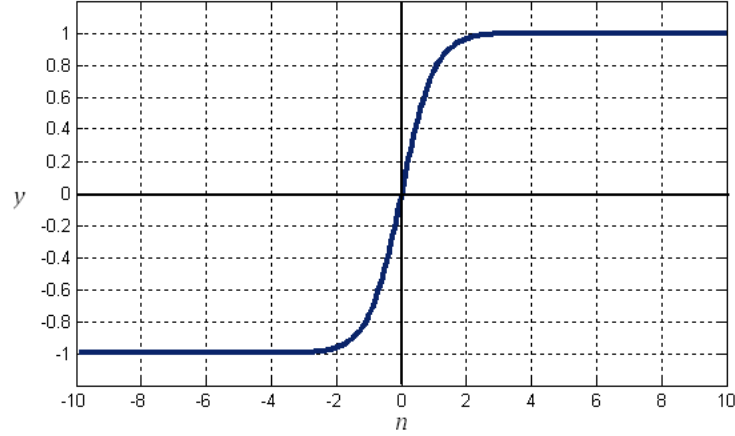
Tanjant sigmoid fonksiyonunun matematiksel ifadesi (3-6)'da verilmiştir. Sigmoid (S Şekilli) ailesinden çıkışı -1 ile +1 arasında sınırlanmış, türevlenebilen ve yine YSA'larda sıklıkla kullanılan bir fonksiyondur.

$$y = \frac{e^n - e^{-n}}{e^n + e^{-n}} \quad (3-6)$$

Tanjant sigmoid fonksiyonun YSA'larda yaygın kullanılmasının bir sebebi de lojistik sigmoid fonksiyonunda olduğu gibi türevinin kolay hesaplanabilmesidir (bkz.(3-7)).

$$y' = 1 - y^2 \quad (3-7)$$

Tanjant sigmoid fonksiyonunun grafiği Şekil 3.7'de verilmiştir.



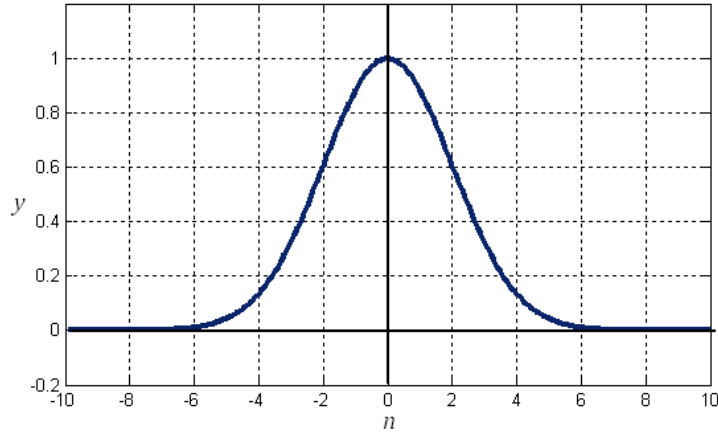
Şekil 3.7 Tanjant sigmoid fonksiyonu

3.3.1.6. Gauss fonksiyonu

Gauss fonksiyonunun matematiksel ifadesi (3-8)'de verilmiştir. YSA'larda özellikle doğrusal olmayan ilişkilendirmelerde kullanılan bir fonksiyondur.

$$y = e^{-r^2} \quad (3-8)$$

Gauss fonksiyonunun grafiği Şekil 3.8'de verilmiştir.

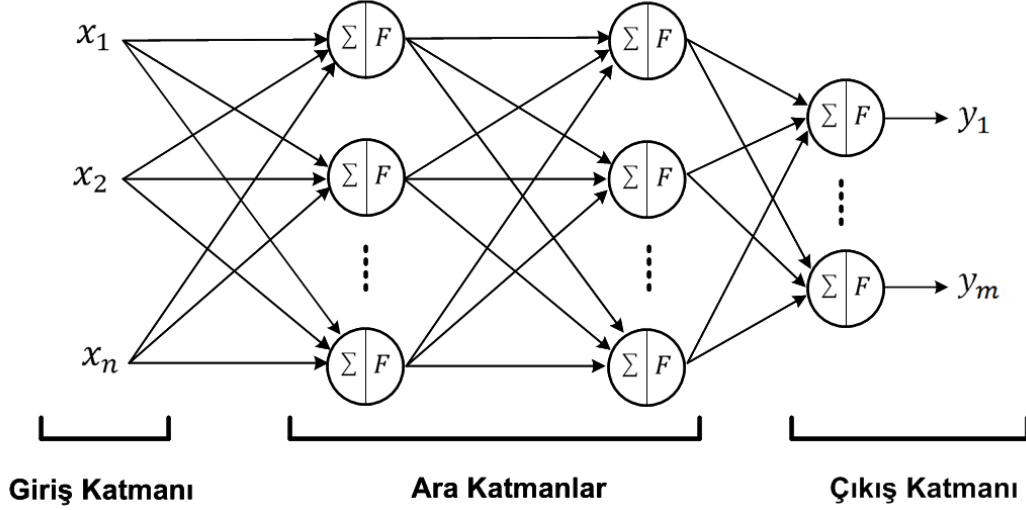


Şekil 3.8 Gauss fonksiyonu

3.3.2. Yapay sinir ağlarının yapısı

Bir yapay sinir ağı giriş katmanı, ara katman/katmanlar ve çıkış katmanı olmak üzere üç kısımdan oluşmaktadır. Ara katmanlar gizli katmanlar olarak da adlandırılmaktadırlar. Giriş katmanı adından da anlaşılacağı gibi; girişlerin ilk

gizli katmandaki nöronlara dağıldığı katmandır. Ara katman ara işlem nöronlarının bulunduğu katmandır. Bu katmanın sayısı bir ya da birden fazla olabilir. Çıkış katmanı ise YSA'nın çıkış nöronlarının bulunduğu ve YSA sonuçlarının alındığı katmandır. n adet girişi, m adet çıkışı ve iki adet ara katmanı bulunan bir YSA'nın ağ modeli örneği Şekil 3.9'da verilmiştir.



Şekil 3.9 İki ara katmanlı bir YSA örneği

3.3.2.1. YSA'ların genel özellikleri

YSA'ların özellikleri ağa özgü seçilen parametrelere göre farklılık göstermekle birlikte tüm modeller için genel özellikler şöyle sıralanabilir [25]:

- YSA'lar makine öğrenmesi gerçekleştirilir.
- YSA'lar bilgiyi saklarlar. YSA'larda saklanan bilgi nöronlar ve nöronlar arası bağlantılarda bulunur ve dağınık haldedir.
- YSA'lar örnekler ile eğitilebilirler.
- YSA'nın güvenilirliğinin sağlanabilmesi için önce eğitilmeleri; eğitilmiş sistemin eldeki test verisiyle test edilmesi; test sonuçlarıyla beklenen değerlerin farkının belirli bir ortalama hatanın altında kalması gerekir.
- YSA'lar eğitim sırasında karşılaşmadıkları bir girdiyle karşılaştıklarında düşük hatayla sonuç üretebilirler. Bu da YSA'ların genelleme özelliğiyle mümkün olmaktadır.

- YSA'lar algılamaya yönelik olaylarda; şekil ilişkilendirme, sınıflandırma, örüntü tamamlama gibi uygulamalarda kullanılabilirler.
- YSA'lar eksik bilgi ile çalışabilirler.
- YSA'lar hataya ve gürültüye karşı duyarlılığa ve toleransa sahiptirler.
- YSA'ların paralel çalışma prensibinden dolayı herhangi bir nöronda meydana gelen hata sistem çıkışını önemli ölçüde etkilemez. Etkilenme hücrenin ağırlıklarıyla orantılı olarak değişir. YSA'ların bu özelliğine dereceli bozulma (İng. graceful degradation) adı verilir.
- YSA'lar nümerik bilgiler ile çalışabilirler.
- YSA'ların doğrusal olmayışı hemen hemen her problemin çözümü için kullanılabilmelerini mümkün kılar.
- YSA'lar değişen sistem dinamiklerine adaptasyon sağlamak üzere tekrar tekrar eğitilebilirler.

3.3.2.2. YSA'ların avantajları

YSA'ları cazip kılan ve bu sayede yaygın kullanımlarını sağlayan pek çok özelliğinin bulunduğu belirtilmektedir. Bu özellikler şöyle sıralanabilir [25];

- YSA'lar bir sisteme ait çözüm üretirken sistemin modeline ihtiyaç duymazlar. Bu sayede matematiksel olarak modellenmesi zor olan ya da modellenmesi mümkün olmayan sistemlerde kullanılabilirler.
- YSA'ların kurallara bağlılıkları yoktur.
- YSA'ların öğrenme yetenekleri vardır ve YSA'ların eğitilmesi pek çok farklı algoritmayla gerçekleştirilebilir.
- YSA'lar çözümlenecek sistemin iç işleyişinin bilgisine ihtiyaç duyulmadan oluşturulabilirler. YSA'ların eğitilmesi yalnızca sisteme ait girdi-çıkı örneklerinin kullanılmasıyla sağlanır.
- Bir sisteme ait olaylar ve bu olaylar arasındaki ilişkileri anlamak zor olabilmektedir. YSA'lar bu tip ilişkilendirmeleri kendiliğinden yaparlar.
- Sisteme ait olaylar ve bu olaylar arasındaki ilişkiler doğrusal olmayabilir. Bu tür sistemleri modellemek zor olabilmektedir. Bu tip zorlukları aşmak için sistem hakkında bir takım varsayımlarda bulunmak gerekir. Bu da matematiksel model ile gerçek model arasında farklılık oluşmasını sağlar.

Ancak YSA'lar doğrusal olmayan sistemlerde geleneksel yöntemlerden daha gerçekçi çözümler üretebilir.

- YSA'ların kullanımı oldukça pratiktir. Sistem ile ilgili örneklerin toplanması ve bu örnekler ile YSA'nın eğitilmesi, sistemin çözümlenip ağıın uygulamada hızlıca kullanılabilmesi için yeterli olmaktadır.
- Sistem parametrelerinin değiştirilmesi durumunda YSA'nın tekrar eğitilmesi ve yeni sisteme uyum sağlaması oldukça kolaydır.
- YSA'ların paralel çalışma prensibi gerçek zamanlı uygulamalarda kullanımlarına olanak sağlamaktadır.

3.3.2.3. YSA'ların dezavantajları

YSA'ların henüz tüm sistemler için uygulanabilir olmaması, güvenilirliklerinin bazı uygulamalar için yetersiz kalması YSA'ların bazı dezavantajlarıyla ilişkilendirilebilir. Bu dezavantajlar şöyle sıralanabilir [25];

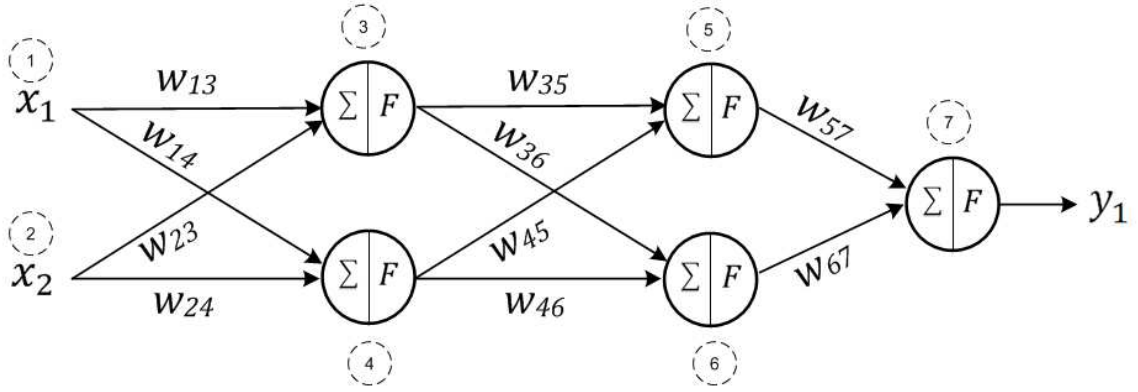
- YSA'lar kara kutu yapısındadır ve ağ içinde ne olup bittiği kullanıcıya açık değildir.
- YSA'ların uygulamaya göre model ve topoloji seçimi kullanıcının tecrübesine bağlıdır. Bu yüzden en uygun parametrelerin ne olduğunu bulabilmek için bir çok denemenin yapılması gerekebilir.
- YSA'ların eğitiminde kullanılacak örnek setinin seçiminde herhangi bir kural söz konusu değildir. Bu işlem de kullanıcının tecrübesiyle ilgili olarak değişiklik gösterebilir ve dolayısıyla YSA'nın farklı performanslarda sonuç üretmesine sebep olabilir.
- YSA'ların davranışlarının açık olmaması ağın güvenilirliğini azaltmaktadır.
- YSA'nın eğitimi bazen uzun zaman alabilir.
- Oluşturulan ağın en iyi sonuç veren ağ olduğunu bilmek mümkün değildir. Halbuki geleneksel yöntemlerle bulunan çözümlerin en uygun olup olmadıkları bilinebilmektedir.
- Sistemin bazı ihtimallerin çözümü için gereken örneklerin üretilmesi zor olabilmektedir; ya da doğru örneklerin neler olduğu kestirilemeyebilir. Bu da ağın sağlıklı çalışmamasına sebep olabilir.

3.3.3. YSA modelleri

Literatürde çok sayıda Yapay Sinir Ağı modeli bulunmaktadır. Bunlara, Çok Katmanlı Perseptron (İng. Multi Layer Perceptron, MLP), Öğrenme Vektörü Niceleme (İng. Learning Vector Quantization, LVQ), Elman ve Fordan Ağları, Radyal Tabanlı (İng. Radial Basis Function, RBF) yapay sinir ağı yapıları örnek olarak verilebilmektedir [28]. Bu ağlar arasında sıklıkla kullanılan MLP yapısı aşağıda anlatılmıştır.

3.3.3.1. Çok katmanlı perseptron (MLP)

MLP tipindeki YSA modelinin bir çok öğrenme algoritmasıyla kolaylıkla eğitilebilmesi bu modelin özellikle mühendislik uygulamalarında en yaygın kullanılan YSA modellerinden biri olmasını sağlamıştır [28]. MLP modelinin, bir giriş katmanı, bir ya da birden fazla ara katman ve bir adet çıkış katmanından oluştuğu bilinmektedir. Bir katmandaki bütün nöronlar bir önceki katmandaki diğer bütün nöronlara bağlıdır. MLP’de işlem tek yönde gerçekleşmekte ve geri besleme bulunmamaktadır. Bu nedenle bu modele ileri beslemeli (İng. feed forward) yapay sinir ağı denilmektedir. Ağın giriş katmanında herhangi bir işlem yapılmadığı bu katmanın girişlerin ilk ara katmandaki nöronlara dağılmalarını sağladığı bilinmektedir. Ara katman sayısı bir ya da birden fazla olabildiği gibi bu katmanlardaki nöron sayısı genellikle deneme yanılma yoluyla belirlenmektedir. Nihai çıkışların alındığı çıkış katmanında çıkış nöronları yer almaktadır. Bu katmandaki nöron sayısı uygulamaya bağlı olarak değişiklik göstermektedir. İki girişi, ikişer nörona sahip iki adet ara katmanı ve bir nörona sahip çıkış katmanı bulunan MLP yapısı Şekil 3.10’da gösterilmiştir.



Şekil 3.10 Örnek bir MLP YSA modeli

MLP eğitimi sırasında ağın giriş katmanından örnek veri girdileri yapılmaktadır. MLP çıkışının olması gereken sonuç ile farkı, yani hata, geri yönde ağırlıklar ve eşik değerlerine yansıtılmaktadır. Hata kabul edilebilir seviyeye ulaşınca kadar bu işlem tekrarlanmaktadır. Bölüm 3.3.4'de, MLP eğitiminde yaygın olarak kullanılan geri yayılım (İng. backpropagation) algoritması anlatılmıştır [28].

3.3.4. Geri yayılım (back propagation) algoritması

Geri yayılım algoritmasında, öğrenmenin ileri yönlü ve geri yönlü olmak üzere iki işlem fazından oluştuğu bilinmektedir [21]. İleri yönlü işlemde girişlere örnek girdi vektörü uygulanmaktadır. Bu girdi, katman katman işlenerek çıktı vektörü üretilmektedir. İleri yönlü olan bu işlemde ağın ağırlıkları sabit kalmaktadır. Geri yönlü işlemde ise tüm ağırlıklar bir hata düzeltme kuralına bağlı olarak değiştirilmektedir. Ağın çıkışı ile olması gereken çıkış arasındaki fark alınarak hata hesaplanmaktadır. Bu hata geri yönlü olarak tüm ağırlıklara yansıtılmakta, böylece ağ çıktısı olması gereken çıktıya oldukça yaklaşabilmektedir. Bunu sağlamak üzere ileri ve geri yönlü işlemler iteratif olarak uygulanmaktadır.

Geri yayılım ile öğrenme iki safhaya ayrılarak incelenebilmekte, bunlar; yayılım ve ağırlık güncelleme olarak sıralanabilmektedir.

- Yayılım:

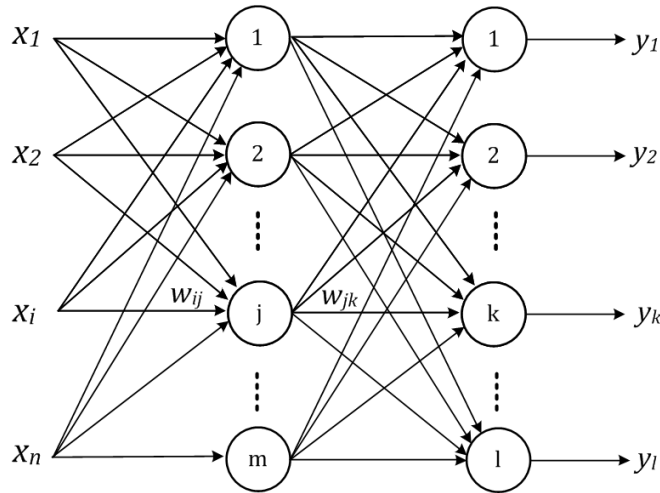
Her yayılım safhası şu iki adımı kapsar:

- İleri yönlü yayılımda ağa örnek giriş uygulanır ve ağın çıktısı elde edilir.

- Geri yönlü yayılda ağ çıktısı ile hedef çıktı arasındaki fark alınarak çıkış ve ara katman nöronlarının deltaları hesaplanır [21].
- Ağırlık Güncellemesi:
 - Her bir ağırlık için:
 - Ağırlığın ait olduğu hattın girişi ile hattın çıkışındaki nöronun deltası çarpılır ve böylelikle ağırlığın gradyanı hesaplanır
 - Hesaplanan gradyan belirli bir oranla çarpılır ve sonuç ağırlığın kendisinden çıkartılır [21].

Gradyanın çarpıldığı orana öğrenme katsayısı (İng. learning rate) adı verilmektedir. Bu oranın ağırlığın öğrenme hızını ve kalitesini etkileyen bir parametre olduğu bilinmektedir. Eğitim sırasında ağırlığın öğrenme hızı gözlenerek, öğrenme katsayısı artırılabilir ya da azaltılabilir. Optimum oran deneme yanılma ile bulunabilir.

Şekil 3.11’de tüm nöronlarının lojistik sigmoid aktivasyon fonksiyonuna sahip olduğu bilinen bir MLP için geri yayılım algoritmasının uygulama adımları anlatılmıştır [29].



Şekil 3.11 Geri yayılım uygulanan MLP ağı

Adım 1 – İklendirme

Bu aşamada; ağırlıkların tüm ağırlık ve eşik (bias) değerleri rastgele sayılar olarak atanmaktadır.

Adım 2 – Aktivasyon

Aktivasyon aşamasında $x_1(p), x_2(p) \dots x_n(p)$ girişlerine karşılık $y_{d,1}(p), y_{d,2}(p), y_{d,3}(p)$ hedef çıkış değerleri kullanılmaktadır.

Ara katmanlardaki nöron çıkışları hesaplanır (bkz. (3-9)).

$$y_j(p) = \text{sigmoid} \left[\sum_{i=1}^n x_i(p) \cdot w_{ij}(p) - \theta_j \right] \quad (3-9)$$

Daha sonra çıkış katmanındaki nöron çıkışları hesaplanır (bkz. (3-10)).

$$y_k(p) = \text{sigmoid} \left[\sum_{j=1}^m x_{jk}(p) \cdot w_{jk}(p) - \theta_k \right] \quad (3-10)$$

(3-9) ve (3-10) ifadelerinde θ nöronun eşik değerini (bias), p ise iterasyon sayısını göstermektedir.

Adım 3 – Çıkış nöronları için ağırlık ve eşik değerlerinin güncellenmesi

Bu adımda ilk olarak, çıkış nöronları için hata (e) ve hata gradyanı (δ) hesaplanmaktadır (bkz. (3-11),(3-12),(3-13)).

$$e_k(p) = y_{d,k}(p) - y_k(p) \quad (3-11)$$

$$\delta_k(p) = \frac{\partial y_k(p)}{\partial X_k(p)} \times e_k(p) \quad (3-12)$$

$$\delta_k(p) = y_k(p) \cdot [1 - y_k(p)] \cdot e_k(p) \quad (3-13)$$

Daha sonra, çıkış nöronlarına bağlı ağırlıkların ve çıkış nöronlarının eşik değerlerinin güncellenmesi için kullanılacak fark değerleri hesaplanır (bkz. (3-14),(3-15)).

$$\Delta w_{jk}(p) = \alpha \cdot y_j(p) \cdot \delta_k(p) \quad (3-14)$$

$$\Delta \theta_k(p) = \alpha \cdot (-1) \cdot \delta_k(p) \quad (3-15)$$

Bulunan fark değerleri ağırlık ve eşiklere eklenerek çıkış katmanında güncelleştirme işlemi tamamlanmaktadır (bkz. (3-16),(3-17)).

$$w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \quad (3-16)$$

$$\theta_k(p+1) = \theta_k(p) + \Delta \theta_k(p) \quad (3-17)$$

Adım 4 – Ara katman nöronları için ağırlık ve eşik değerlerinin güncellenmesi

Bu adımda ilk olarak ara nöronlar için hata gradyanı hesaplanmaktadır (bkz. (3-18)).

$$\delta_j(p) = y_j(p) \cdot [1 - y_j(p)] \cdot \sum_{k=1}^l \delta_k(p) w_{jk}(p) \quad (3-18)$$

Sonraki aşamada, ara katman nöronlarına bağlı ağırlıkların ve ara katman nöronlarının eşik değerlerinin güncellenmesi için kullanılacak fark değerleri hesaplanır (bkz. (3-19),(3-20)).

$$\Delta w_{ij}(p) = \alpha \cdot x_i(p) \cdot \delta_j(p) \quad (3-19)$$

$$\Delta \theta_j(p) = \alpha \cdot (-1) \cdot \delta_j(p) \quad (3-20)$$

Daha sonra, bulunan fark değerleri ağırlık ve eşiklere eklenerek ara katmanlarda güncelleştirme işlemi tamamlanır (bkz. (3-21),(3-22)).

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p) \quad (3-21)$$

$$\theta_j(p+1) = \theta_j(p) + \Delta \theta_j(p) \quad (3-22)$$

Adım 5 – Yineleme (İterasyon)

Yineleme aşamasında, önce p değeri bir arttırılır, daha sonra Adım 2 ye dönülür ve istenen hata kriteri sağlanana kadar Adım 2-5 tekrarlanır [29].

4. PROGRAMLANABİLİR KAZANÇLI YÜKSELTEÇ (PGA)

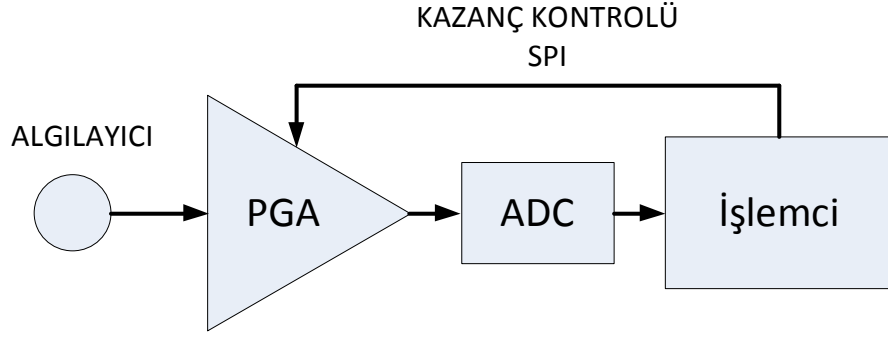
4.1. Programlanabilir Kazançlı Yükselteçlere Genel Bakış

Geniş bir dinamik aralığa sahip çoğu veri toplama sistemlerinin, analogtan sayısala çevirici (İng. analog to digital converter, ADC) devrelerine giden sinyal seviyelerini yükseltmek ya da düşürmek için bir takım metodlara ihtiyaç duydukları belirtilmektedir [30]. Tipik olarak ADC'lerin tam aralıkları (İng. full scale) 2V ile 10V arasında yer almaktadır. ADC'nin teknik özelliklerinde yer alan hassasiyet değerinin geçerli olabilmesi için, en yüksek giriş sinyalinin ADC'nin müsaade ettiği üst voltaj sınırına yakın bir değerde olması gerekmektedir [30].

Transdüserler geniş bir aralıkta çıkış voltajı üretebilmektedirler. Düşük algılayıcı voltajlarının yükseltilebilmesi için yüksek kazanç ihtiyacı duyulur. Fakat bu durumda yüksek seviyedeki bir girişin yüksek bir kazançla tekrar yükseltilmesi de ADC'nin doyuma ulaşmasına neden olabilmektedir. Bu yüzden kazancı öngörülse olarak kontrol edilebilen bir ağıta ihtiyaç duyulmuştur. Programlanabilir kazanç sahip yükselteçlerin geniş bir uygulama yelpazesine sahip oldukları belirtilmektedir [30]. Bu uygulama alanlarından bazıları şunlardır.

- Enstrümantasyon
- Fotodiyot devreleri
- Ultrasound önyükselteçleri
- Sonar uygulamaları
- Geniş dinamik aralıklı algılayıcılar
- ADC girişi için sinyal koşullandırma devreleri
- Otomatik kazanç kontrol (İng. Automatic Gain Control, AGC) döngüleri

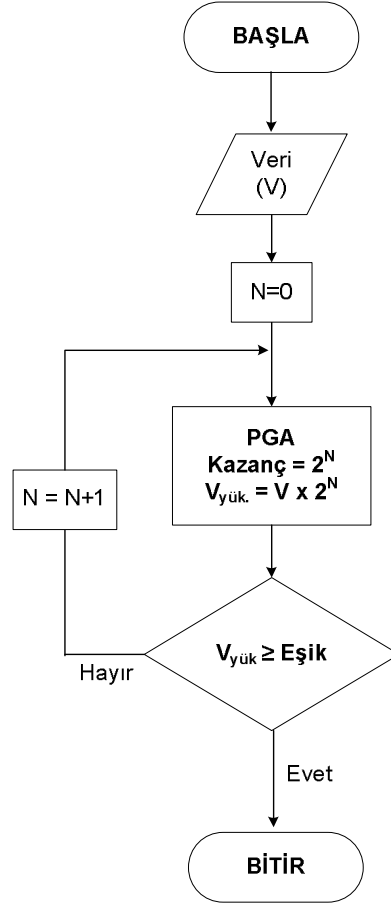
Programlanabilir kazanç özelliğindeki bu ağıtların kazançları genellikle sayısal girdiler ile kontrol edilir. Bu tip yapılara Programlanabilir Kazançlı Yükselteç (İng. Programmable Gain Amplifier – PGA) denmektedir. Tipik PGA'lar genellikle ya 10, 100, 1000 gibi 10'un üsleri; ya da 1, 2, 4, 8 gibi 2'nin üsleri değerlerindeki kazançlara ayarlanabilirler. Uygulamanın gerekliliklerine göre hangi tip PGA'nın seçileceğine karar verilebilir [30].



Şekil 4.1 Veri toplama sistemlerinde PGA

PGA'lara giren sinyaller çeşitlilik göstermektedir. Bazı sinyaller geniş bant aralığına ihtiyaç duyarlar. Bazılarının gürültüye toleransları düşüktür. Bazı sinyaller yüksek bazıları düşük empedanslı kaynaklardan üretilirler. Bazı sinyaller diferansiyel olabilir. Bunun yanısıra PGA çıkışının bazen doğrudan ADC girişini sürmesi beklenir; ya da PGA daha küçük bir alt sistemin (örneğin bir AGC döngüsünün) parçası olabilir [30].

PGA'lar Bölüm 2.5'de anlatılan hibrit yöntemlerde kullanılırlar. PGA genelde bir algılayıcı ve onun ADC'si arasında yer alır (Şekil 4.1). PGA'ların kazanç kontrolleri piyasada yaygın bulunan PGA entegrelerinin desteklediği SPI, I²C gibi seri arayüzler kullanılarak yapılır. Uygulamaya bağlı olarak ek sinyal koşullandırma işlemleri PGA'dan önce ya da sonra gerçekleştirilebilir. Örneğin fotodiyotlar, kendileri ve PGA arasında bir akım-voltaj çeviricisine ihtiyaç duyar. Diğer birçok sistemde ise önce sinyali yükseltmek ardından yükseltilmiş sinyali koşullandırmak daha iyi bir yöntemdir [30]. PGA'ların genel kullanım mantığı ise Şekil 4.2'de verilmektedir.



Şekil 4.2 PGA'ların genel kullanım mantığı

Çözünürlüğü 10mV olan bir ADC'nin girişine 5mV uygulandığında bu sinyal sistem tarafından algılanamayacaktır. Ancak uygulamada bir ve iki olmak üzere iki farklı kazanç değeri olan bir PGA kullanıldığında ve PGA'nin kazancı iki olarak ayarlandığında, 5mV'luk bir sinyal yükseltilerek 10mV seviyesine ulaşacak ve artık algılanabilir hale gelecektir. Sistemin işlemcisi de PGA kazanç bilgisini ve sayısal çıktıyı kullanarak gerçek veriye ulaşacaktır. Bu sayede ADC çözünürlüğü bir bit artmış olacaktır [30].

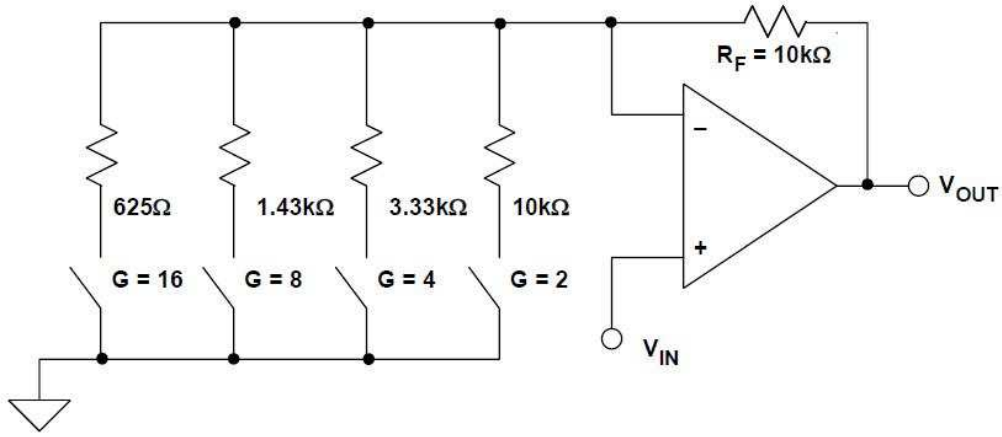
4.2. PGA Tasarımı

Pratikte PGA'ler ideal olmadıklarından PGA'lardan kaynaklanan hataların göz önünde bulundurulması ve bunları aşmak için çalışılması gerekmektedir. PGA tasarımında dikkat edilmesi gereken başlıca konular şunlardır [30].

- Kazancın nasıl anahtarlanacağı
- Anahtarlama yolu direncinin (R_{ON}) etkisi

- Kazanç hassasiyeti
- Kazanç doğrusallığı
- Bant genişliği, frekans ve kazanç ilişkisi
- DC offset
- Sıcaklığa bağlı kazanç ve offset kayması
- Anahtarlamanın ardından oturma zamanı (İng. Settling Time)

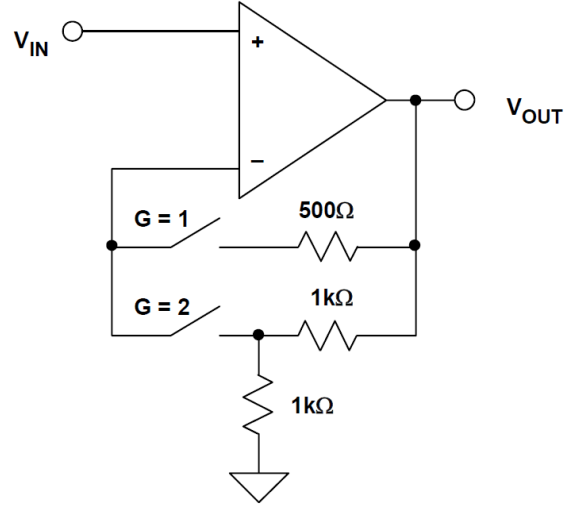
En temel PGA problemlerinden biri kazanç programlamadaki hassasiyettir. Elektromekanik röleler minimum R_{ON} direncine sahiptir fakat yavaş, büyük ve pahalı olduklarından kazanç anahtarlama için en iyi çözüm değildirler. CMOS anahtarlar küçüktür fakat R_{ON} dirençleri voltaj ve sıcaklığa bağlı olarak değişir. Bu da PGA'nın AC karakteristiğini doğrudan etkileyen bir faktördür [30].



Şekil 4.3 Düşük performanslı bir PGA tasarımı (Jung,[30]'dan değiştirilerek)

R_{ON} direncinin PGA performansı üzerindeki etkisini anlamak için Şekil 4.3'deki devre incelenebilir. Şekilde evirmeyen konfigürasyonda kullanılan bir op amp dört farklı kazanç direncine bağlanmıştır. R_{ON} direncinin 25Ω olduğu varsayılırsa, kazanç 16 iken çıkış voltajında %2,4'lük bir hata meydana gelmektedir ve bu durum PGA'nın çözünürlüğünün 8 bitten daha düşük olacağını göstermektedir.

Bu sorunu çözmek için direnç değerleri yükseltilebilir ancak bu sefer de gürültü ve offset problemleri baş gösterecektir. Bu yüzden verilen devrenin hassasiyet sorununa tek çözüm anahtarlama elemanı olarak rölelerin kullanılması olacaktır.



Şekil 4.4 R_{ON} direncinin etkisini en aza indiren alternatif PGA tasarımı

PGA devresinin R_{ON} direncinden etkilenmemesi arzu edilen bir çözümdür. Şekil 4.4’de verilen devrede anahtarlama elemanı op-amp’ın eviren girişine seri olarak bağlanmıştır. Op-amp’ın giriş empedansının çok yüksek olmasından dolayı bu dirence seri bağlanan R_{ON} direnci artık kazançta etki etmemekte ve PGA’nın kazancını yalnızca harici dirençler belirlemektedir. Şu da not edilmelidir ki eğer op-amp’ın bias akımı yüksekse, R_{ON} direnci bir miktar offset hatasına sebep olur; fakat evirmeyen girişe, R_{ON} ile eşdeğer bir direnç bağlandığında bu sorun da ortadan kalkacaktır [30].

4.3. Uygulamada Kullanılan PGA’nın Yapısı

Bu tez kapsamında gerçekleştirilen uygulamada kullanılan PGA’nın ayrıntıları “7.3.2 Kullanılan PGA’nın özellikleri” bölümünde verilmiştir.

5. ALAN PROGRAMLANABİLİR KAPI DİZİLERİ (FPGA)

5.1. Programlanabilir Donanımların Tarihsel Gelişimi

Sayısal elektroniğin ilgi alanı, bilgiyi belirli bir sayıda çıktı ile ifade edebilen devrelerdir. Uygulamaların çoğu gerçekte yalnızca '0' ve '1' olmak üzere iki farklı durumu kullanırlar. Bu seçimin arkasında yatan sebep George Boole tarafından geliştirilen Boolean matematiğinin mantık problemlerinin çözümünde pratik bir şekilde kullanılabilir olması ve ikilik sayı sistemiyle ifade edilen bu aritmetiğin yeterince olgunlaşmış bir uygulama alanı olmasıdır.

'0' ve '1' ile gösterilen iki durumun voltaj veya akım karşılıkları farklı bir takım mantık ailelerini oluşturmaktadır. Örneğin Transistor-Transistör Logic (TTL) adı verilen ailede bulunan elemanlar '1' durumu için belirli bir voltaj eşiğinin (tipik olarak 2.4V) üstünü işaret eder. Aynı ailenin elemanları için, eğer '1' durumu için giriş eşiğini 2V olarak tanımlarsak, 0.4V luk bir gürültü payı (İng. noise margin) elde etmiş oluruz ki bu da TTL entegrelerin bir tasarımda mantık seviyelerinde tutarsızlık oluşmadan birlikte kullanımına olanak sağlar. Bu özellik bilginin belirli bir miktar gürültü altında dahi korunabilmesini sağlar. İkilik sistemde değerlerin yalnız iki adet voltaj veya akım seviyeleriyle gösterilmesi ikilik sistemin neden tercih edildiğini ve tasarım aşamasında bu seçimin gürültü payları açısından nasıl bir avantaj sağladığını açıklamaktadır.

1970'lerde 74 serisinden olan TTL entegreleri sayısal sistem tasarımına hız kazandırmıştır. Ardından tasarımcılar farklı mantıksal fonksiyonları programlayabilmeyi arzu etmişlerdir. Burada programlamadan kasıt bir entegrenin üretiminden sonra müdahaleye açık olması ve mantıksal işleyişinin değişime imkan sağlamasıdır.

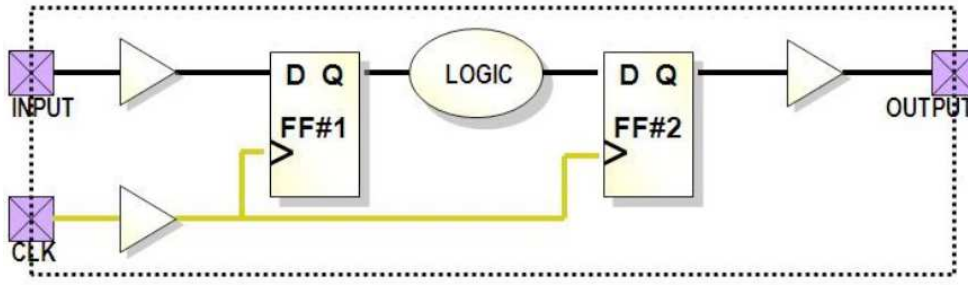
Programlanabilirlik konusundaki ilk gelişmeler yalnızca NAND kapılarından oluşmuş kapı dizilerinin ortaya çıkmasıyla başlamıştır. Bu NAND dizileri kullanılarak tasarımcılar istedikleri fonksiyonları gerçekleyebilmekteydiler. Bu tip uygulamaların yalnız üretimden önce, tasarım aşamasında yapılabildiği, bir takım hataların oluşması durumunda ise tasarıma baştan başlanmasının gerekli olmadığı ifade edilmektedir [31]. 1980'lerde Programlanabilir Mantık Dizileri'nin

(İng. Programmable Logic Arrays, PLA) piyasaya çıkmasıyla birlikte gerçek anlamda programlama çözümünün üretilmiş olduğu rapor edilmektedir. Bunlar bağlantıları programlanabilen iki seviyeli AND-OR yapılarıdır. Programlanabilir Dizi Mantığı (İng. Programmable Array Logic, PAL) aygıtları ise PLA yapısından daha gelişmiş performans sunan entegrelerdir. Günümüzde bu türlerin tamamı Programlanabilir Mantık Aygıtları (İng. Programmable Logic Devices, PLD) olarak da bilinmektedir.

Programlama konusundaki bir sonraki gelişmenin ise; karmaşık PLD'lerin (İng. Complex PLD, CPLD) üretimi olduğu belirtilmektedir [31]. CPLD'ler de birçok PLD'nin birleşiminden oluşmuş programlanabilir bağlantılara sahip yapılardır. Alan Programlanabilir Kapı Dizileri (İng. Field Programmable Gate Arrays, FPGA) ise çok daha fazla sayıda basit bloklardan ve iç bağlantılardan oluşan, programlanabilir donanım ailesinde en ön sırayı almış sayısal devrelerdir [31].

5.2. FPGA'lerle Tasarıma Genel Bakış

FPGA içerisinde bulunan tipik bir mantık devresi, Şekil 5.1'de gösterildiği gibi flip-flop dizileri arasına sıkıştırılmış tümleşik mantık bloklarından oluşmaktadır. Tümleşik bloklar, çıkışı yalnızca girişine bağlı olan alt devreler olarak bilinmektedir. İki girişli AND, OR gibi tüm iyi bilinen basit mantıksal fonksiyonlar bu gruba dahildir. Herhangi bir karmaşıklık seviyesinde olan bir fonksiyon bu basit bloklar kullanılarak gerçekleştirilebilir. Yol seçiciler (İng. multiplexer), kodlayıcılar (İng. encoder), kod çözücüler (İng. decoder) gibi yapılar tümleşik bloklara örnekler olarak verilebilmektedirler. Şekil 5.1'de tipik bir mantıksal tasarım devresi yer almaktadır. Bu devrenin girdisi bir çok bitten oluşabilir. Devreye ayrıca kare dalga şeklinde sabit frekanslı bir saat sinyali de sağlanmaktadır. Devredeki flip-floplar da aynı saat sinyaliyle beslenip D girişlerindeki sinyali saat sinyalinin her yükselen veya düşen kenarında Q çıkışlarına yansıtırlar. Anlık bir geçiş durumu haricinde, D girişinin Q çıkışına bağlantısı bulunmamaktadır [31].



Şekil 5.1 Tipik bir mantıksal tasarım devresi [31]

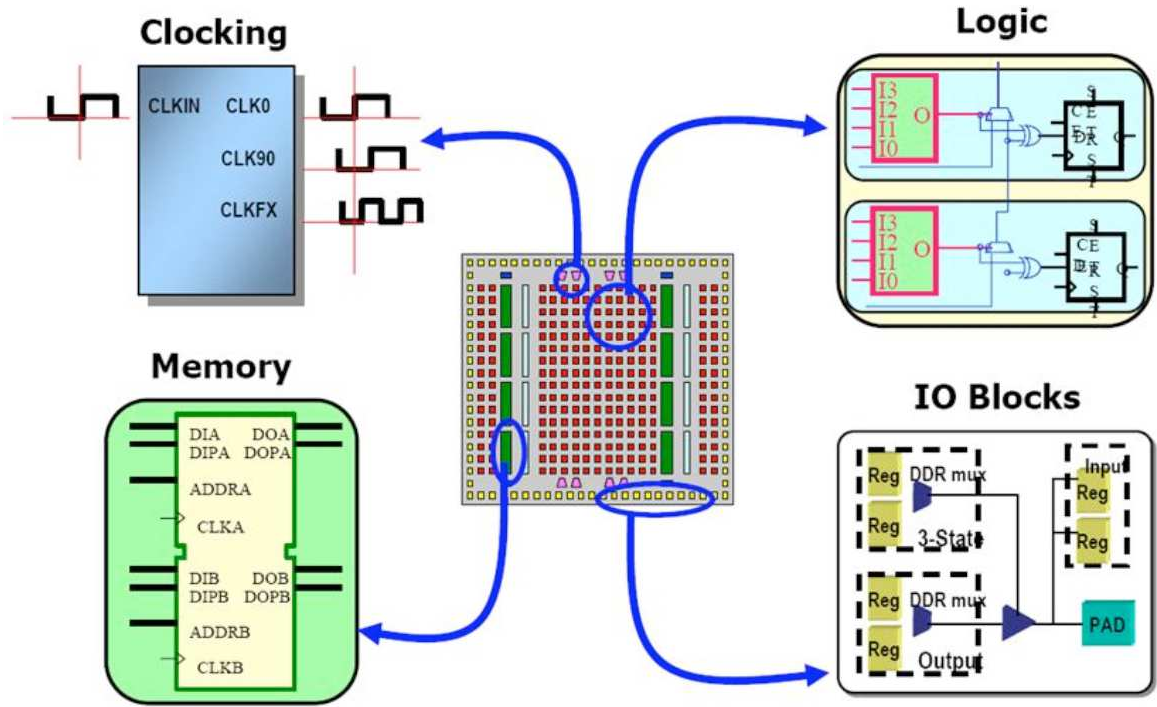
Verilen bu devrenin yapısı oldukça basittir ve bu haliyle bir blok olarak kullanımı hemen hemen tüm sayısal tasarımların gereksinimlerini karşılayabilecek güçtedir; öyle ki bir parçacık hızlandırıcı kontrolcüsünün dahi tasarımı bu bloklar ile gerçekleştirilebilir. Tasarımcı bir bloğa ait girişin, çıkışa olan etkisinin bir saat periyodu içerisinde tamamlandığından emin olmalıdır. Bu koşul sağlandığında, saat sinyalinin bir sonraki yükselen kenarı gelmeden, sıradaki bloğun girişleri kararlı hale gelmiş olacaktır. Son zamanlarda bu tip zamanlama uyumları bilgisayarlar tarafından otomatik gerçekleştirilebilmekte ve bu sayede tasarımcı yalnızca devrenin fonksiyonel davranışlarına odaklanabilmektedir [31].

Eşzamanlı tasarım olarak bilinen bu tasarım stratejisinin en büyük artışı zamanlama analizlerinin oldukça kolay gerçekleştirilebilmesini sağlamasıdır. Örneğin bir tümleşik devrenin çıkışının başka bir flip-flopun saat sinyali olarak kullanıldığı bir tasarımda, benzetim aşaması oldukça iyi sonuçlar verirken, gerçek uygulamada oluşan küçük sinyal gecikmeleri devrenin çalışamaz hale gelmesini sağlayabilmektedir. Eşzamanlı tasarımda ise bir takım zaman payları koymak suretiyle önlemler alınarak bu küçük gecikmelerin sebep olduğu istenmeyen sonuçlar kolayca ortadan kaldırılabilir [31].

5.3. FPGA'lerin Genel Yapısı

Modern FPGA'ler tasarımcıya, bağlantıları programlanabilen bir tümleşik (İng. combinatorial) blok ve flip-flop havuzu sunarlar. Buna ek olarak üreticiler tasarımların bellek kullanımına sıkça ihtiyaç duyduklarını iyi bilirler ve bu nedenle entegrilerinin içine farklı kapasitelerde Rasgele Erişilebilir Bellekler'i (İng. Random Access Memory, RAM) de dahil ederler. Ayrıca saat sinyalinin koşullandırılmasını destekleyen DLL'ler (Delay Locked Loops) ve PLL'ler (Phase

Locked Loop) FPGA entegreleri içinde yerlerini almaktadırlar. FPGA entegreleri tek başına çalışan, diğer çevre elemanlarından bağımsız aygıtlar değildir. FPGA'ların diğer entegrelerle ve harici sinyallerle kolayca iletişim kurması gerekmektedir. Bu nedenle de üreticiler esnek olarak kullanılabilen çok sayıda giriş/çıkış (I/O) bloklarını entegrelerine dahil etmektedirler. Her I/O bloğu giriş, çıkış ya da hem giriş hem de çıkış olarak kullanılabilir. FPGA'ların desteklediği elektriksel sinyal standartları oldukça geniş bir yelpazede olmakla birlikte, bu entegrelere özgü bantgenişliğini arttıran bir takım yöntemler de bulunmaktadır. Saat sinyalinin hem yükselen hem de düşen kenarında işlem yapabilmek bu yöntemlere örnek olarak verilebilmektedir [31].



Şekil 5.2 FPGA'nın dahili yapısı (Xilinx Inc.)

FPGA'nın Şekil 5.2'de gösterilen kısımları entegrenin %20'sinden daha az bir kısmını oluşturmaktadır. Geri kalan geniş silikon bölgede ise programlanabilir ara bağlantılar ve programlamayı gerçekleştiren yardımcı bloklar yer almaktadır. Burada, silikonun fazla miktarlarda ve verimsiz olarak kullanımı programlanabilirliğin bedeli olarak karşımıza çıkmaktadır. Ayrıca bu durum Uygulamaya Özel Entegre Devreler'in (İng. Application Specific Integrated Circuits – ASIC) yüksek hacimlerdeki üretimlerde kullanılıp, FPGA'ların ise ancak düşük

kapasitelerdeki kullanımlarının nedenini ortaya koymaktadır. Tüm bu olumsuzluklara rağmen günümüzde, ASIC'lerin ve DSP'lerin baskın olduğu alanlarda FPGA'lerin kullanımı Moore Yasası'nın da etkisiyle hızla artmaktadır.

Üreticiler, FPGA'lerdeki silikon verimsizliğinin üstesinden gelmek için, sıklıkla kullanılan bazı fonksiyonlara ait hazır blokları entegrelerinin içine dahil etmektedirler. İçeriği programlanamayan bu hazır bloklar genel amaçlı işlemciler, yüksek hızlı seri arayüzler, aritmetik işlemler ve Ethernet MAC (Medium Access Control) ünitelerinden oluşmaktadır [31].

5.4. Tasarım Akışları

Herhangi bir problemin çözümüne başlayan bir tasarımcının başlangıçtan nihai donanıma kadar uzanan süreçte, bir seri tasarım aşamalarını takip etmek zorunda olduğu bilinmektedir. Bu aşamaların tamamına "tasarım akışı (İng. design flow)" adı verilmektedir. Tasarıma başlamadan önce tüm gereksinimler belirlenmelidir. FPGA üreticilerinin kendi entegreleri için sundukları araçlar bu fazda tasarımcıya yardımcı olmayacak, sadece tasarımın çalışan bir donanıma dönüştürülmesi aşamasında rol oynayacaklardır [31].

FPGA'larda sık kullanılan tasarım akışı; tasarım girdisi, sentez, yerleştirme ve rotalama ve konfigürasyon aşamalarından oluşmaktadır. Aşağıda bu aşamalar incelenmiştir.

5.4.1. Tasarım girdisi

Bu aşama, tasarım fikirlerinin makine gösterimleri şekline dönüştürülmesini kapsamaktadır. Bu işlem en çok Donanım Tanımla Dili (İng. Hardware Description Language, HDL) kullanılarak gerçekleştirilmektedir. En popüler donanım tanımlama dilleri Verilog ve VHDL (Very High Speed Integrated Circuit HDL) olarak bilinmektedir. HDL'nin kelime anlamından da anlaşılacağı üzere bu diller kullanıcının tasarladığı var olan bir fikri "tanımlamak" için kullanılan dillerdir ancak elektronik devreler tasarlamak için kullanılan araçlar değildir. HDL'lerin sıralı işlem yapma konseptini desteklemediği, bu nedenle de geleneksel programlama dillerinden farklı olduğu bilinmektedir. Yani bir HDL kodunun şematik gösterimi ele

alındığında, üstte kalan parçanın altta kalan parçadan daha önce işleneceği düşünülmemelidir [31].

5.4.2. Sentez

Sentez araçları HDL kodunu ve seçilen bir üreticiye ait FPGA modelini girdi olarak almaktadır. Bu iki bilgiyi kullanarak, HDL kodunda tanımlanan fonksiyonu gerçeklemek üzere ilgili donanımın dahili bağlantılarının ne olması gerektiğini gösteren bir ağ listesi (İng. netlist) üretmektedir. Çoğu sentez araçları buna ek olarak optimizasyon, kaydedici (İng. register) ve yük dengeleme, zamanlama performansının artırılması gibi başka teknikler de uygulayarak HDL koduna ait en verimli ağ listesini oluşturmaktadırlar [31].

5.4.3. Yerleştirme ve rotalama

Yerleştiricinin (İng. placer), sentezlenmiş ağ listesini alıp, tasarımın temel bloklarının yerleştirilmesi için entegredeki en uygun yerleri seçtiği Rotalayıcı'nın (İng. router) ise tüm bu temel bloklar arasındaki bağlantıları yapıp, zamanlama kısıtlarını sağladığı bilinmektedir. Bir tasarım için en belirgin kıstas sistem saat sinyalinin frekansıdır. Ancak bunun dışında tasarımcının üretici yazılım araçlarını kullanarak tasarıma uyguladığı daha spesifik zaman kıstasları da olabilmektedir [31].

5.4.4. Konfigürasyon

FPGA'lar genellikle açılış esnasında harici bir kalıcı depolama aygıtı tarafından (örneğin flash bellekler) konfigüre edilebilirler. Yerleştirme ve rotalama işlemleri tamamlandıktan sonra, FPGA'da bulunan her bir programlanabilir elemanın konfigürasyonu için yapılan seçimler harici depolama aygıtında bir dosya içinde kayıt edilmelidir. FPGA'lar doğrudan da konfigüre edilebilir ancak bu, bir sonraki güç çevriminde FPGA konfigürasyonunun kaybolmasına neden olmaktadır. Her bir güç çevriminde FPGA'nın aynı işlevleri gerçekleştirebilmesi için harici bir depolama aygıtı tarafından baştan konfigüre edilmesi gerekmektedir [31].

6. TERMİSTÖRLER

6.1. Termistörlere Genel Bakış

Piyasada çeşitli uygulamaların gereksinimlerini karşılayabilen pek çok farklı özellikte sıcaklık algılayıcısının yer aldığı bilinmektedir. Bu algılayıcıların başlıcaları arasında ısıçift (İng. thermocouple), ısıya duyarlı dirençler (termistörler) ve silikon bazlı entegreler yer almaktadır [33].

“Termistör” kelimesi İngilizce “THERMally Sensitive ResISTOR” kelimelerinden türemiştir. Termistörler; Pozitif Sıcaklık Katsayılı termistör (PTC) ve Negatif Sıcaklık Katsayılı termistör (NTC) olmak üzere iki gruba ayrılmaktadır. NTC’lerin sıcaklıkları arttığında, doğrusal olmayan bir şekilde dirençleri azalmakta, PTC’lerin ise sıcaklıkları arttığında dirençleri doğrusal olmayan bir ilişkiyle artmaktadır. NTC’ler hassas sıcaklık ölçümü gereken uygulamalar için; PTC’ler ise sıcaklığa bağlı anahtarlama ihtiyacı olan devreler için idealdirler. Bu tez kapsamında gerçekleştirilen uygulamada NTC tipinde algılayıcı kullanılmıştır.

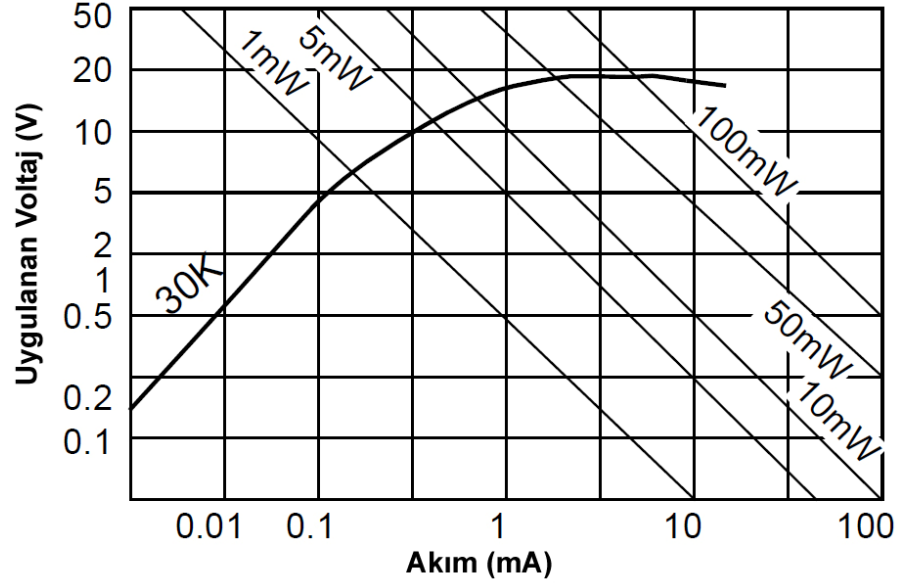
NTC’lerin üç farklı modda kullanıldıkları ifade edilmektedir [33]. Her bir mod farklı tipteki uygulamaların ihtiyaçlarını karşılamaktadır. Kullanılan bu üç mod aşağıdaki gibidir.

- Akım-voltaj modu
- Zaman-akım modu
- Sıcaklık-direnç modu

6.1.1. Akım-voltaj modu

Akım-Voltaj modunda, kendi üzerinde harcanan güçle ısınan bir ya da birden fazla NTC termistör kullanılmaktadır. Termistörün bu modda kullanıldığı uygulamalar genellikle akış ölçüm uygulamalarıdır. Termistörün sıcaklığı kendi üzerinden geçen elektriksel akımdan dolayı artar. Termistör yüzeyine temas eden sıvı ya da gaz akışı ise elemanın üzerindeki ısıyı atma miktarını değiştirmektedir. Bu sayede termistörün direnci, yüzeyine temas eden sıvı ya da gaz akışının termistörü soğutmasıyla birlikte değişmektedir. Bu tür uygulamalarda termistör boyutunun

küçük olması termistörün sistemin geri kalanının yaydığı ısıdan etkilenmemesi açısından önemlidir. Termistörün bu modu için kullanışlı bir akım-voltaj grafiği Şekil 6.1'de verilmiştir [33].



Şekil 6.1 Akım-voltaj modu NTC karakteristiği eğrisi [33]

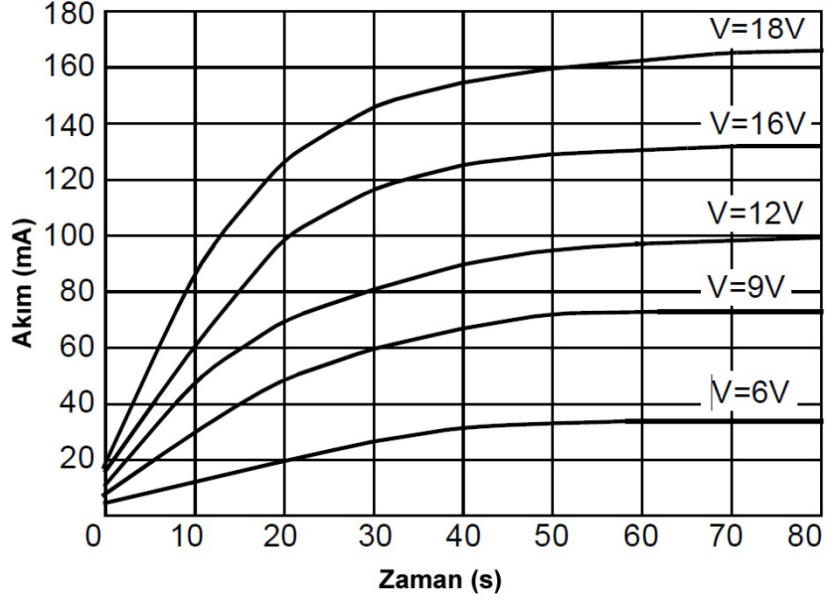
Vakum manometreleri, rüzgarölçerler, sıvı seviyesi kontrolcüler, sıvı hızı ve gaz tespit uygulamaları, termistörlerin akım-voltaj modunda kullanıldığı uygulamalardan bazılarıdır.

6.1.2. Zaman-akım modu

Termistörün zaman-akım karakteristiği, termistör paketinin ısı atma kapasitesine ve ısı sığasına bağlı olarak değişmektedir. Termistöre akım verildiğinde, termistör paketi ısınmaya başlayacaktır. Eğer akım uygulanmaya devam edilirse termistörün direnci zamanla azalacaktır. Termistörün bu zaman-akım karakteristiği kısa süreli yüksek voltaj darbelerinin etkisini yavaşlatmak için kullanılabilir. Bu modun bir başka kullanım alanı da rölelerin gürültüden dolayı yanlışlıkla tetiklenmesinin önlenmesidir.

Termistörün zaman-akım karakteristiğinin grafiği Şekil 6.2'de verilmiştir. Termistörün bu moddaki tepki hızı diyot ve silikon bazlı sıcaklık algılayıcılarına kıyasla oldukça hızlıdır. Diyot ve silikon bazlı sıcaklık algılayıcılarının paket

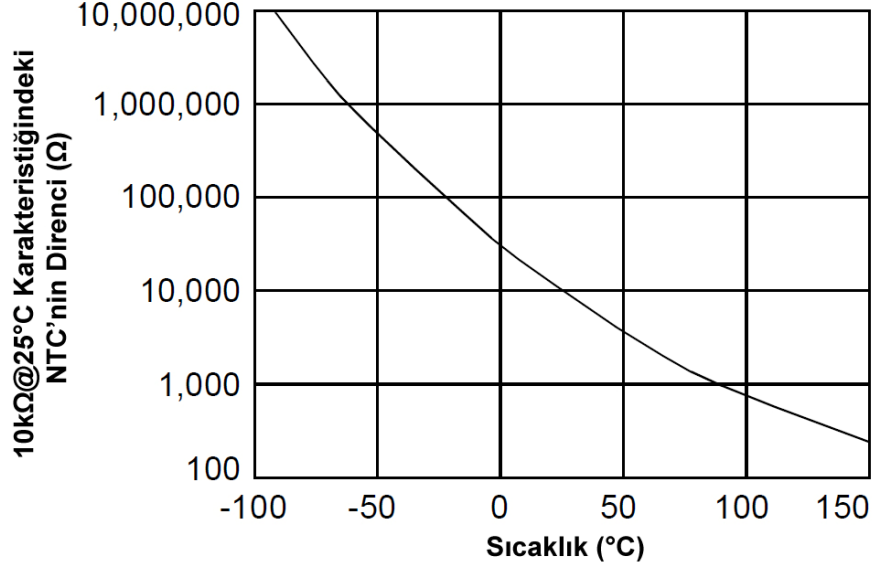
sıcaklıklarının doyuma ulaşması dakikalar alabilmektedir. Termistörlerin bu modu genellikle geciktirici, sıralı anahtarlama, elektriksel darbe emme ve ani akım limitleme uygulamalarında kullanılmaktadır [33].



Şekil 6.2 Termistörün zaman-akım karakteristiği [33]

6.1.3. Sıcaklık-direnç modu

Sıcaklık-direnç modu termistörlerin uygulamalarda en sık kullanıldıkları mod olarak bilinmektedir. Termistörlerin bu modda kullanıldığı devrelerde genellikle hassas sıcaklık ölçümü, kontrol veya kompanzasyon işlemleri gerçekleştirilir. Diğer iki modun tersine bu modda kullanılan termistörler “sıfır güç sarfiyatı” koşulunda çalışmaktadırlar. Bu koşul termistörün voltaj ya da akım ile uyarılması sırasında kendi üzerinde kendini ısıtabilecek kadar ısı üretmediğini ifade etmektedir. 10k Ω 'luk bir NTC'nin sıcaklık-direnç eğrisi Şekil 6.3'de verilmiştir [33].



Şekil 6.3 10kΩ'luk bir NTC'nin sıcaklık-direnç eğrisi [33]

NTC'lerin oda sıcaklığındaki dirençleri 1kΩ ile 10MΩ arasında olmaktadır. NTC'lerin paketleri uygulamanın izolasyon ya da dayanıklılık ihtiyacına bağlı olarak değişkenlik gösterse de, bu elemana giden kablo sayısı sabit olup yalnızca iki adettir. Bu kabloların sıcaklığa bağlı direnç değişimleri termistörün direnç değişimine kıyasla ihmal edilecek seviyededir. Bu nedenle Dirençsel Sıcaklık Algılayıcı (İng. Resistance Temperature Detector, RTD) elemanlarda olan dört kablolu bağlantı termistörlerde gerekli değildir.

Şekil 6.3, termistörlerin doğrusal olmadığını göstermektedir. Her ne kadar termistörlerin doğrusallığı, ısı çiftlerinden çok daha iyi olsa da termistörlü algılayıcı uygulamalarında doğrusallaştırmaya ihtiyaç vardır. Termistörün doğrusal olmayan davranışı yazılımsal olarak üçüncü derece polinom kullanılarak, başvuru çizelgesi kullanılarak ya da yapay sinir ağı algoritmaları kullanılarak doğrusallaştırılabilmek mümkündür. Bunun yanısıra termistör verisini sayısallaştırmadan önce de bir takım donanımsal doğrusallaştırma önlemleri alınabilmektedir. Doğrusallaştırma işleminde kullanılan üçüncü derece polinoma Steinhart-Hart Termistor eşitliği denmektedir. Bu eşitlik termistörün giriş-çıkış karakteristiğinin yaklaşık ifadesidir. Endüstride yaygın olarak kabul gören bu eşitlik hassas termistör hesaplamaları için kullanılmaktadır.

Steinhart-Hart eşitlikleri (6-1)'de verilmiştir.

$$T = \frac{1}{A_0 + A_1(\ln R_T) + A_3(\ln R_T^3)} \quad (6-1)$$
$$\ln R_T = B_0 + \frac{B_1}{T} + \frac{B_3}{T^3}$$

T : Termistörün sıcaklığı (Kelvin)

$A_0, A_1, A_3, B_0, B_1, B_3$: Termistör üreticisinin sağladığı sabitler

R_T : T sıcaklığındaki termistör direnci

Tipik bir termistörde, (6-1)'de verilen üçüncü dereceden formül sıcaklığın $\pm 0.1^\circ\text{C}$ hassasiyetle ölçülmesini sağlamaktadır.

Her ne kadar termistörlerin kullanılabilirdiği sıcaklık aralığı diyot ve silikon bazlı sıcaklık algılayıcılarından ($-55^\circ\text{C} - 175^\circ\text{C}$) daha geniş olsa da, pratikte bu aralık $-100^\circ\text{C} - 175^\circ\text{C}$ 'dir. Bu değer RTD'ler ($-200^\circ\text{C} - 600^\circ\text{C}$) ve ısıçiftlere ($+1820^\circ\text{C}$ 'ye kadar) nazaran oldukça dar bir aralıkta kalmaktadır [33].

7. FPGA TABANLI ALGILAYICI DOĐRUSALLAŐTIRMA UYGULAMALARI

7.1. Giriő

Bu tez alıőmasında, PGA, YSA ve FPGA avantajları birleőtirilerek hata oranı dűőük ve iőlem hızı yüksek olabilen bir algılayıcı dođrusallaőtırma uygulaması gerekleőtirilmiőtir. Algılayıcı olarak NTC tipindeki sıcaklık algılayıcısı kullanılmıőtir. Ele alınan tűm yűntemlerin benzetimleri űncelikle yazılım ortamında gerekleőtirilmiőtir, ardından bu yűntemlerin tűmű FPGA deney setinde uygulanmıőtir. űnerilen PGA ve YSA birleőtiminden oluőtun hibrit uygulamanın sonuları, YSA, KPD, PGA+KPD yűntemlerinin kullanılmasıyla elde edilen sonularla karőtılaőtirilmiőtir.

Gerekleőtirilen benzetim ve donanım dođrusallaőtırma uygulamaları sırasıyla aőtađıda verilmiőtir.

- Benzetim uygulamaları
 - KPD yűntemiyle dođrusallaőtırma benzetim uygulaması
 - PGA+KPD yűntemiyle dođrusallaőtırma benzetim uygulaması
 - YSA yűntemiyle dođrusallaőtırma benzetim uygulaması
 - PGA+YSA yűntemiyle dođrusallaőtırma benzetim uygulaması

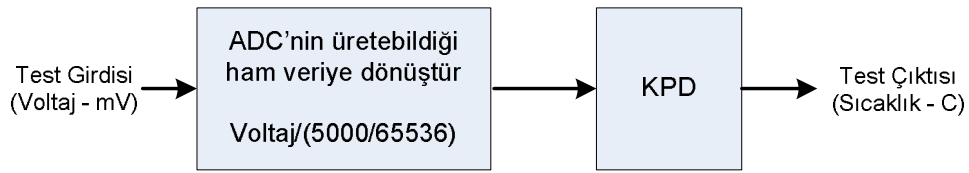
- Donanım uygulamaları
 - KPD yűntemiyle FPGA tabanlı dođrusallaőtırma donanım uygulaması
 - PGA+KPD yűntemiyle FPGA tabanlı dođrusallaőtırma donanım uygulaması
 - YSA yűntemiyle FPGA tabanlı dođrusallaőtırma donanım uygulaması
 - PGA+YSA yűntemiyle FPGA tabanlı dođrusallaőtırma donanım uygulaması

Benzetim alıőmaları MATLAB programında yapılmıőtir olup ardından bu yűntemler donanım űzerinde gereklenerek sonular ortalama karesel hata, hız ve bellek kullanımı yűnűnden karőtılaőtirilmiőtir.

7.2. Benzetim Çalışmaları

7.2.1. KPD yöntemi kullanılan doğrusallaştırma sisteminin benzetim uygulaması

Uygulamada öncelikle kısmi doğrusallaştırılan bölgelerin başlangıç ve bitiş noktalarını tutmak üzere başvuru çizelgesi oluşturulmuştur. Başvuru çizelgesi iki sütundan oluşmaktadır. İlk sütun algılayıcı gerilimi, ikinci sütunsa algılayıcı gerilimine karşılık gelen sıcaklık değerini içerir. Gerilim değeri başvuru çizelgesinde ADC'nin üretebildiği formata dönüştürerek saklanmıştır. Bu sayede ADC'den okunan ham veri doğrudan çizelgenin voltaj sütunundaki değerlerle karşılaştırılabilmektedir. KPD yöntemiyle benzetimi yapılan sistemin gösterimi Şekil 7.1'de verilmiştir.



Şekil 7.1 BÇ yöntemiyle benzetimi yapılan sistemin gösterimi

KPD yönteminde kullanılan başvuru çizelgesi Bölüm 7.3.1'de verilen algılayıcı benzetim devresi baz alınarak (7-5) ve (7-6) eşitlikleri kullanılarak, -55 °C ile 150 °C arasında yaklaşık 0,2 °C aralıklarla hesaplanan 1024 adet voltaja karşı sıcaklık değerlerinden oluşmaktadır. Voltaj değerleri 16 bitlik ADC'den okunacak sayısal değerlere (ham veri) çevrilerek kaydedilmiştir (Bkz. (7-3)). Kullanılan ADC -2.5V ile +2.5V arasında okuma yapabildiğinden okunabilen tam aralık 5000mV olmaktadır.

$$Voltaj (ADC Ham Verisi) = \frac{Voltaj (mV) * 2^{16}}{5000mV} \quad (7-1)$$

Başvuru çizelgesinde yer alan kayıtlardan örnekler Çizelge 7-1'de verilmiştir.

Çizelge 7-1 Başvuru çizelgesi kayıtlarından örnekler

Başvuru Çizelgesi Kayıtları (1024 adet)	
Voltaj (ADC ham verisi)	Sıcaklık (°C)
26292	-55,000
26268	-54,799
26244	-54,599
26220	-54,398
⋮	⋮
181	149,198
180	149,398
179	149,599
178	149,799
177	150,000

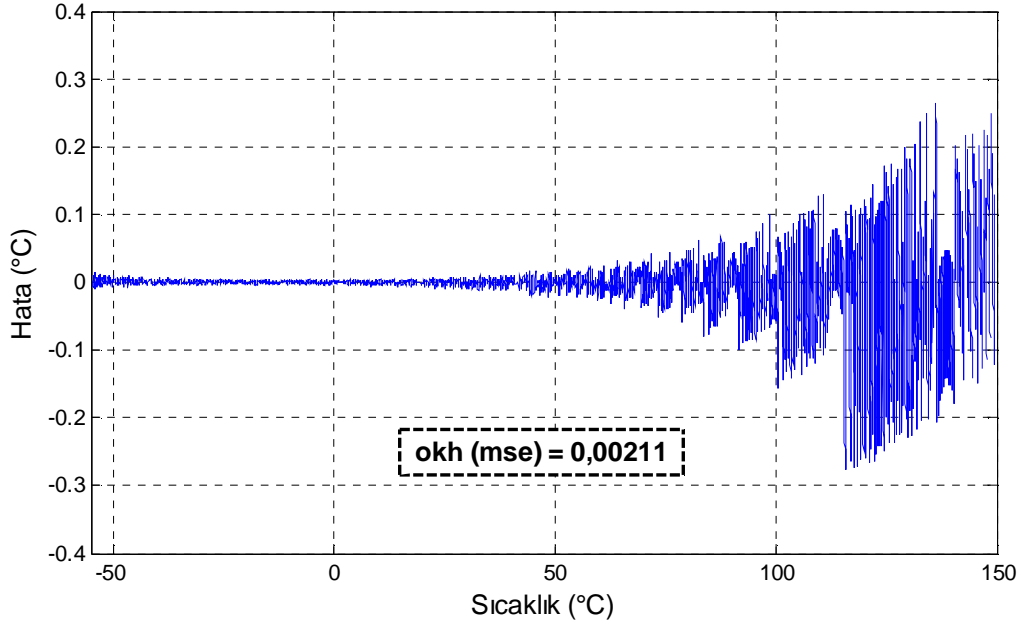
7.2.1.1. KPD yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi

1024 aralıktan oluşan KPD yöntemine sahip sistemin performansını test etmek amacıyla -55 °C ile 150 °C aralığında 0,01 °C'lik adımlarla 20501 adet veri, (7-5) ve (7-6) eşitlikleri kullanılarak hesaplanmış ve sisteme uygulanmıştır. Sistem çıktıları kaydedilerek beklenen sonuçlarla karşılaştırılmıştır.

Testin sonucunda ortalama karesel hata (7-4)'de verilen eşitlik kullanılarak 0,00211 olarak hesaplanmıştır. (7-4)'de verilen eşitlikte n toplam örnek sayısını, i örnek sırasını, e_i ise i 'inci örnek için hesaplanan hatayı ifade etmektedir.

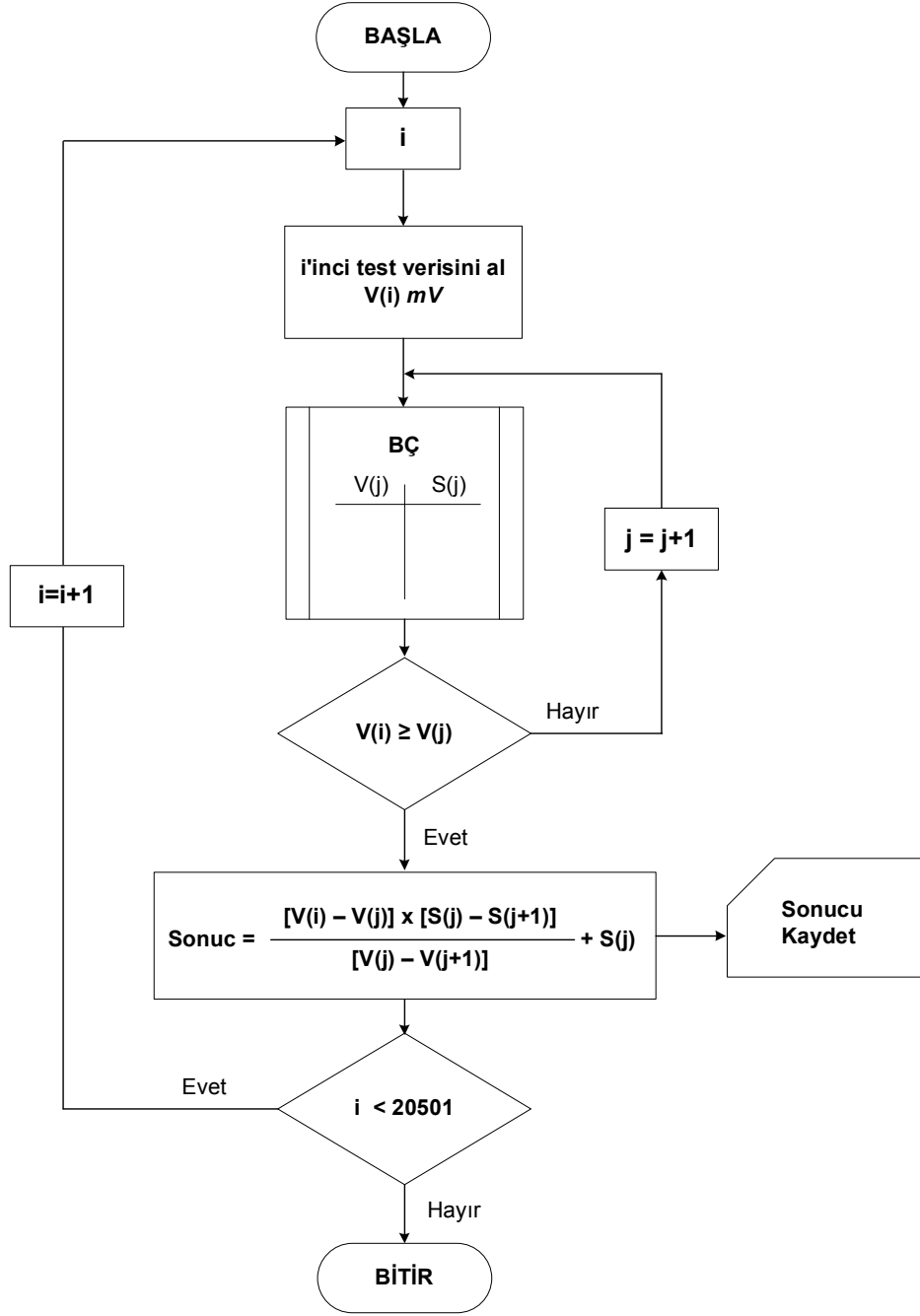
$$okh = \frac{1}{n} \sum_{i=1}^n e_i \quad (7-2)$$

Şekil 7.2'de KPD yöntemi kullanılan sistemin benzetim sonuçlarında görülen hatanın -55 °C – 150°C aralığındaki dağılımı verilmiştir.



Şekil 7.2 BÇ yöntemi kullanılan sistemin benzetim sonuçlarında hesaplanan hatanın -55 °C - +150 °C aralığındaki dağılımı

KPD yöntemi kullanılan sistemin benzetiminde gerçekleştirilen testin gösterimi Şekil 7.3'de verilmiştir.



Şekil 7.3 KPD yöntemi kullanılan doğrusallaştırma sisteminin benzetimi sırasında kullanılan test akış şeması

7.2.2. PGA+KPD yöntemi kullanılan doğrusallaştırma sisteminin benzetim uygulaması

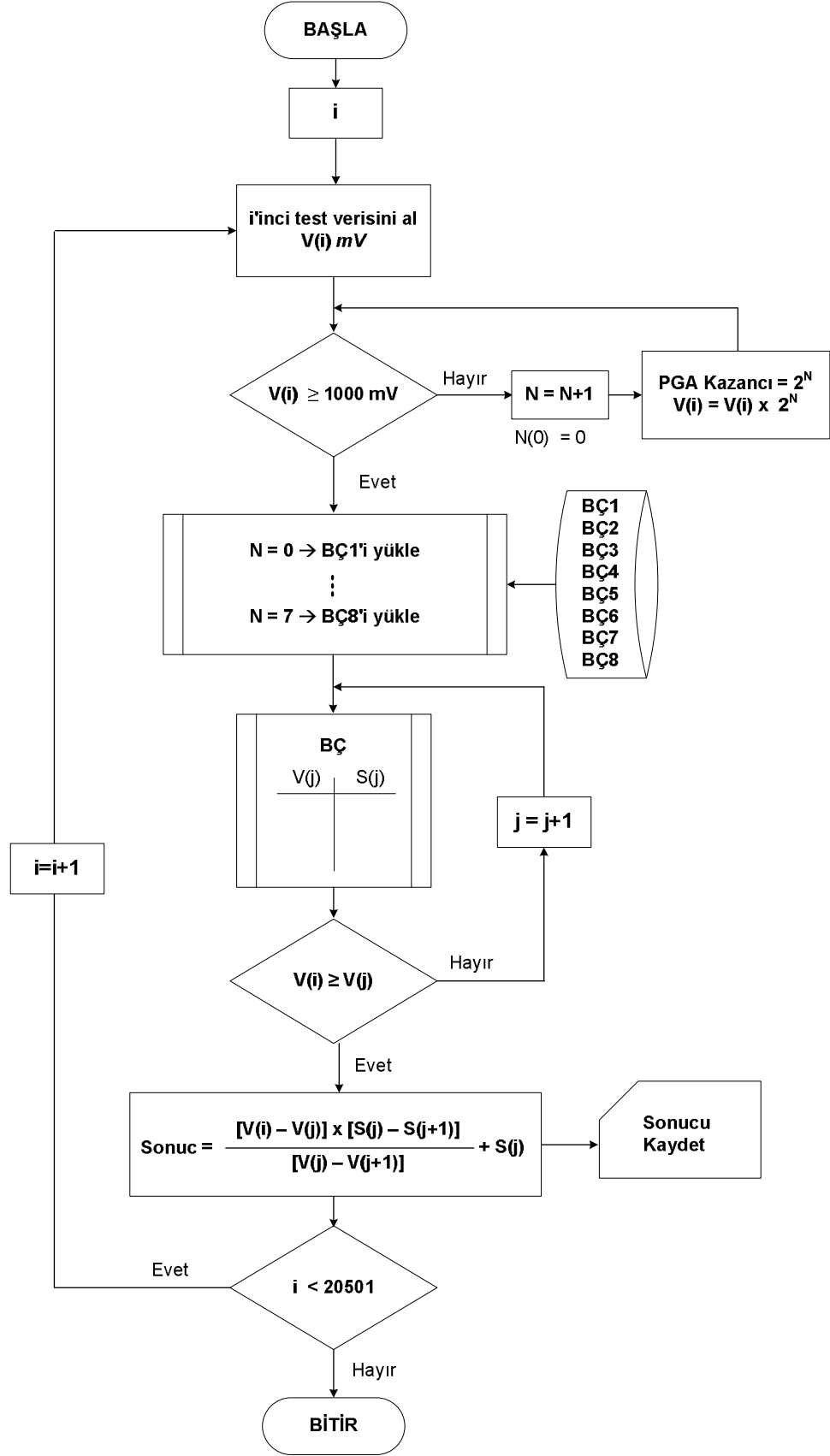
Bu uygulamada algılayıcı gerilimi 1000mV'un altına düştüğünde gerilim PGA ile yükseltilmiş ve yükseltilecek değer okunarak sıcaklığa dönüştürme işlemi gerçekleştirilmiştir. PGA'nın her bir kazanç aralığı için ayrı KPD çizelgesi oluşturulmuştur. PGA ile yükseltilmiş algılayıcı sinyali eğrisi grafiği Şekil 7.13'da verilmiştir. Bu grafiğe bağlı olarak 8 farklı bölge (K=1, 2, 4, 8, 16, 32, 64, 128) için ayrı ayrı başvuru çizelgeleri oluşturulmuştur. KPD yöntemi kullanılan sistemin 1024 kayıtlı BÇ'si alınmış ve her bir giriş 1000mV'un üzerine çıkana dek yükseltilmiş ve aynı kazançla yükseltilecek veriler bir araya getirilmiştir. Her bir küme ilgili kazanç değerinin BÇ'sini oluşturmuştur.

7.2.2.1. PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi

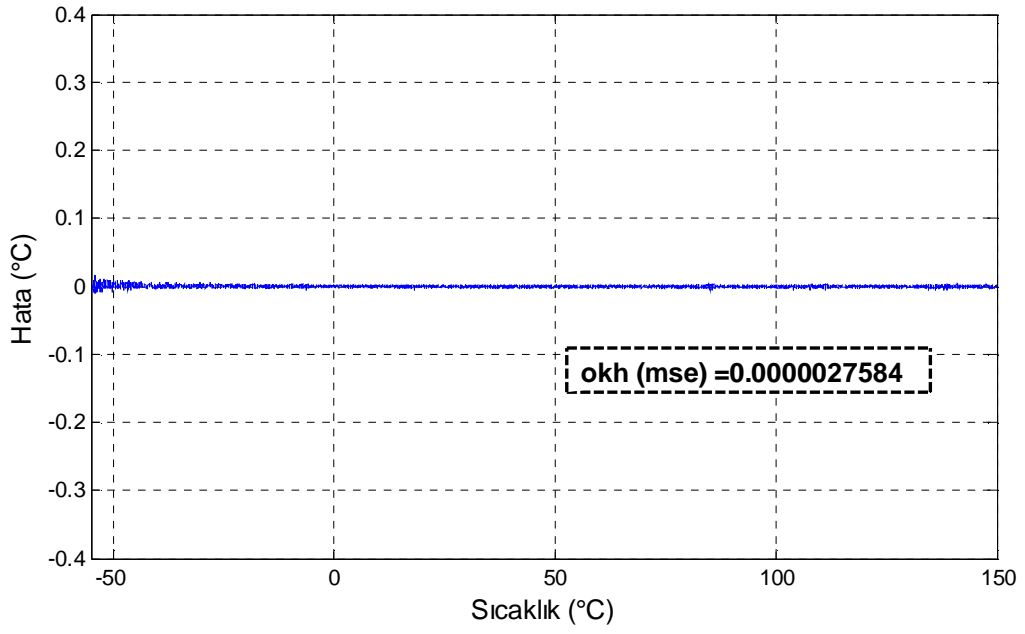
Toplamda 1024 kayıtlı başvuru çizelgelerinden oluşan KPD tabanlı sistemin performansını test etmek amacıyla sisteme -55 °C ile 150 °C aralığında 0,01 °C aralıkla 20501 adet girdi uygulanmış ve çıktılar kaydedilerek beklenen sonuçlarla karşılaştırılmıştır. Testin akış şeması Şekil 7.4'de verilmiştir.

Testin sonucunda ortalama karesel hata (okh,mse) (7-4)'de verilen eşitlik kullanılarak olarak 0.0000027584 olarak hesaplanmıştır.

Şekil 7.5'de PGA+KPD yöntemi kullanılan sistemin benzetim sonuçlarında görülen hatanın -55 °C – 150°C aralığındaki dağılımı verilmiştir.



Şekil 7.4 PGA + KPD yöntemi kullanılan sistemin benzetimi sırasında kullanılan test akış diyagramı



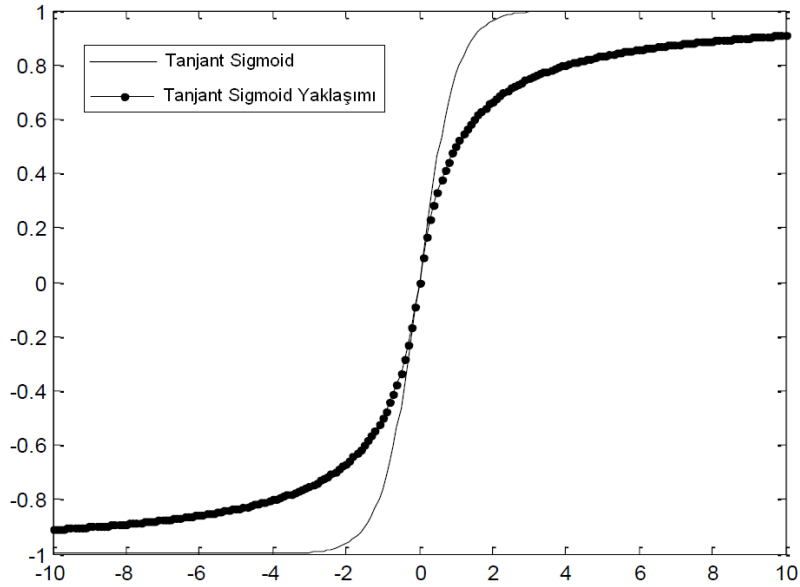
Şekil 7.5 PGA + KPD tabanlı sistemin benzetim sonucunda görülen hatanın -55 °C - +150°C aralığındaki dağılımı

7.2.3. YSA yöntemi kullanılan doğrusallaştırma sistemin benzetim uygulaması

Benzetim uygulamasında aktivasyon fonksiyonu olarak tanjant sigmoid fonksiyonunun yaklaşımı kullanılmıştır. Tanjant sigmoid fonksiyonu üstel ifade içerdiğinden (bkz. 3.3.1.5) bu fonksiyonun FPGA platformunda gerçekleşmesi maliyetli ve zaman alıcı olmaktadır. Bu sebeple, tez kapsamında gerçekleştirilen YSA tabanlı uygulamalarda (7-3)'de verilen tanjant sigmoid yaklaşımı ve (7-4)'de verilen tanjant sigmoid türevi yaklaşımı [34] kullanılmıştır. Orijinal tanjant sigmoid fonksiyonu ve kullanılan yaklaşım fonksiyonunun grafikleri Şekil 7.6'de verilmiştir.

$$tansigmoid(x) = \frac{x}{1 + |x|} \quad (7-3)$$

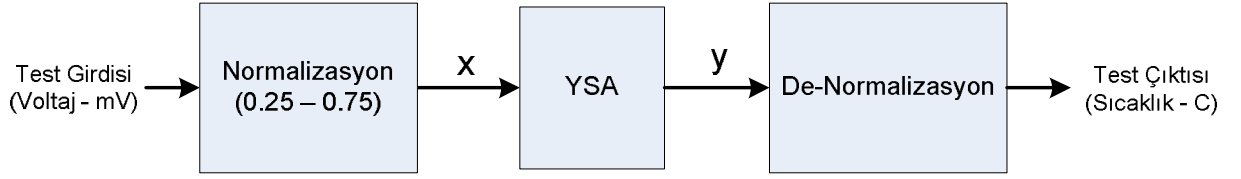
$$tansigmoid'(x) = \begin{cases} \frac{1}{1 - x^2}, & x < 0 \\ \frac{1}{1 + x^2}, & x \geq 0 \end{cases} \quad (7-4)$$



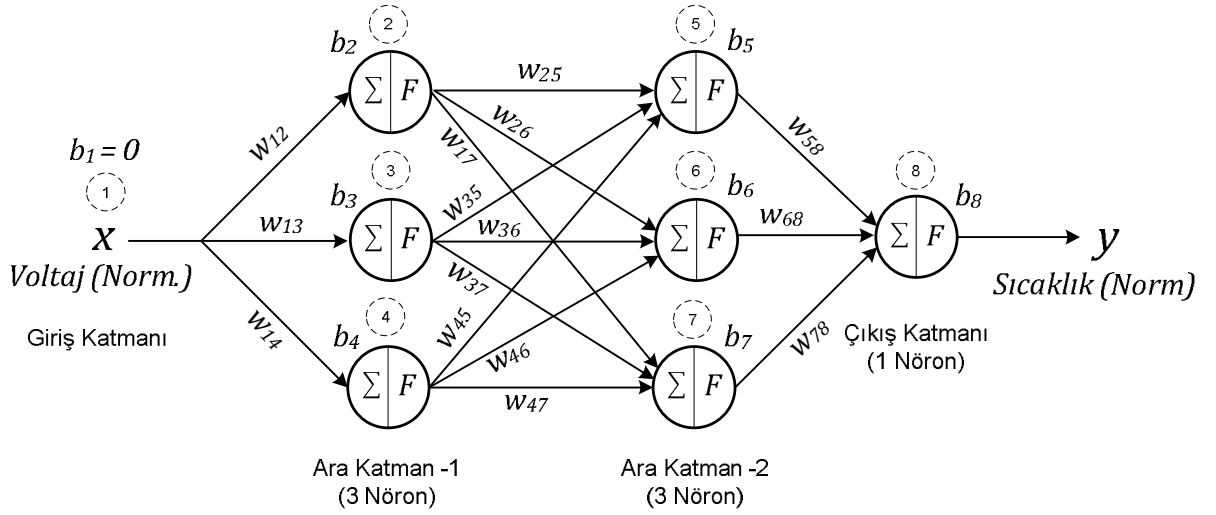
Şekil 7.6 Tanjant sigmoid fonksiyonu ve tanjant sigmoid fonksiyonunu yaklaşımının eğrileri (Çavuşlu, [34]'dan değiştirilerek)

Bu uygulamada YSA yöntemi kullanılan sistemin benzetimi yapılmıştır (Şekil 7.7). Kullanılan YSA, 3 er adet nörona sahip iki ara katmandan, bir adet giriş ve bir adet

de çıkış nöronundan oluşan Şekil 7.8'de verilen MLP tipindeki ağıdır. Burada x normalize edilmiş voltajı, y ise normalize edilmiş sıcaklığı ifade etmektedir.



Şekil 7.7 YSA tabanlı algılayıcı doğrusallaştırma sistemi



Şekil 7.8 YSA tabanlı sistemde kullanılan ağı yapısı.

7.2.3.1. YSA yöntemi kullanılan sistemdeki ağı eğitilmesi

Ağı eğitimi -55 °C – 150 °C aralığında 0,1 °C'lik adımlarla hesaplanan sıcaklığa karşı voltaj değerlerinden oluşan eğitim verisi kullanılmıştır. Eğitim verisine ait örnekler

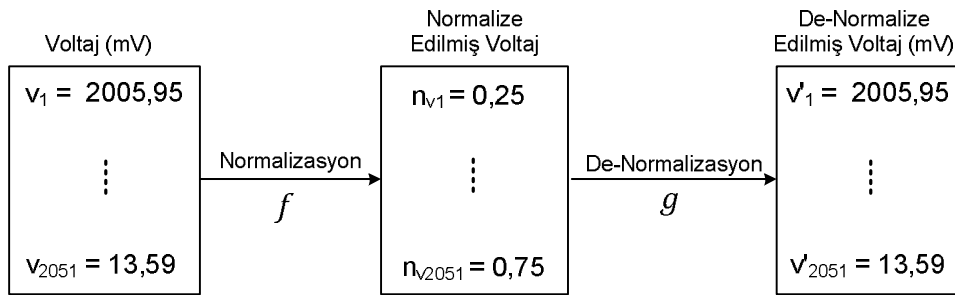
Çizelge 7-2'de verilmiştir.

Eğitim, Intel'in Core2Duo E4400 2,00 GHZ işlemcisine, 2,00 GB RAM'e ve Windows XP işletim sistemine sahip bir bilgisayarda gerçekleştirilmiştir. MATLAB 7.11.0 versiyonu kullanılmıştır.

Çizelge 7-2 Eğitim verisi örnekleri

Eğitim Verisi (2051 Adet)	
Voltaj (mV)	Sıcaklık (°C)
2005,95045011187	-55,0
2005,05325689794	-54,9
2004,15156244704	-54,8
2003,24535290362	-54,7
⋮	⋮
13,6836415242633	149,7
13,6549659943242	149,8
13,6263624224786	149,9
13,5978306016920	150

Test verileri YSA'ya iletilmeden önce normalizasyon işlemine tabi tutulmuştur. Normalizasyon, gerilim ve sıcaklık verilerinin birbirinden bağımsız olarak 0,25 – 0,75 değerleri aralığına haritalanmasıyla gerçekleştirilmiştir. Bu işlem sırasında hesaplanan ölçekleme ve kaydırma değerleri sonradan de-normalizasyon işlemine kullanılmak üzere saklanmıştır. Normalizasyon ve de-normalizasyon işlemlerine ait eşitlikler Şekil 7.9'de verilmiştir.



Normalizasyon

$$n_v = f(v) = [v - \min(v)] / [2 \times (\max(v) - \min(v))] + 0,25$$

De-Normalizasyon

$$v' = g(n_v) = [n_v - 0,25] \times [2 \times (\max(v) - \min(v))] + \min(v)$$

Şekil 7.9 Normalizasyon ve de-normalizasyon bağıntıları

Ağın eğitiminde öğrenme katsayısı 0,5 olarak seçilmiştir. Ağ, 10 milyon döngü boyunca beş kere eğitilmiştir. Her bir eğitimin sonunda oluşan ağırlık ve eşik değerleri ayrı ayrı kaydedilmiştir. Eğitimlerde elde edilen ortalama karesel hatalar ve eğitim süreleri Çizelge 7-3’de verilmiştir.

Çizelge 7-3 Beş farklı eğitimde elde edilen ortalama karesel hatalar (okh) ve eğitim süreleri

Eğitim No.	Ortalama Karesel Hata	Eğitim Süresi (sn)
1	0,0062657760	567,575298
2	0,0030058373	557,572125
3	0,0041130059	558,535854
4	0,0012949189	569,351530
5	0,0445557722	554,383091

Çizelge 7-3 incelendiğinde en iyi performansın Eğitim 4 ile sağlandığı görülmektedir.

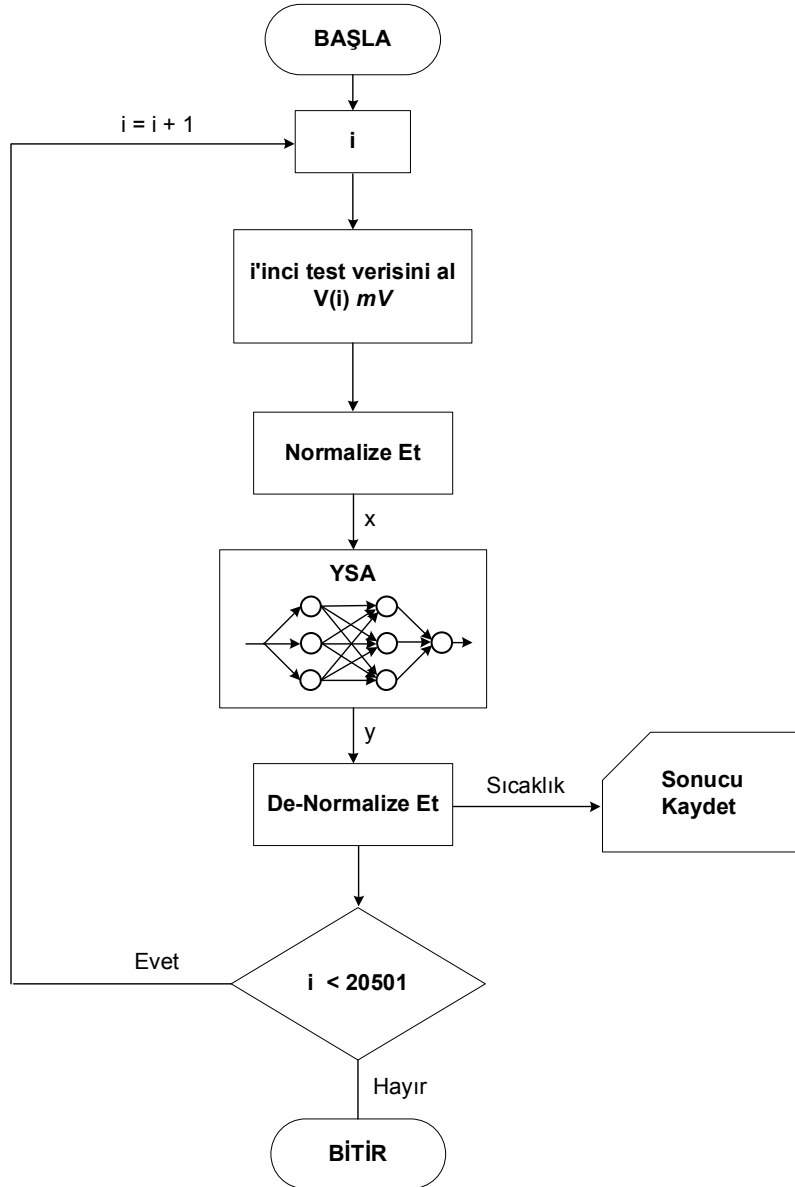
Eğitim 4’de elde edilen ağırlık ve eşik değerleri (bkz. Şekil 7.8) Çizelge 7-4’de verilmiştir.

Çizelge 7-4 Eğitim 4 sonucunda elde edilen ağırlık ve eşik değerleri

Ağırlıklar				Eşikler	
W₁₂	-9,7823852832	W₃₇	-4,8860873734	b₁	0
W₁₃	-6,0105460436	W₄₅	-0,3395812676	b₂	0,7841944226
W₁₄	75,1040807756	W₄₆	-13,7803418560	b₃	-8,5281785583
W₂₅	4,3043984832	W₄₇	1,9801751348	b₄	18,9014064309
W₂₆	2,4044028638	W₅₈	1,7853295893	b₅	-0,7233701822
W₂₇	0,8379407437	W₆₈	7,0769792455	b₆	2,3294454582
W₃₅	2,1488091174	W₇₈	-2,9239278507	b₇	-1,7589945956
W₃₆	-0,5596725835			b₈	-6,7467390304

7.2.3.2. YSA yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi

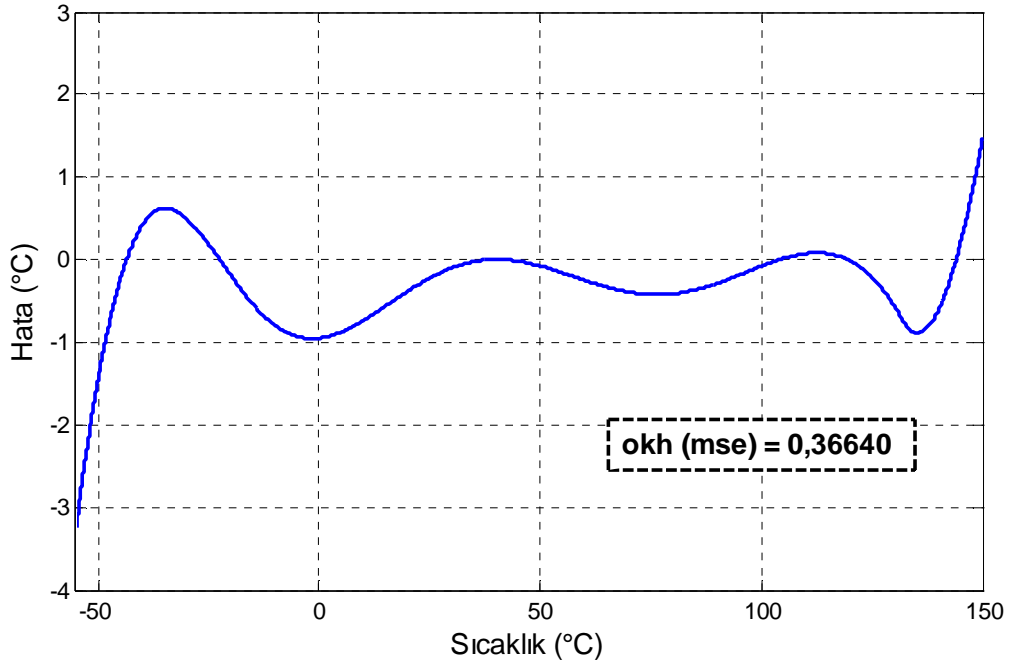
Eğitilen ağı test etmek üzere -55 °C'den başlayarak 0,01 °C adımlarla 150 °C'ye kadar olan sıcaklık aralığında her bir sıcaklık değerine karşılık gelen transdüser voltajı hesaplanmış ve ağa girdi olarak sağlanmıştır. Toplamda 20501 adet test verisi ile test gerçekleştirilmiştir. Ağın test edilmesi amacıyla uygulanan işlemlerin akış diyagramı Şekil 7.10'de verilmiştir.



Şekil 7.10 YSA yöntemi kullanılan doğrusallaştırma sisteminin benzetimi sırasında kullanılan test akış diyagramı

YSA çıkışları de-normalizasyon işleminden geçirildikten sonra kaydedilmiştir. Ardından kaydedilen değerler beklenen değerlerle karşılaştırılmıştır. Bu karşılaştırma sonucunda elde edilen sonuçların ortalama karesel hatası (okh, mse) 0,36640 olarak bulunmuştur.

Şekil 7.11’de YSA tabanlı sistemin benzetim sonucunda görülen hatanın -55 °C – 150°C aralığındaki dağılım grafiği verilmiştir.

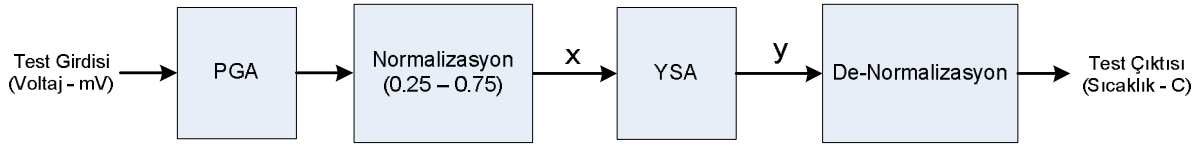


Şekil 7.11 YSA yöntemi kullanılan sistemin benzetim sonucunda görülen hatanın -55 °C – 150°C aralığındaki dağılımı

7.2.4. PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin benzetim uygulaması

Uygulamada kullanılan NTC tipi algılayıcının doğrusal olmayan voltaj-sıcaklık ilişkisine bakıldığında voltaj aralığının geniş bir aralıkta değiştiği; fakat yüksek sıcaklıklara çıkıldığında sıcaklığın dar bir voltaj aralığına sıkıştığı görülmektedir. Bu durum, voltajın sıcaklık bilgisine dönüştürülmesi sırasında özellikle bu dar alanda kalan yüksek sıcaklık değerlerinde ölçüm hatalarının artmasına yol açmaktadır. Bunun önüne geçmek için transdüser voltajı belirli bir eşik değerinin altına düştüğünde voltajın yükseltilmesi, okuma ve sıcaklığa dönüştürme işlemlerindeki hataların azalmasını sağlamaktadır.

Bu uygulamada giriş voltajının 1000mV'luk eşik değerinin altına düşmesi durumunda voltajın PGA ile yükseltilmesi ve ardından yükseltilerek okunmuş verinin kazançla ait bağımsız YSA ile değerlendirilmesinin benzetimi yapılmıştır (Şekil 7.12).

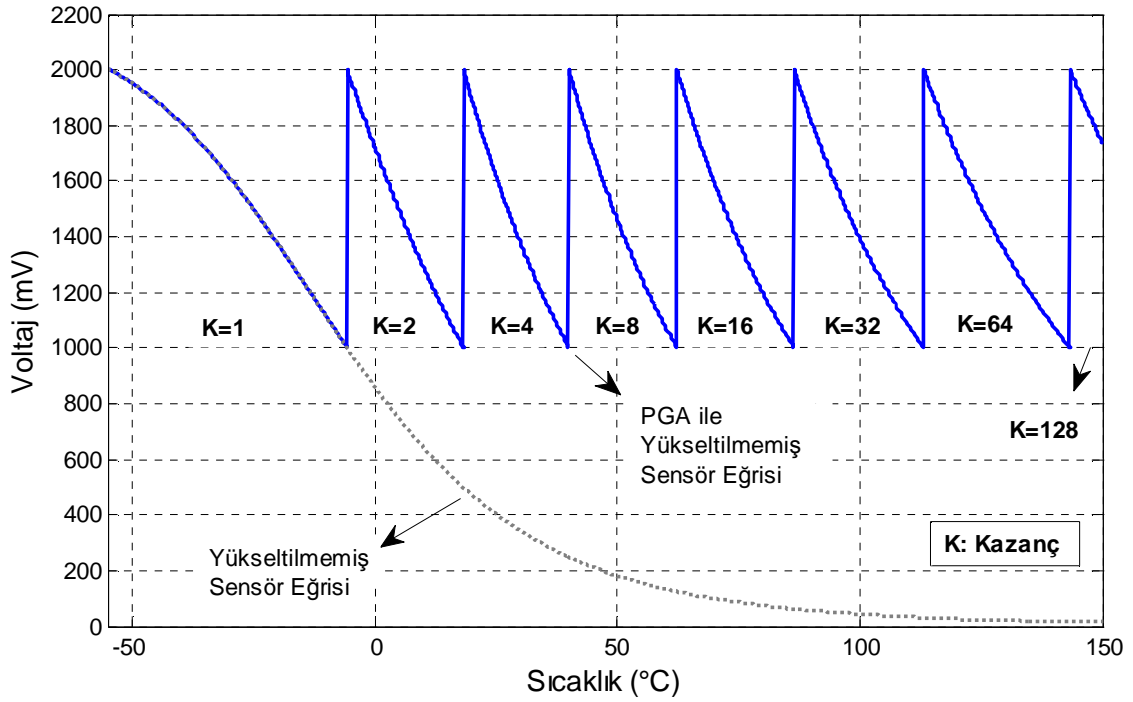


Şekil 7.12 Benzetimi yapılan PGA + YSA yöntemi kullanılan doğrusallaştırma sistemi

Transdüserin voltaj-sıcaklık eğrisi ile PGA ile yükseltilmiş voltaj-sıcaklık eğrisi Şekil 7.13'da verilmiştir. Yükseltme sırasında 1, 2, 5, 8, 16, 32, 64 ve 128 kazançlarıyla yükseltilmiş 8 adet bölge oluşmaktadır.

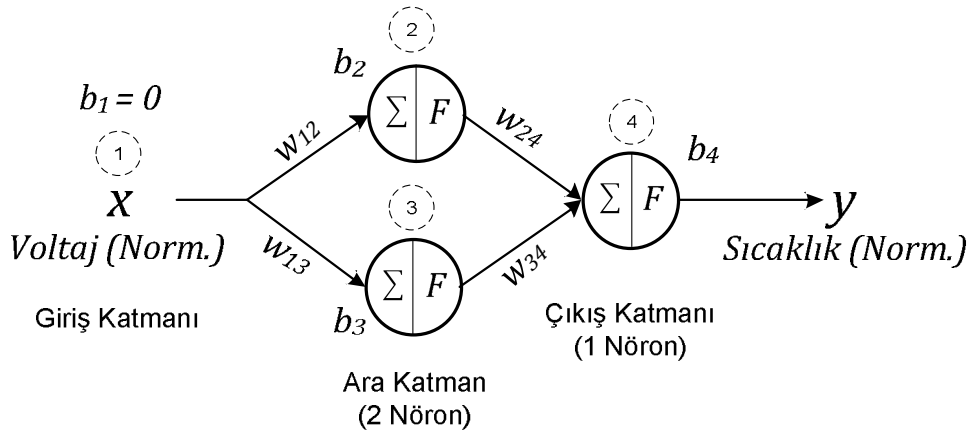
7.2.4.1. PGA + YSA yöntemi kullanılan sistemdeki ağın eğitilmesi

Ağın eğitimi sırasında kullanılan eğitim verisi -55 °C ile +150 °C aralığında 0,1°C adımlarla hesaplanmış sıcaklığa karşı voltaj değerlerinden oluşmaktadır. Uygulamada 8 farklı kazanç bölgesi oluşmuştur. Kazancın 1 olduğu bölgeye düşen veri sayısının fazla olması sebebiyle bu bölge için iki adet YSA eğitilmiştir. Toplamda 9 adet YSA eğitilmiştir. Eğitim aralıklarının daralması, ilgili aralıklardaki fonksiyon kestirimini kolaylaştırmış ve bu sayede basitleştirilmiş bir ağ yapısının kullanımı yeterli olmuştur.



Şekil 7.13 Transdüserin voltaj-sıcaklık eğrisi ile 8 bölgede PGA ile yükseltilmiş voltaj-sıcaklık eğrisi

Bu uygulamada kullanılan YSA mümkün olduğunca basitleştirilmiş ve Şekil 7.14'de verilen bir ara katmanlı MLP yapısı kullanılmıştır. Ara katmanda iki adet nöron bulunmaktadır.



Şekil 7.14 PGA'lı YSA uygulamasında kullanılan MLP yapısı

Her bölgeye ait YSA ayrı ayrı 1'er milyon döngü boyunca 10'ar kere eğitilmiş ve her bir aralıkta elde edilen sonuçlardan en iyi performans gösterenler seçilmiştir. Seçilen ağlara ait ortalama karesel hata ve eğitim süreleri

Çizelge 7-5'de, ağırlık ve eşik değerleri Çizelge 7-6'da verilmiştir.

Çizelge 7-5 Seçilen eğitime ait ortalama karesel hata ve eğitim süreleri

Kazanç/YSA No.	Ortalama Karesel Hata	Eğitim Süresi (sn)
K=1 / YSA 1	7,2442132277e-005	25,734237
K=1 / YSA 2	1,8829210836e-005	25,119289
K=2 / YSA 3	3,7592291344e-005	25,219321
K=4 / YSA 4	2,1132093820e-005	24,982738
K=8 / YSA 5	3,5348933367e-005	25,738549
K=16 / YSA 6	2,3737079576e-005	25,654334
K=32 / YSA 7	2,2752584918e-005	25,109001
K=64 / YSA 8	2,8124314748e-005	25,143334
K=128 / YSA 9	3,7826807113e-006	25,433356

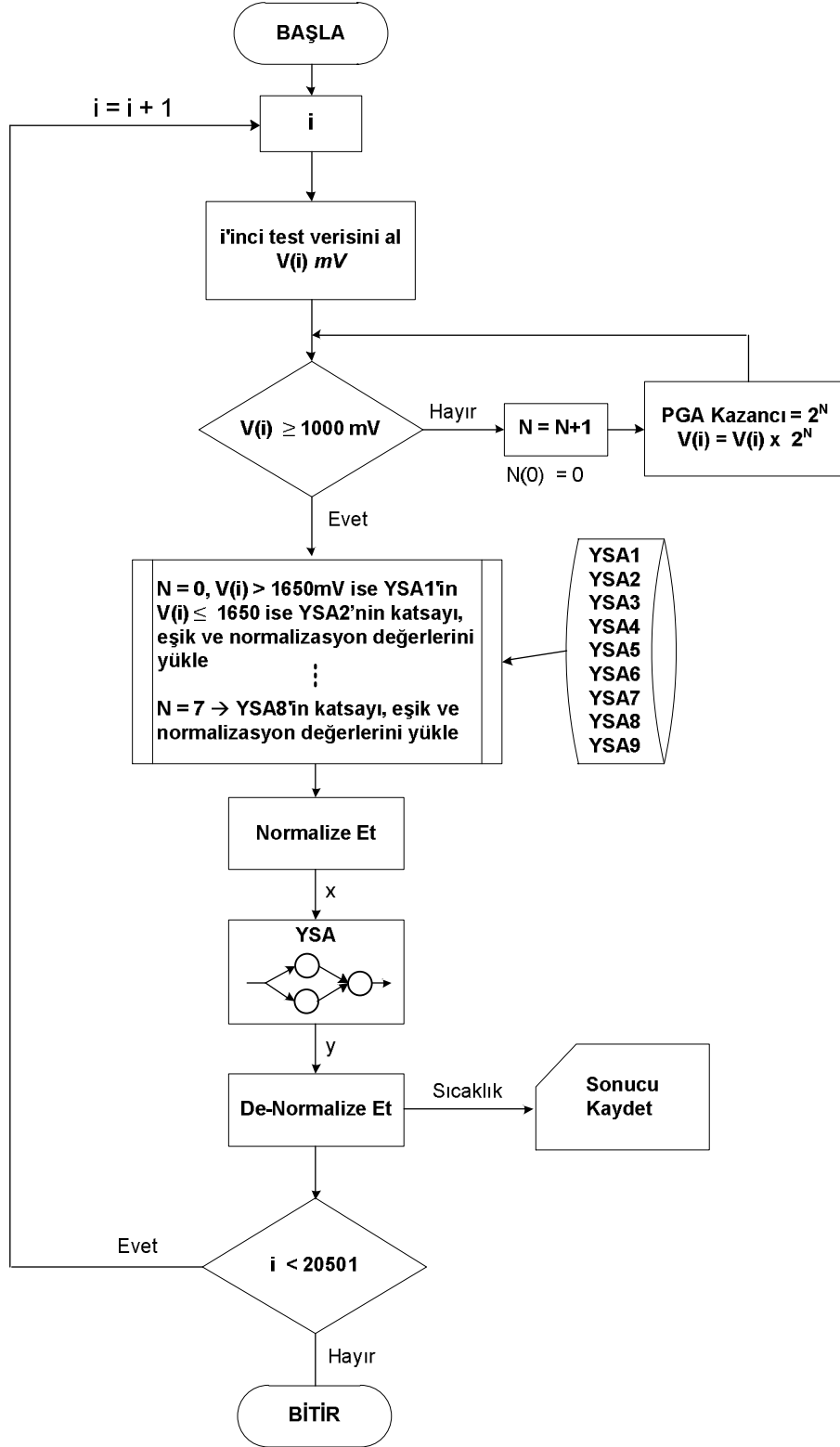
Çizelge 7-6 Eğitim sonunda elde edilen ağırlık ve eşik değerleri

	YSA1	YSA2	YSA3	YSA4	YSA5	YSA6	YSA7	YSA8	YSA9
w₁₂	-1,4792	4,0160	-5,4777	4,6991	-2,4472	-2,6460	5,7963	5,7462	-2,6083
w₁₃	-2,9209	-0,7616	-3,2154	-1,7232	5,9089	5,4023	-2,5415	-2,5108	5,4078
w₂₄	0,8520	-3,0969	6,9304	-5,9799	-1,7979	-2,0673	-6,2668	-6,0979	-1,7019
w₃₄	3,2129	0,9133	-2,0827	-1,6261	-6,5491	-6,3150	-1,7437	-1,6271	-6,8954
b₁	-0,9538	1,1333	-1,2050	1,0398	-0,7791	-0,8383	1,2902	1,2897	-0,1985
b₂	-0,8555	-0,6426	-0,9469	-0,6672	1,2672	1,2516	-0,8372	-0,8550	0,9348
b₃	-2,2932	-2,2927	-4,2549	-3,9737	-4,3903	-3,8841	-4,1555	-4,0901	-4,4720

7.2.4.2. PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin test benzetimi

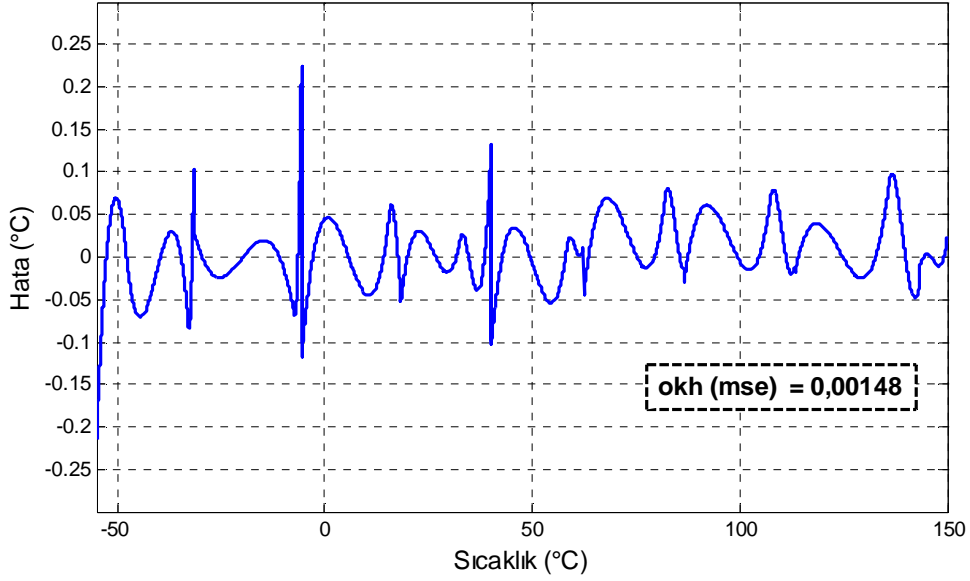
Eğitilen ağı test etmek üzere -55 °C'den başlayarak 0,01 °C adımlarla 150 °C'ye kadar olan sıcaklık aralığında her bir sıcaklık değerine karşılık gelen transdüser voltajı hesaplanmış ve ağa girdi olarak sağlanmıştır. Toplamda 20501 adet test verisi ile test gerçekleştirilmiştir. Ağın test edilmesi amacıyla uygulanan işlemlerin akış diyagramı Şekil 7.15'de verilmiştir.

YSA çıkışları de-normalizasyon işleminden geçirildikten sonra kaydedilmiştir. Ardından kaydedilen değerler beklenen değerlerle karşılaştırılmıştır. Bu karşılaştırma sonucunda elde edilen sonuçların ortalama karesel hatası (okh, mse) 0,00148 olarak hesaplanmıştır.



Şekil 7.15 PGA + YSA yöntemi kullanılan sistemin benzetimi sırasında kullanılan test akış diyagramı

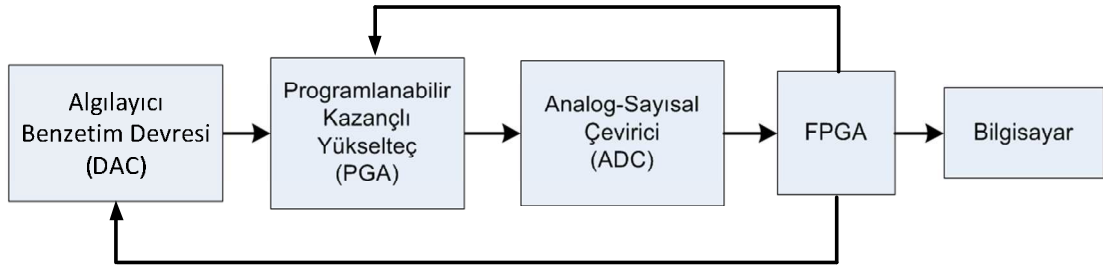
Şekil 7.16'de PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin benzetim sonuçlarında görülen hatanın -55 °C – 150°C aralığındaki dağılımı verilmiştir.



Şekil 7.16 PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin sonuçlarında görülen hatanın -55 °C – 150°C aralığındaki dağılımı

7.3. Donanım Çalışmaları

Şekil 7.17'de verilen sistem donanımsal olarak gerçekleştirilmiş ve 7.2'de benzetimleri yapılan yöntemler donanım üzerinde çalıştırılmıştır.

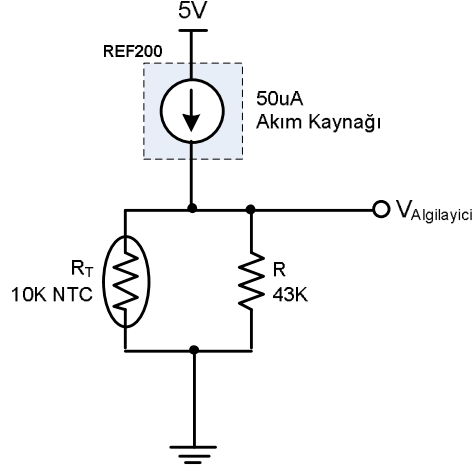


Şekil 7.17 Donanımda gerçekleştirilen algılayıcı doğrusallaştırma uygulaması akış şeması

7.3.1. Algılayıcı benzetim devresi

Uygulamada Vishay firmasının 10K Ω 'luk NTC termistörü (NTCS0805E3103FMT) kullanılmıştır. NTC direncini voltaja dönüştürmek için NTC'ye seri bağlı 50uA'lik sabit akım kaynağından (REF200) yararlanılmıştır. Sıcaklık ölçümünün -55°C –

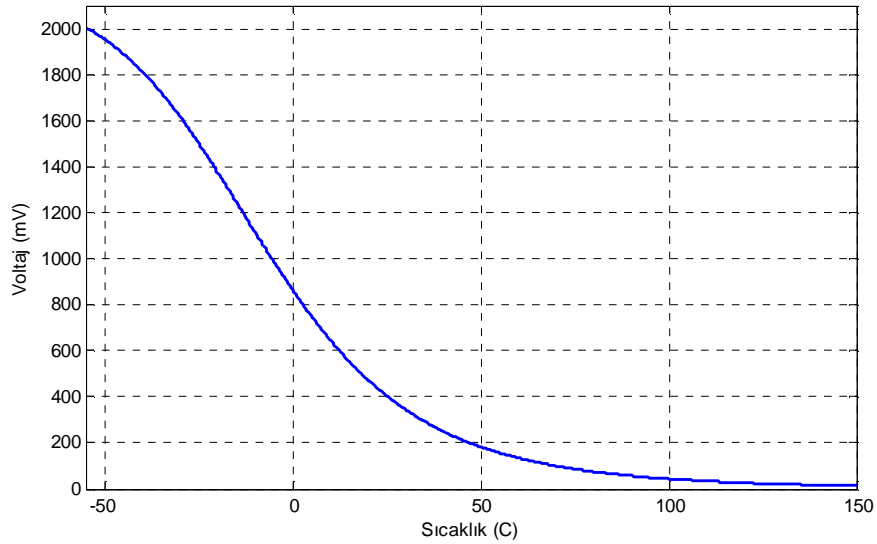
150°C aralığında yapılacağına karar verilmiş ve algılayıcı voltajını bu aralıktta 0-2V aralığına sınırlamak için NTC'ye paralel olarak 43KΩ'luk bir direnç bağlanmıştır (Şekil 7.18).



Şekil 7.18 -55°C - +150°C sıcaklık aralığında algılayıcı voltajını ($V_{Algilayici}$) 2V – 0V aralığına haritalayan devre

$$V_{Sensör} = 50\mu A \times \frac{R_T R}{R_T + R} \quad (7-5)$$

Sıcaklık ile $V_{Algilayici}$ arasındaki ilişki (7-5)'de verilmiştir. $R = 43K$ için sıcaklık - $V_{Algilayici}$ eğrisi Şekil 7.19'de verilmiştir.



Şekil 7.19 $R = 43K$ için Sıcaklık - $V_{Algilayici}$ eğrisi

Algılayıcı doğrusallaştırma uygulamasının performansını test etmek üzere sisteme bir takım sonucu bilinen girişler uygulamak; sistemden alınan çıkışları da beklenen bu sonuçlarla karşılaştırmak gerekmektedir. Kullanılan algılayıcının sıcaklık algılayıcısı olması itibariyle test esnasında gerekli girdileri oluşturmak üzere NTC sıcaklığını değiştirmek $-55\text{ }^{\circ}\text{C} - 150\text{ }^{\circ}\text{C}$ gibi geniş bir aralıkta mümkün olamamaktadır. Bu sebeple Şekil 7.18'de verilen devrenin benzetimi sayısal-analog çevirici (DAC) entegre vasıtasıyla yapılmıştır. Kullanılan DAC entegresi Analog Devices firmasının AD5752 model sayısal-analog çeviricisidir. Bu entegre $0 - 5\text{V}$ aralığında 16 bit çözünürlükte, yani yaklaşık $76.3\text{ }\mu\text{V}$ hassasiyette voltaj çıktısı üretebilmektedir. Bu sayede algılayıcı devresinin istenen sıcaklıkta hangi voltajı üretmesi gerektiği hesaplanarak DAC çıkışı bu değere ayarlanmakta ve sistemin geri kalanı DAC'ın ürettiği bu voltajı algılayıcı verisi gibi işleyebilmektedir.

Algılayıcı voltajını belirlemek için öncelikle NTC'nin ilgili sıcaklıktaki direncinin hesaplanması gerekmektedir. Bu işlem için gerekli eşitlik üretici firma tarafından sağlanmıştır (bkz. (7-6)).

$$R_T = R_{25} \exp \left(A + \frac{B}{T} + \frac{C}{T^2} + \frac{D}{T^3} \right) \quad (7-6)$$

İlgili NTC için üreticinin sağladığı sabitler şunlardır;

$$R_T = 10000$$

$$A = -13,4088568$$

$$B = 4547,9615$$

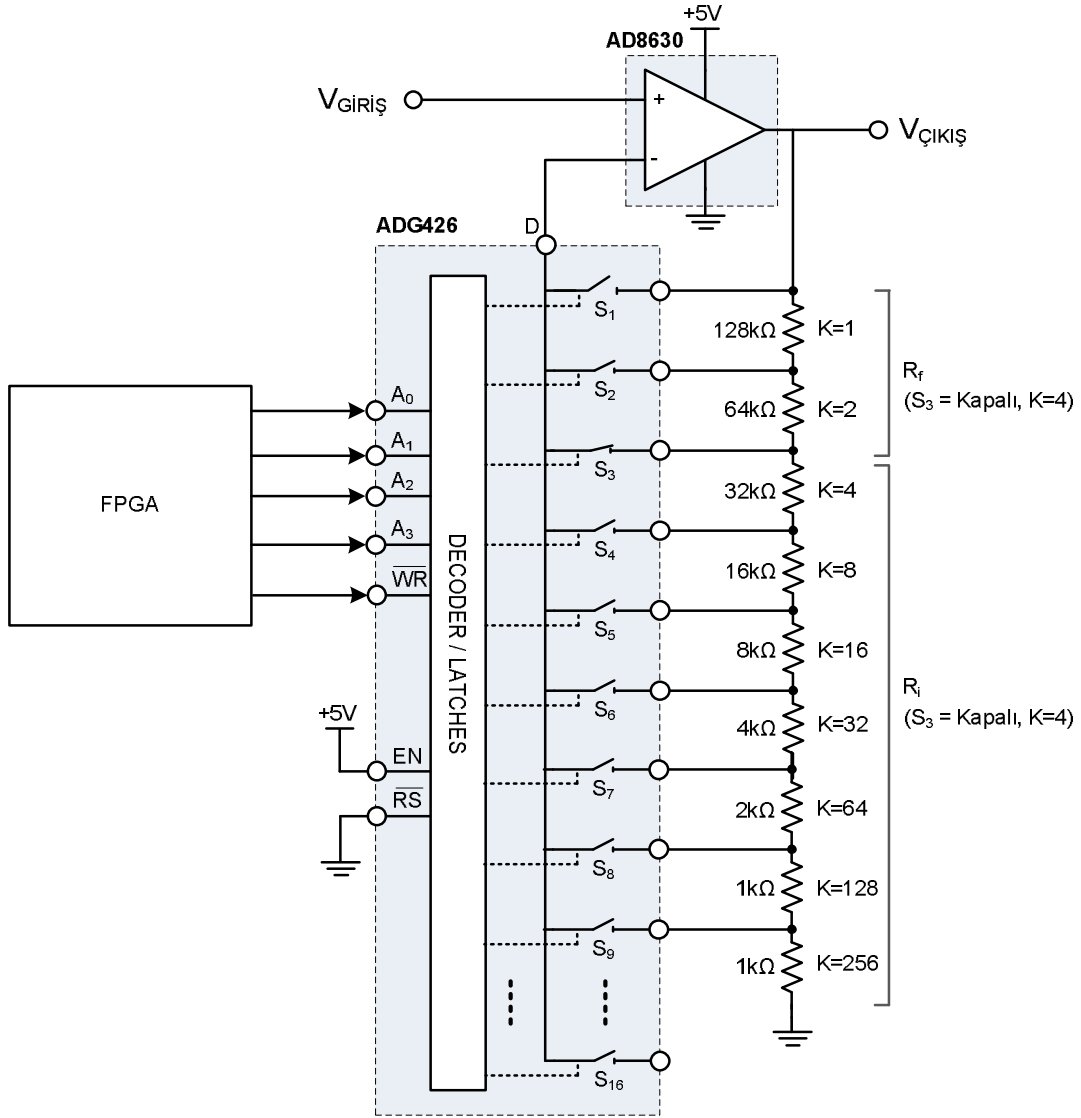
$$C = -176965,92$$

$$D = 3861154$$

(7-6)'de verilen denklem yardımıyla NTC'nin istenen sıcaklıktaki direnç değeri bulunduktan sonra devrenin voltaj çıktısı, yani DAC tarafından üretilmesi gereken voltaj, (7-5)'de verilen formülle hesaplanmıştır.

7.3.2. Kullanılan PGA'nın özellikleri

Uygulamada, Şekil 7.20'de verilen PGA tasarlanmış ve kullanılmıştır. PGA'nın kazançları 1, 2, 4, 8, 16, 32, 64, 128 ve 256 olarak ayarlanabilmektedir. Anahtarlama elemanı olarak ADG426 analog yol seçici (İng. analog multiplexer) kullanılmıştır. ADG426 entegresi, bir adet analog girişi, 4 bitlik binary adresleme girişleri (A_0 , A_1 , A_2 ve A_3) ile kontrol edilebilen 16 adet çıkıştan birine bağlayabilmektedir. Bu entegrenin R_{ON} direnci $+25^\circ\text{C}$ 'de yaklaşık olarak 90Ω 'dur. Ancak R_{ON} direnci op-amp'ın yüksek empedanslı eviren girişine seri bağlandığından PGA kazancını etkilememektedir. Yükselteç olarak AD8630 op-amp'ı kullanılmıştır.



Şekil 7.20 Uygulamada kullanılan PGA'nın şematığı

AD8630'un adresleme ve WR' girişleri FPGA ile kontrol edilmiştir. PGA kazancı (7-7)'de verilen formül ile hesaplanır.

$$\frac{V_{\text{ÇIKIŞ}}}{V_{\text{GİRİŞ}}} = \frac{R_f + R_i}{R_i} \quad (7-7)$$

S₃ anahtarı kapatıldığında (bkz.Şekil 7.20), R_f = 128kΩ + 64kΩ = 192 kΩ , R_i = 32kΩ + 16kΩ + 8kΩ + 4kΩ + 2kΩ + 1kΩ + 1kΩ = 64kΩ olmakta ve bu durumda kazanç,

$$K = \frac{192k\Omega + 64k\Omega}{64k\Omega} = 4 \quad (7-8)$$

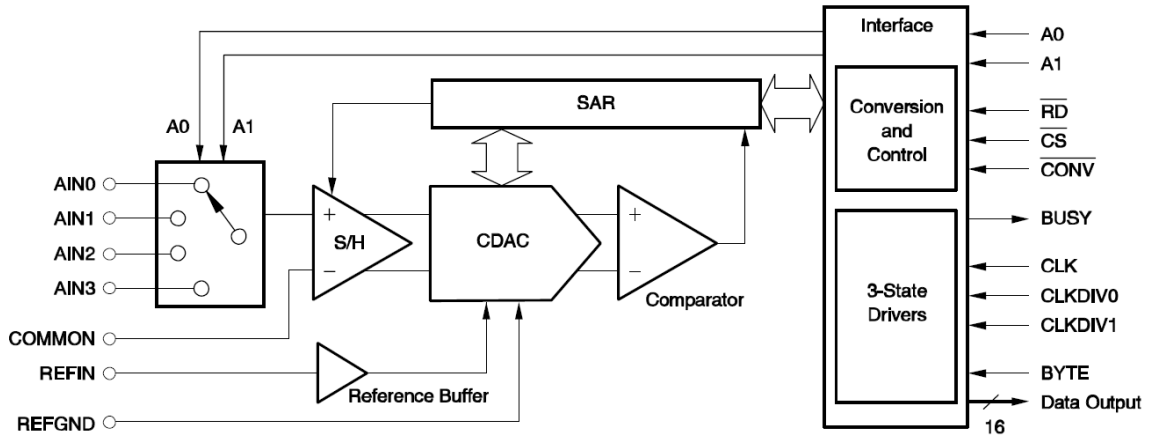
olarak hesaplanmaktadır. S₁ anahtarı kapatıldığında K=1 , S₂ anahtarı kapatıldığında K=2, S₄ anahtarı kapatıldığında K=8, S₅ anahtarı kapatıldığında K=16 , S₆ anahtarı kapatıldığında K=32, S₇ anahtarı kapatıldığında K=64, S₈ anahtarı kapatıldığında K=128 ve son olarak S₉ anahtarı kapatıldığında iseK=256 olarak hesaplanır.

Bu yapıda direnç değerleri ihtiyaca göre değiştirilerek istenilen kazançlar elde edilebilir. Bu da PGA'nın esnekliğini arttırarak özelleşmiş uygulamalara adaptasyonu kolaylaştırmaktadır.

Piyasada çeşitli firmaların ürettikleri hazır PGA entegreleri bulunmaktadır. Ancak bu entegrelerin pek çoğu adresleme girdilerini SPI, I²C gibi seri haberleşme protokolleri üzerinden almaktadırlar. Yapılan çalışmada hız önemli bir kriter olduğundan seri iletişim kuran yavaş PGA'ların yerine paralel ve yaklaşık 220ns'de anahtarlama yapabilen bu yapı kullanılmıştır.

7.3.3. Kullanılan ADC'nin özellikleri

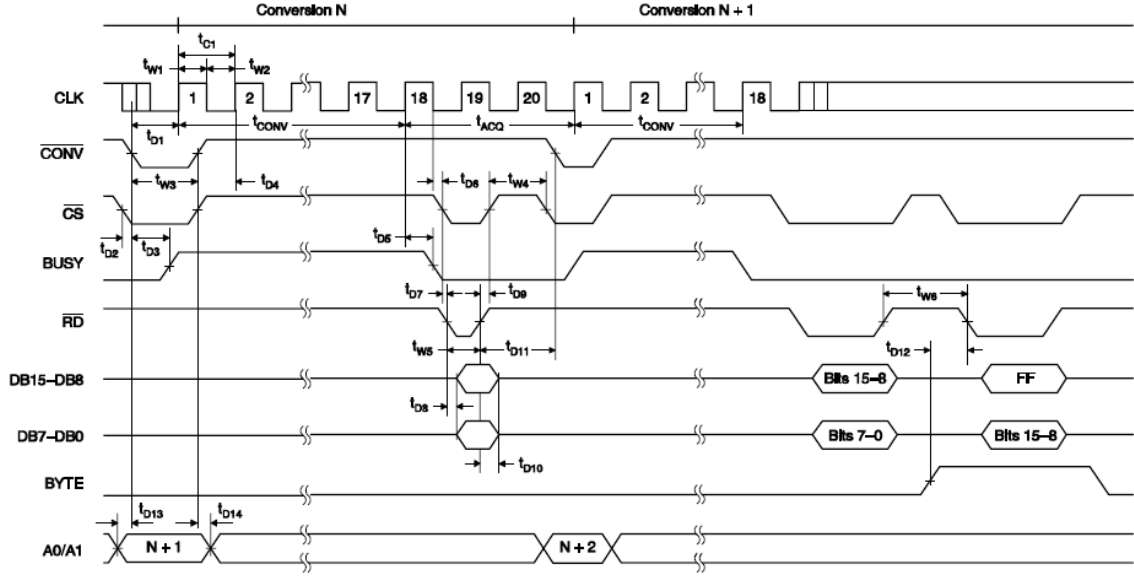
PGA'da yükseltileen sinyalin sayısallaştırılması amacıyla Texas Instruments firmasının paralel çıkış veren ADS8342 entegresi kullanılmıştır (Şekil 7.21). Bu entegre $\pm 5V$ besleme voltajıyla çalışıp $\pm 2.5V$ aralığını 8 bit ya da 16 bit çözünürlükte sayısallaştırabilmektedir. Entegre saniyede 250.000 örnekleme yapabilme kabiliyetine sahiptir.



Şekil 7.21 ADS8342 ADC entegresinin blok diyagramı

Genellikle sayısal verileri SPI, I²C gibi seri protokollerle ileten ADC entegrelerinin aksine ADS8342 entegresi çıktısını paralel olarak dış dünyaya iletmektedir. ADC'nin çevrim işlemi tamamlandığında 8 ya da 16 adet pini kullanılarak sonuç doğrudan okunmaktadır.

Entegrenin ihtiyacı olan saat işareti FPGA tarafından sağlanmıştır. Bunun yanısıra entegrenin çevrime başlama (CONV), meşgul (BUSY) ve oku (RD) sinyalleri de Şekil 7.22'de verilen zamanlama diyagramına uygun olarak FPGA tarafından yönetilmiştir.

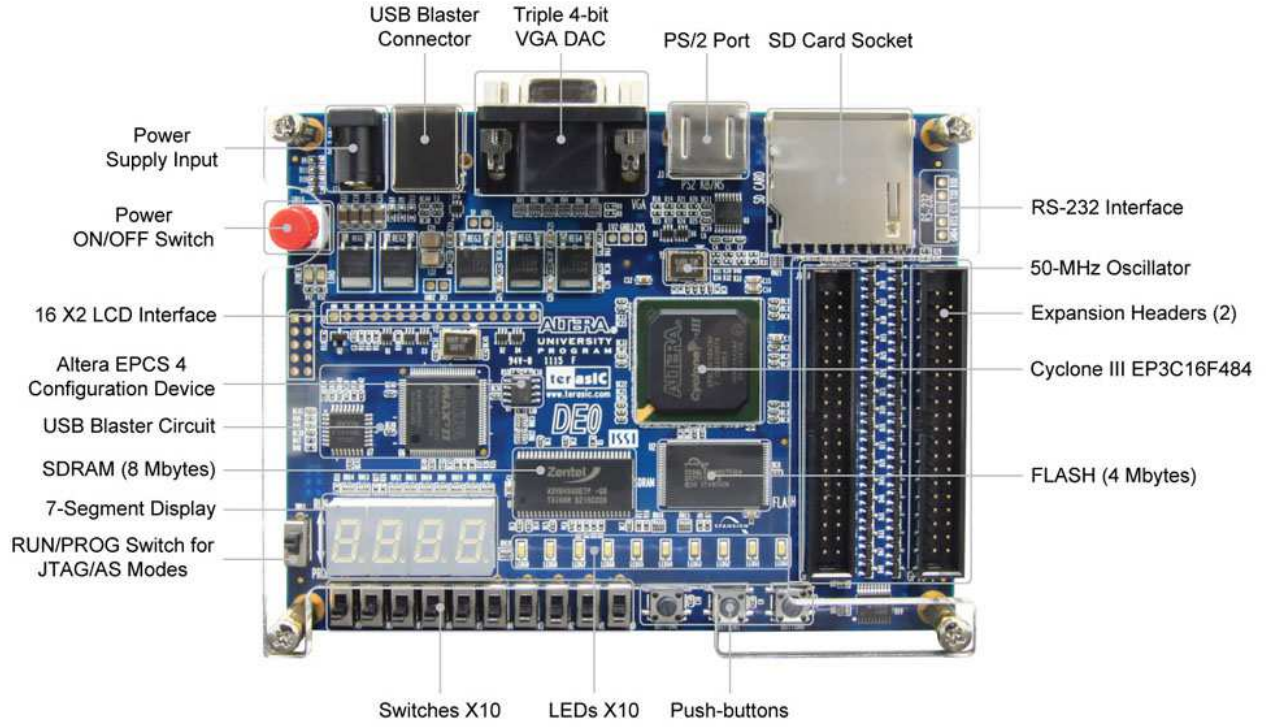


Şekil 7.22 ADS8342 zamanlama diyagramı

7.3.4. Kullanılan FPGA platformunun özellikleri

Bu tezde gerçekleştirilen uygulamada, Terasic firmasının ürettiği Altera Firmasının Cyclone III ailesinden EP3C16F484 model FPGA entegresinin kullanıldığı DE0 geliştirme kartı (Şekil 7.23) kullanılmıştır [32].

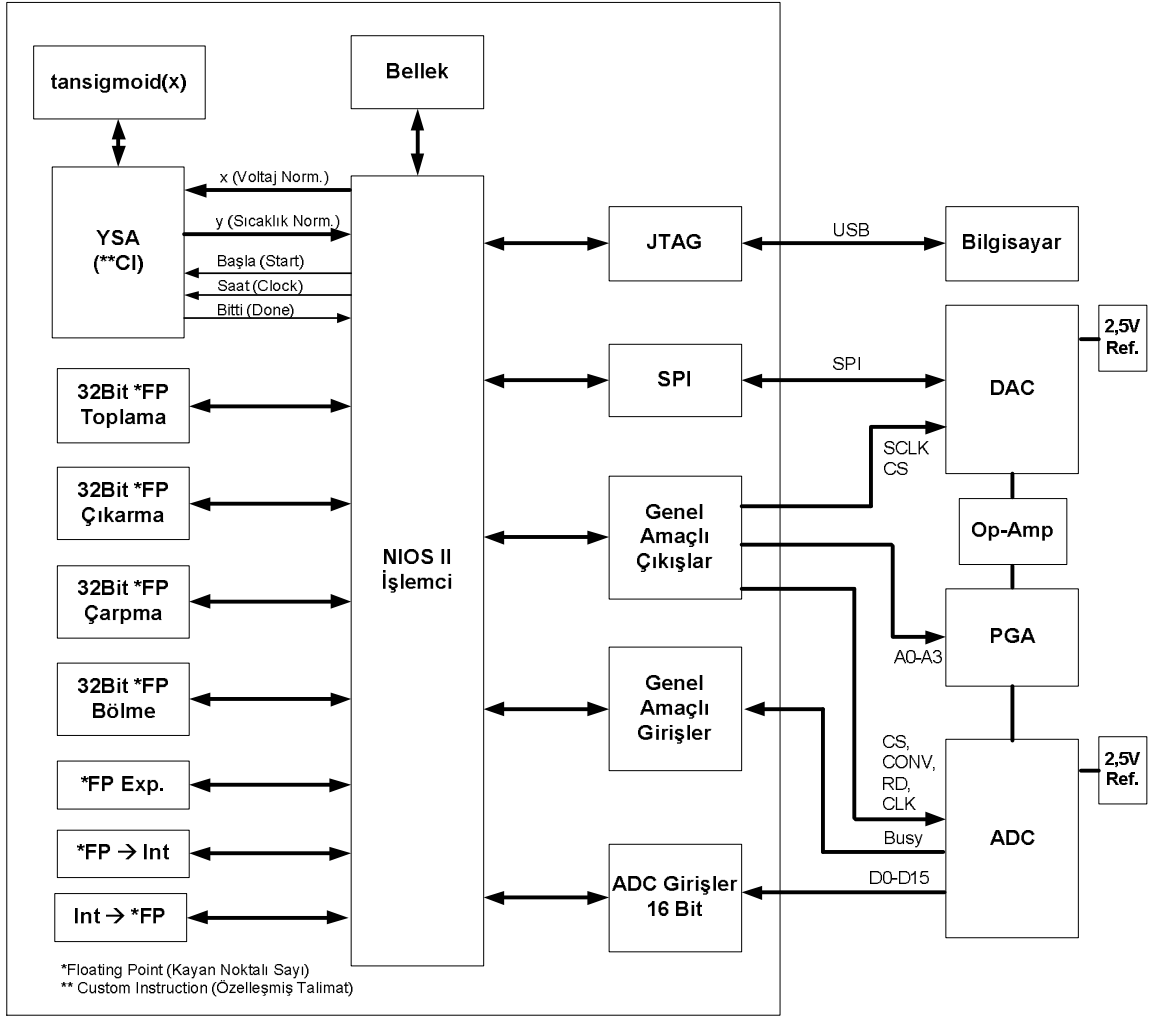
DE0 geliştirme ve eğitim kartı temel ihtiyaçlara cevap veren ve üzerinden pek çok sayıda genel kullanım amaçlı donanımlar bulunduran; kullanıcının sayısal tasarım ve FPGA'ler hakkında bilgi edinmesini kolaylaştıran bir platformdur. DE0 kartı Altera'nın Cyclone III 3C16 serisinden, 15.408 adet mantık elemanı (İng logic element – LE) sunan bir FPGA entegresine sahiptir. Kartın toplamda 346 adet giriş/çıkış pini bulunmaktadır. Terasic firması Altera'nın düşük güç tüketimli, düşük maliyetli ve yüksek performanslı Cyclone III FPGA'si ile DE0 kartının pek çok özelliğini bir araya getirmiştir [32].



Şekil 7.23 DE0 geliştirme kartının üstten görünümü [32]

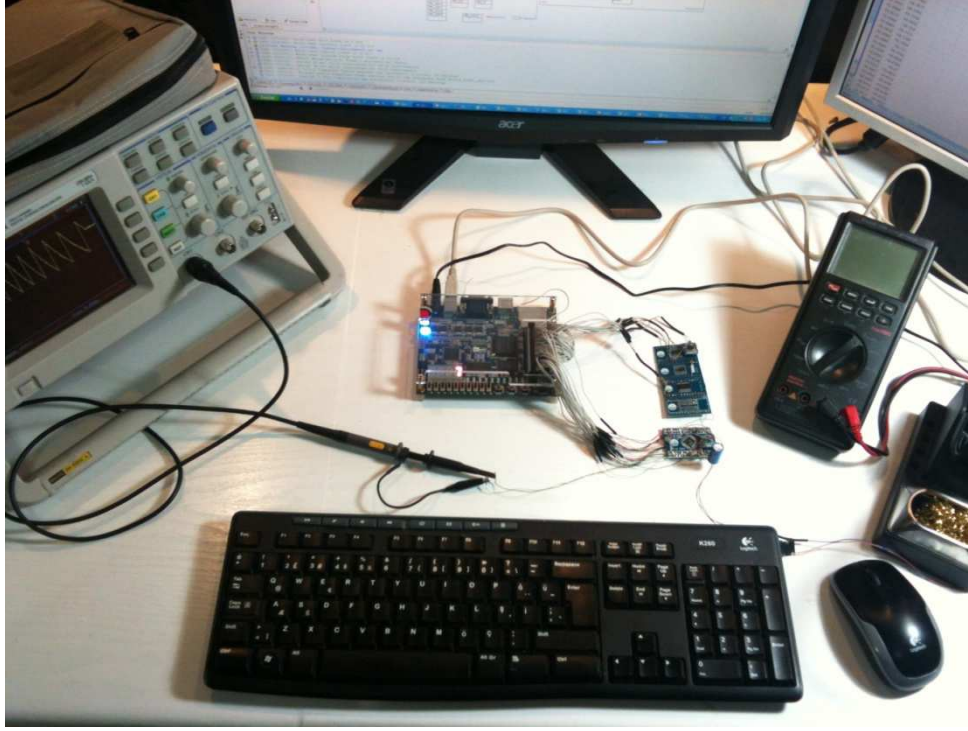
FPGA üzerinde Nios II işlemcisi çalıştırılmıştır. YSA uygulamaları için özelleşmiş talimat (İng. Custom Instruction) yazılmış ve ağların FPGA'in paralel işlem kabiliyeti kullanılarak mümkün olduğunca hızlı çalışması sağlanmıştır. Alteranın Qsys aracı kullanılarak FPGA üzerinde ihtiyaç duyulan gereken bileşenler bir araya getirilmiştir. Oluşturulan donanımın blok diyagramı Şekil 7.24'de verilmiştir. Donanım ve çalışma ortamının görüntüleri ise Şekil 7.25 ve Şekil 7.26'da verilmiştir.

FPGA

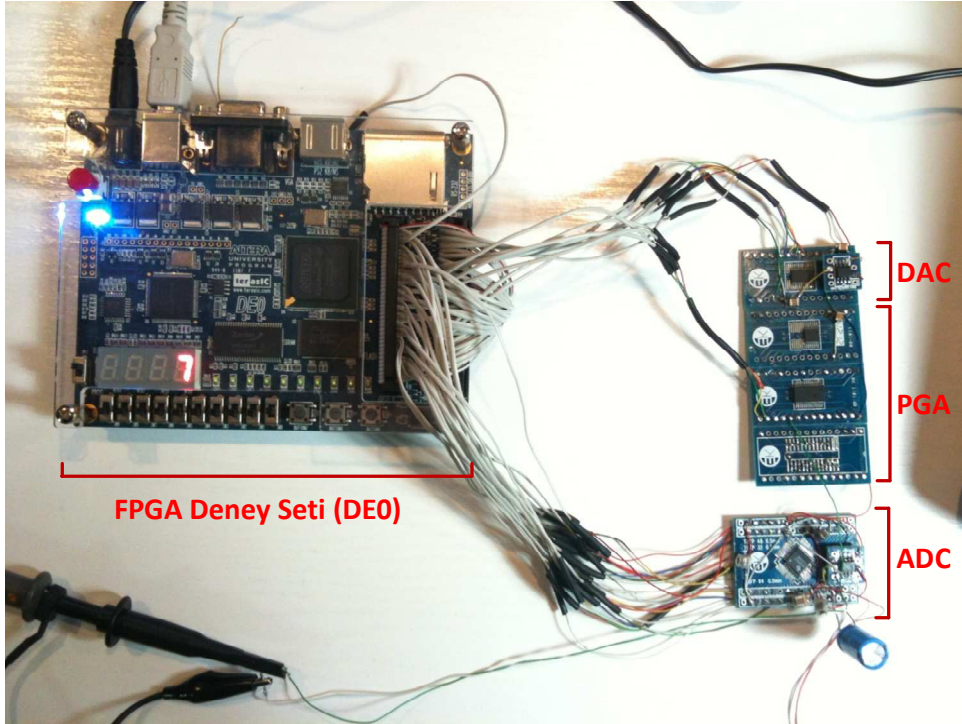


Şekil 7.24 Donanımın Blok Diyagramı

Test edilen sistemin çıktıları Nios II işlemcisinin JTAG protokolü üzerinden USB bağlantısıyla alınmıştır. Bu yöntemle alınan veriler MATLAB programına aktarılarak beklenen değerlerle karşılaştırılmış ve sistemlerin performansları ayrı ayrı değerlendirilmiştir.



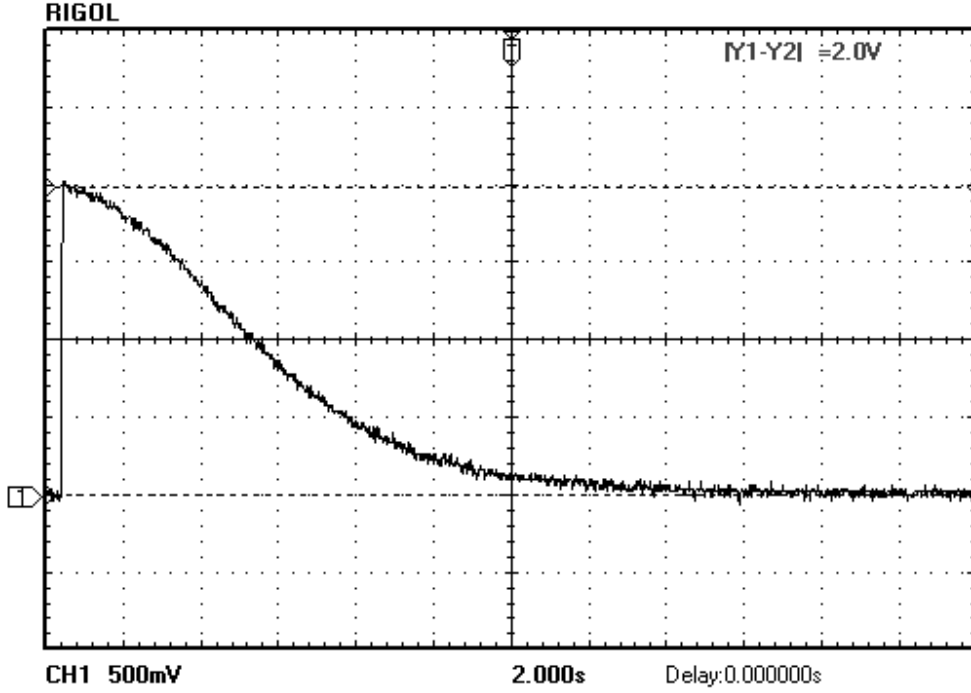
Şekil 7.25 Deney ortamı



Şekil 7.26 Kullanılan donanıma yakından bakış

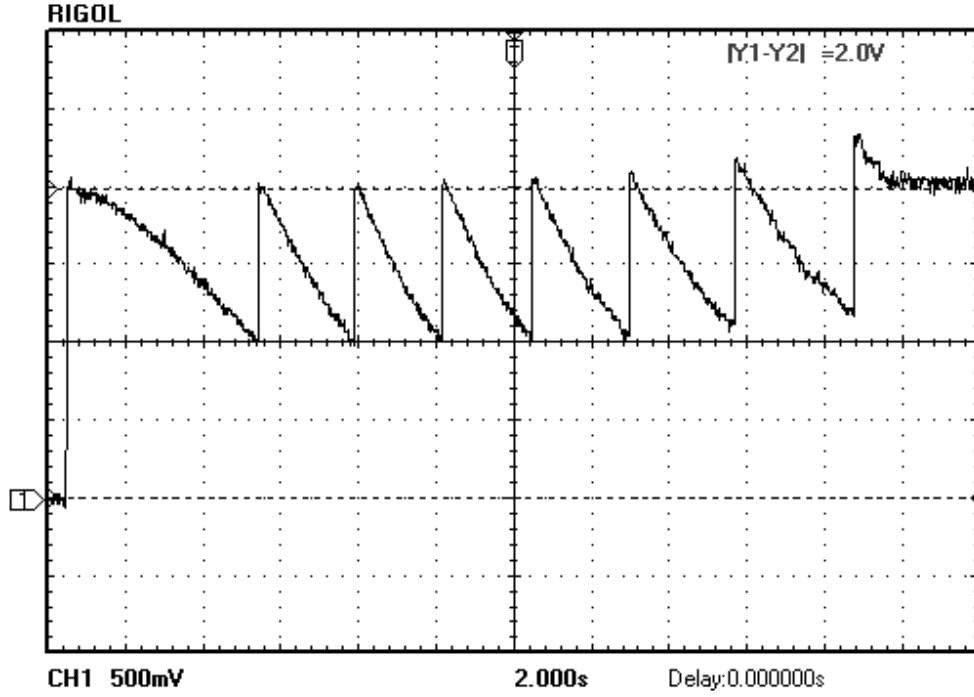
Sistem test edilirken ihtiyaç duyulan algılayıcı voltajı DAC çıkışıyla sağlanmıştır. - 55 °C'den +150°C'ye kadar olan aralıkta 0,1 °C adımlarla algılayıcı voltajı

üretmiş ve üretilen sinyal PGA üzerinden ADC'ye iletilmiştir. Tüm aralıkta PGA kazancının 1 olduğu durumda üretilen DAC voltajının ADC girişinden okunmasıyla elde edilen grafik Şekil 7.27'de, uygulamaya özgü olarak voltajın 1000mV'un altına düştüğünde PGA ile yükseltilmesi durumunda ADC girişinden okunan sinyal Şekil 7.28'de verilmiştir. PGA kazançlarının hangi voltaj aralıklarında uygulandığı Çizelge 7-7'de verilmiştir.



Şekil 7.27 DAC ile üretilen algılayıcı voltajı

Şekil 7.28'da kazancın artmasına bağlı olarak DC fark sinyalinin (İng. DC offset) arttığı görülmektedir. Bu durumun ölçüm sonuçlarına yansımaması için test işleminin başında her bir kazanç değeri için DC farklar ölçülmüş ve ADC okumalarından bu farklar çıkartılarak gerçek değer hesaplanarak işlem yapılmıştır.



Şekil 7.28 DAC ile üretilen algılayıcı voltajının PGA ile yükseltilmesi sonucu oluşan sinyal

Çizelge 7-7 PGA kazançlarına ait gerilim aralıkları

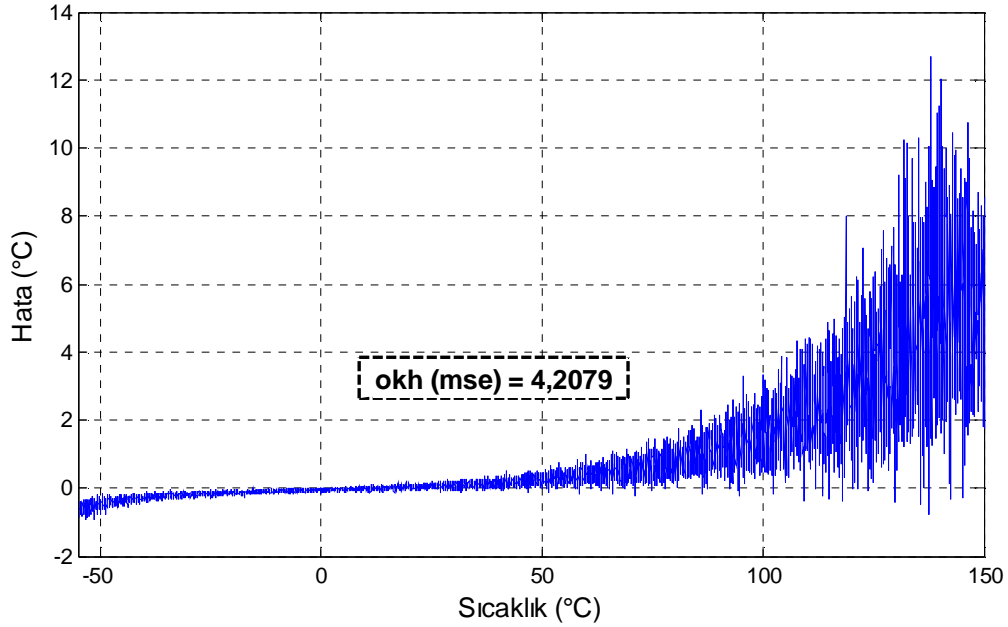
Kazanç	Gerilim Aralığı (mV)
1	1000 – 2000
2	500 – 1000
4	250 – 500
8	125 – 250
16	62,5 – 125
32	31,25 – 62,5
64	15,625 – 31,25
128	7,8125 – 15,625

7.3.5. KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerinde gerçekleştirilmesi

Bölüm 7.2.1'de benzetimi yapılan sistem donanım üzerinde çalıştırılmıştır. Test esnasında gerçekleştirilen işlemler

Şekil 7.30'de verilmiştir.

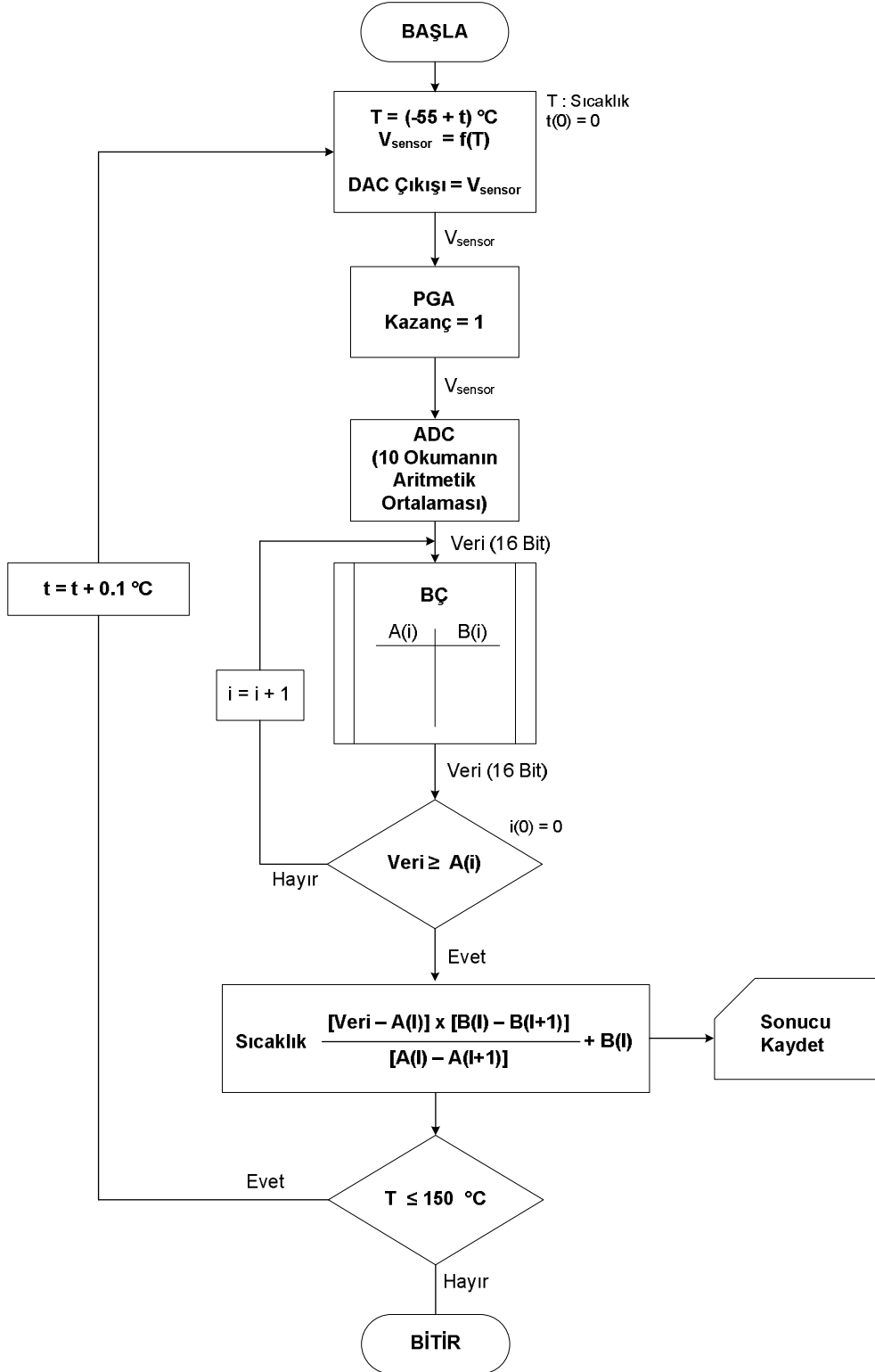
Test sonucunda elde edilen sıcaklık değerleri ile beklenen değerler arasındaki fark grafiği Şekil 7.29'de verilmiştir. Ortalama karesel hata 4,2079 olarak hesaplanmıştır.



Şekil 7.29 Donanım üzerinde gerçekleştirilen KPD yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı

Şekil 7.30'de görüldüğü gibi, başvuru çizelgesini tarama işleminin süresi girdinin seviyesine bağlı olarak değişmektedir. Ölçülen minimum işlem süresi 288,1 µs, maksimum işlem süresi 1210,3 µs ve ortalama işlem süresi 772,6 µs olmuştur.

Yöntemin kullandığı bellek miktarı 6144 byte olmuştur.

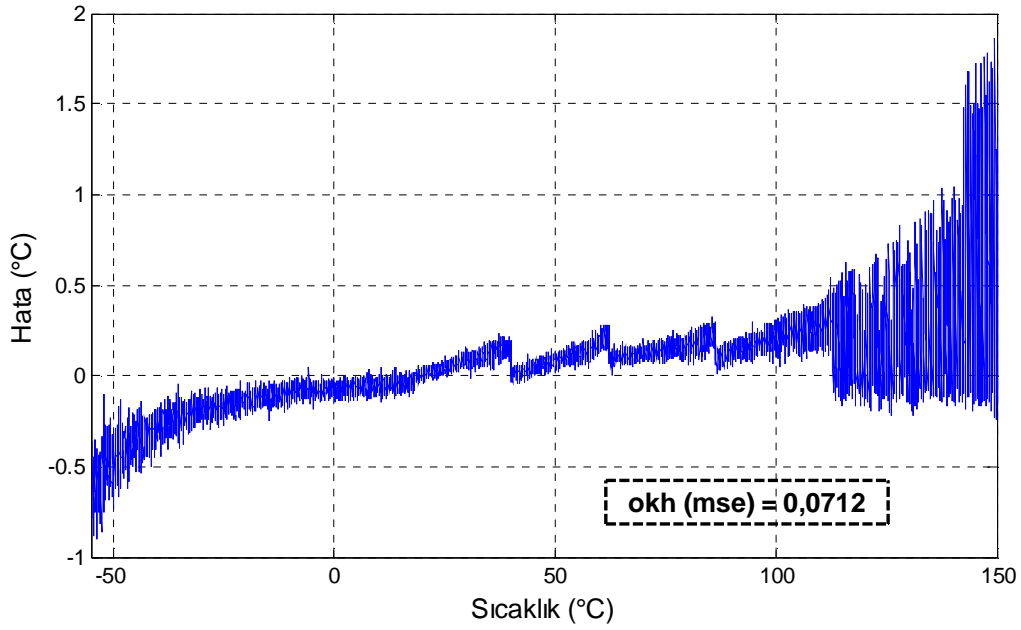


Şekil 7.30 KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerindeki testinin gösterimi

7.3.6. PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım uygulaması

Bölüm 7.2.2’de benzetimi yapılan sistem donanım üzerinde çalıştırılmıştır. Test esnasında gerçekleştirilen işlemler Şekil 7.32’de verilmiştir.

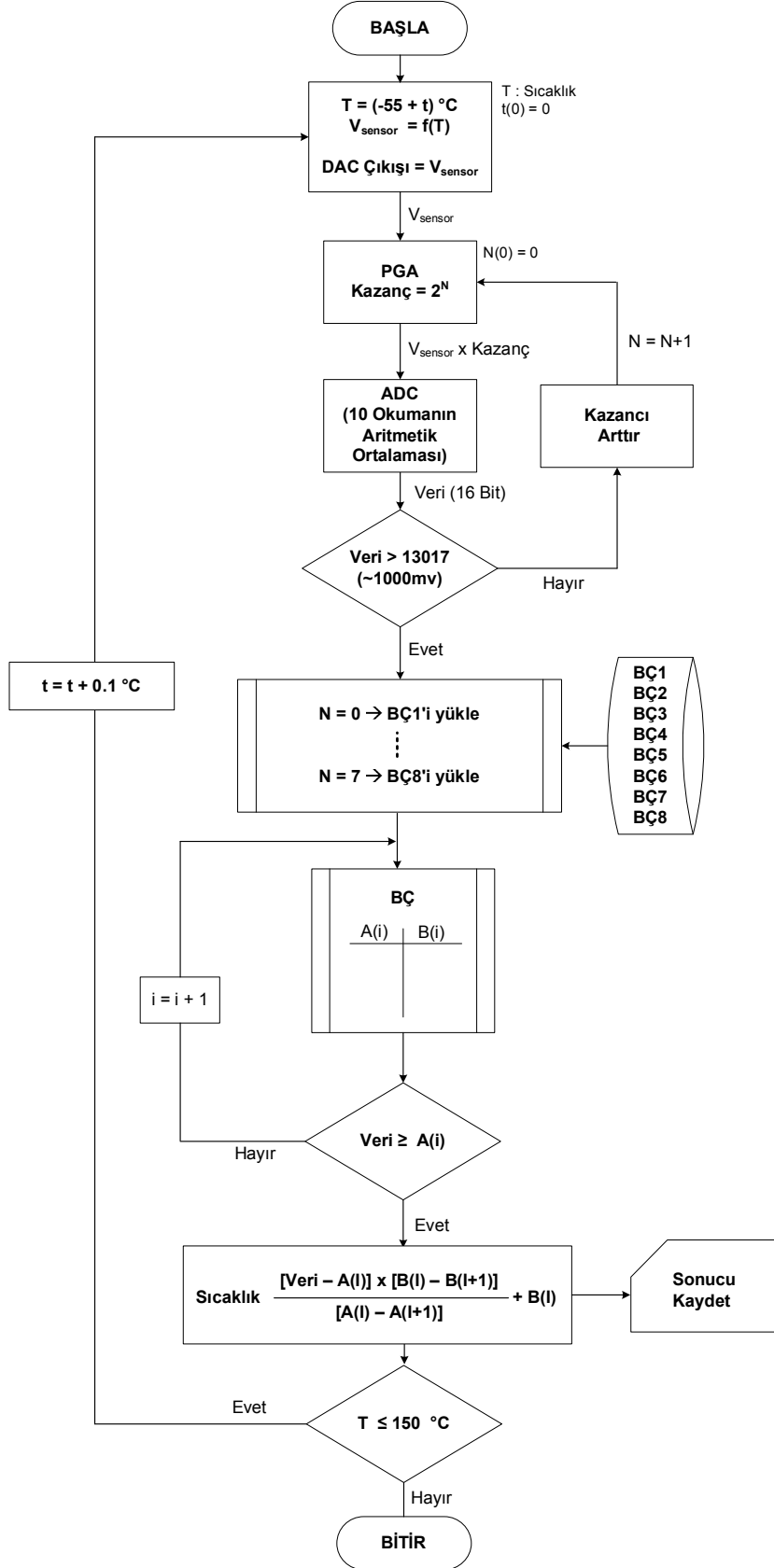
Test sonucunda elde edilen sıcaklık değerleri ile beklenen değerler arasındaki fark grafiği Şekil 7.31’de verilmiştir. Ortalama karesel hata 0,0712 olarak hesaplanmıştır.



Şekil 7.31 Donanım üzerinde gerçekleştirilen PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı

Şekil 7.32’de görüldüğü gibi sistemin çıktı üretme süresi hem kazanç arttırmalarına bağlı olarak girdinin seviyesine hem de nihai sinyalin çizelgede karşılık gelen konumuna göre farklılık göstermektedir. Ölçülen minimum işlem süresi 301,1 µs, maksimum işlem süresi 2488,2 µs ve ortalama işlem süresi 432,1 µs olmuştur.

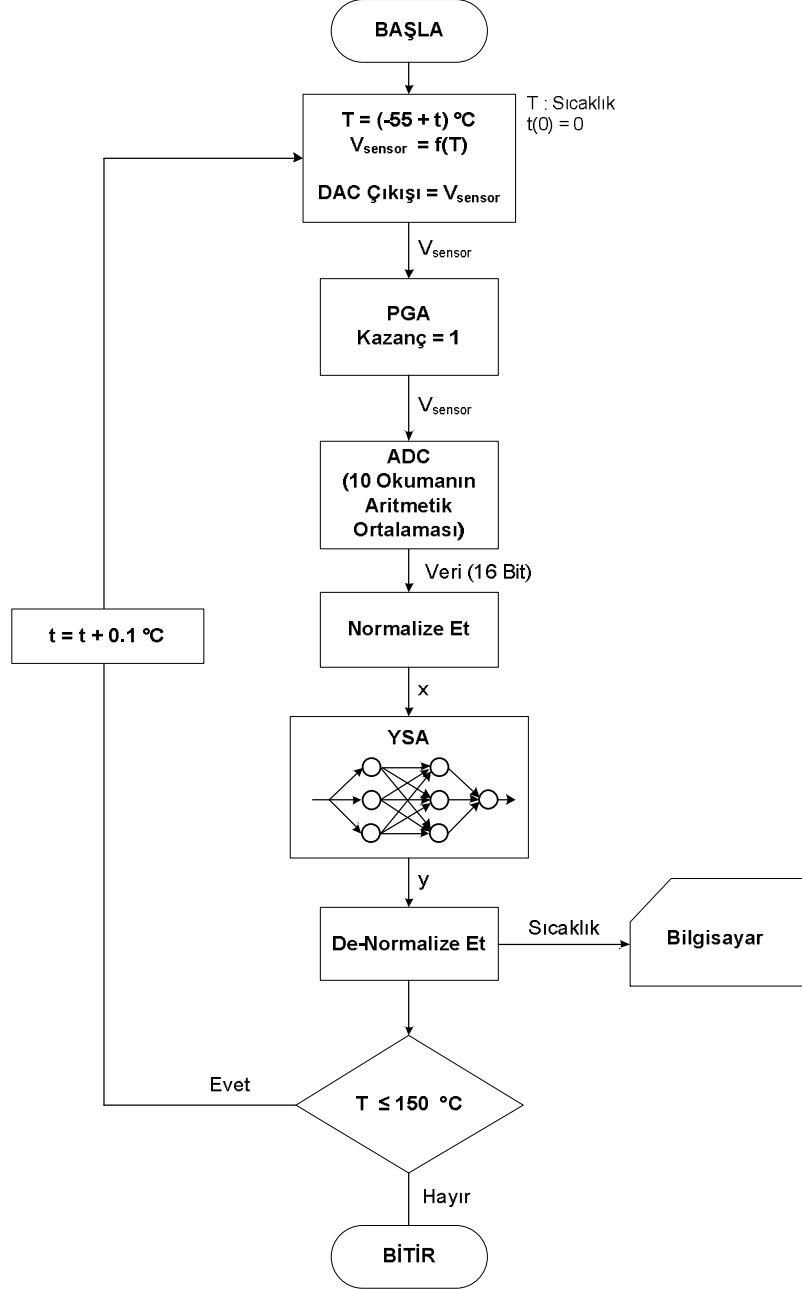
Yöntemin kullandığı bellek miktarı 6144 byte olmuştur.



Şekil 7.32 PGA + KPD yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerindeki testin akış diyagramı

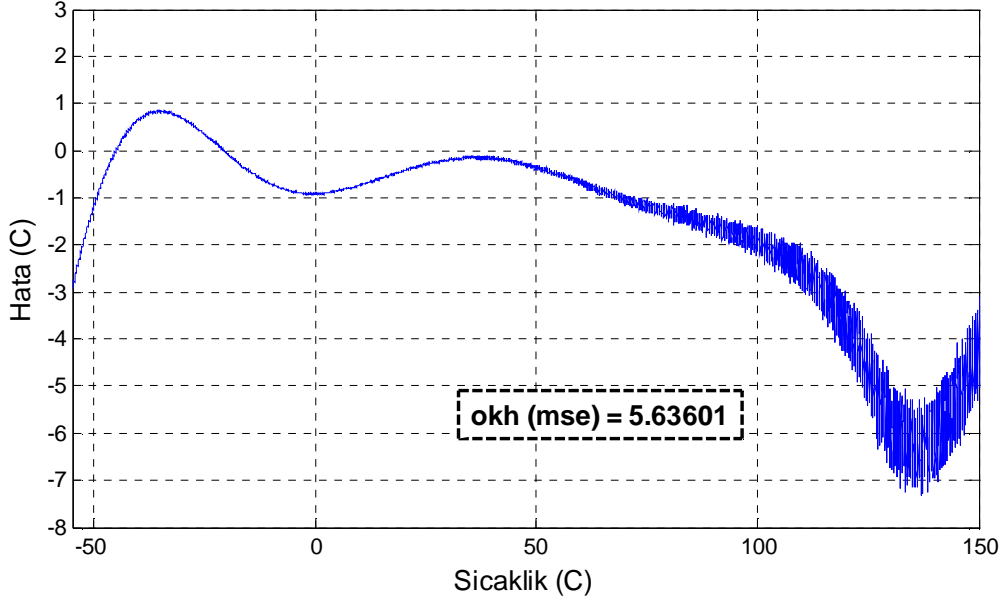
7.3.7. YSA yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerinde gerçekleştirilmesi

Bölüm 7.2.3'de benzetimi yapılan sistem donanım üzerinde çalıştırılmıştır. Test esnasında gerçekleştirilen işlemler Şekil 7.33'de verilmiştir.



Şekil 7.33 YSA yöntemi kullanılan sistemin donanım üzerindeki testinin akış diyagramı

Test sonucunda elde edilen sıcaklık deęerleri ile beklenen deęerler arasındaki fark grafięi Őekil 7.34'de verilmiřtir. Ortalama karesel hata 5,63601 olarak hesaplanmıřtır.



Őekil 7.34 Donanım üzerinde geręeklenen YSA yntemi kullanılan doęrusallařtırma sisteminin ıkıřında llen hatanın -55  C – 150 C aralıęındaki daęılımı

Sistemin her bir test verisini iřlemesi iin geen srenin ortalama 633,6 s olduęu llmřtr.

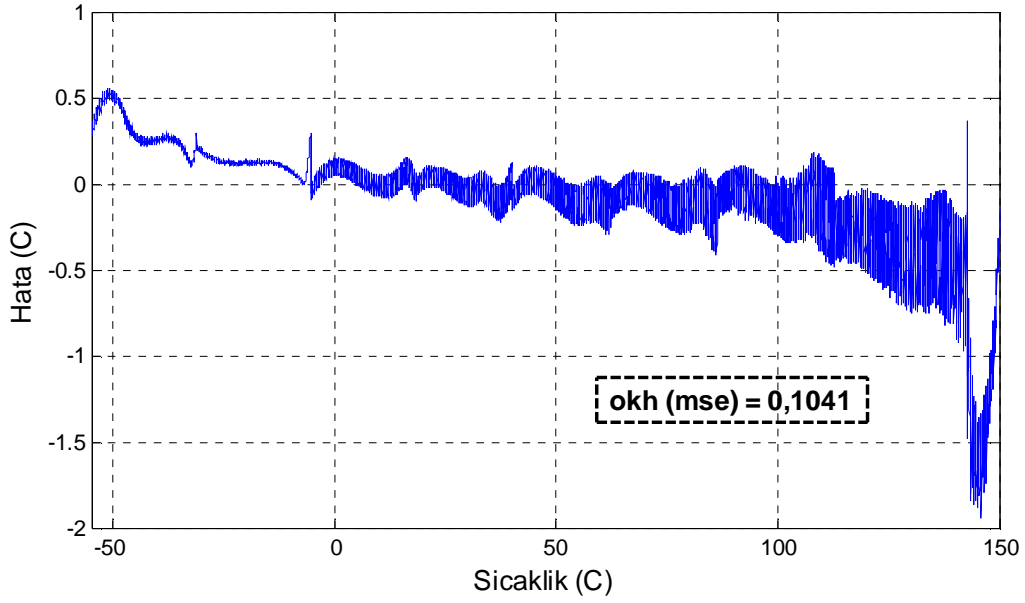
Yntemin kullandıęı bellek miktarı 104 byte olmuřtur.

7.3.8. PGA + YSA yntemi kullanılan doęrusallařtırma sisteminin donanım üzerinde geręeklenmesi

Blm 7.2.4'de benzetimi yapılan sistem donanım üzerinde alıřtırılmıřtır. Test esnasında gerekleřtirilen iřlemler Őekil 7.36'da verilmiřtir.

Test sonucunda elde edilen sıcaklık deęerleri ile beklenen deęerler arasındaki fark grafięi Őekil 7.35'da verilmiřtir. Ortalama karesel hata 0,1041 olarak

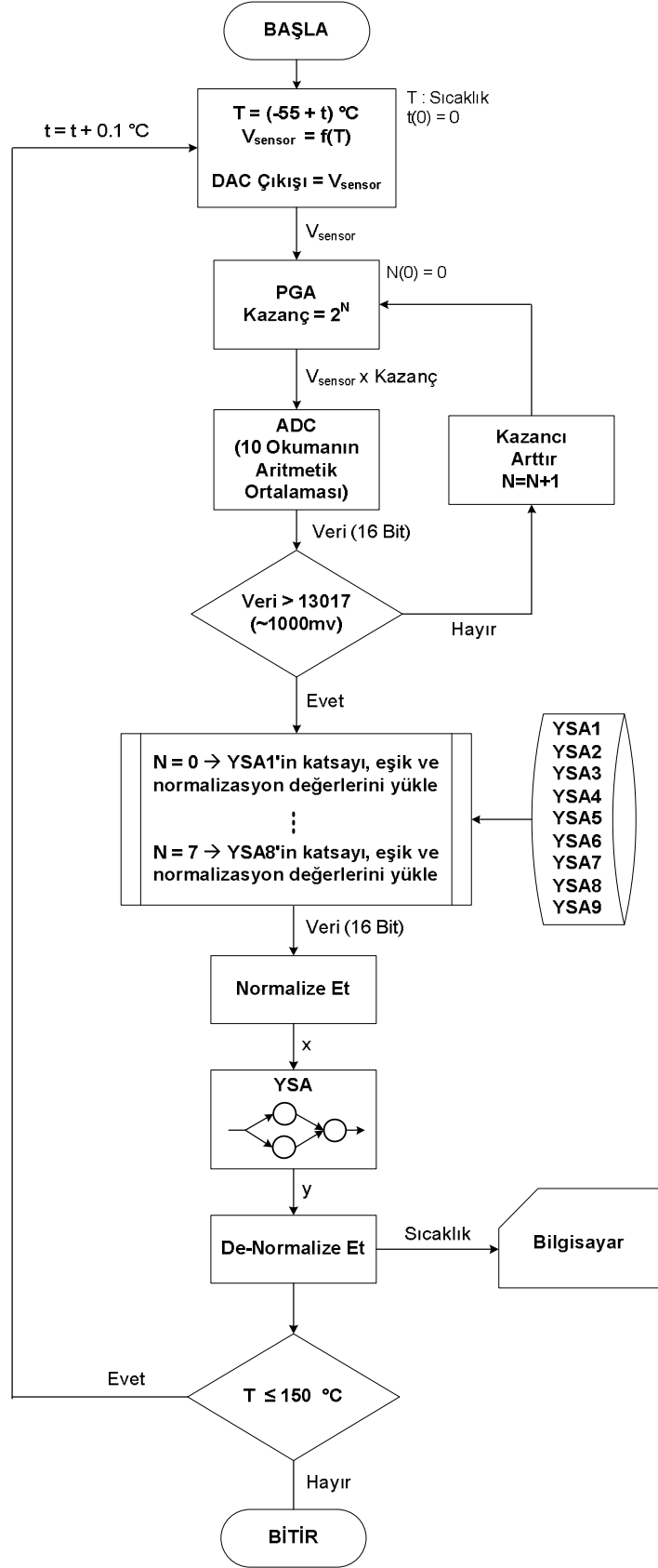
hesaplanmıştır.



Şekil 7.35 Donanım üzerinde gerçekleştirilen PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin çıkışında ölçülen hatanın -55 °C – 150°C aralığındaki dağılımı

Şekil 7.36'de de görüldüğü gibi sisteme 1000mV'un altında bir girdi sağlandığında, sistem bu sinyali 1000mV'un üzerine çıkarmak için PGA kazancını arttırmaktadır. Bu da farklı seviye aralıklarındaki sinyallerin işleme sürelerini değiştirmektedir. Ölçülen çıktı üretme süresi minimum 296,0 µs, maksimum 2300,0 µs ve ortalama 1157,9 µs olmuştur.

Yöntemin kullandığı bellek miktarı 396 byte olmuştur.



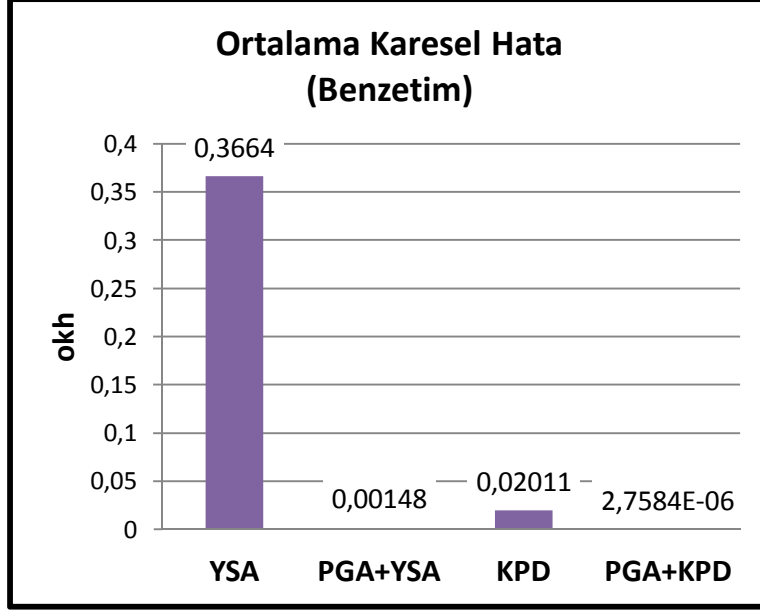
Şekil 7.36 PGA + YSA yöntemi kullanılan doğrusallaştırma sisteminin donanım üzerindeki testin akış diyagramı

7.4. Benzetim ve Donanım Uygulamaları Sonuçlarının Değerlendirilmesi

Algılayıcı doğrusallaştırma uygulamalarında, daha hızlı tepki süresi, daha az bellek kullanımı ve daha düşük ortalama karesel hata istenilen başlıca kriterlerdir. Bu tez kapsamında gerçekleştirilen uygulamalarda KPD, PGA+KPD, YSA ve son olarak da önerilen yöntem olan PGA+YSA yöntemleri ele alınmıştır. Toplamda dört adet yöntemin kullanıldığı algılayıcı doğrusallaştırma sistemlerinin benzetimleri ve donanım üzerinde uygulamaları gerçekleştirilmiş ve elde edilen sonuçlar karşılaştırılmıştır.

Benzetim sonucunda elde edilen ortalama karesel hatalar karşılaştırıldığında YSA yöntemi kullanılan sistemin (bkz.Şekil 7.37) performansının diğer üç yöntemin kullanıldığı sistemlere kıyasla düşük olduğu görülmektedir. YSA yöntemi kullanılan bu uygulamada kullanılan ağırlık iki ara katmanı ve her ara katmanda üçer adet nöronu bulunmaktadır. Performansı iyileştirmek üzere ara katman ve nöron sayıları arttırılabilir ancak bu durumda artan ağırlık ve eşik değerleri sayısı ile paralel işlem yapmak için kullanılması gereken donanım kaynakları artacağından uygulamanın maliyeti artacaktır ve hızı düşecektir. PGA+YSA yöntemi kullanılan uygulamada ise performansın büyük ölçüde iyileştiği görülmektedir. Ayrıca PGA+YSA yönteminde hızı arttırmak ve bellek kullanımını azaltmak amacıyla ağ yapısı basitleştirilmiş, tek ara katmana sahip ve bu katmanında iki adet nöron bulunan ağ yapısına geçilmiştir. Şekil 7.37’de görüldüğü gibi benzetim sonucunda en iyi ortalama karesel hata PGA+YSA yöntemi kullanılan sistemden elde edilmiştir.

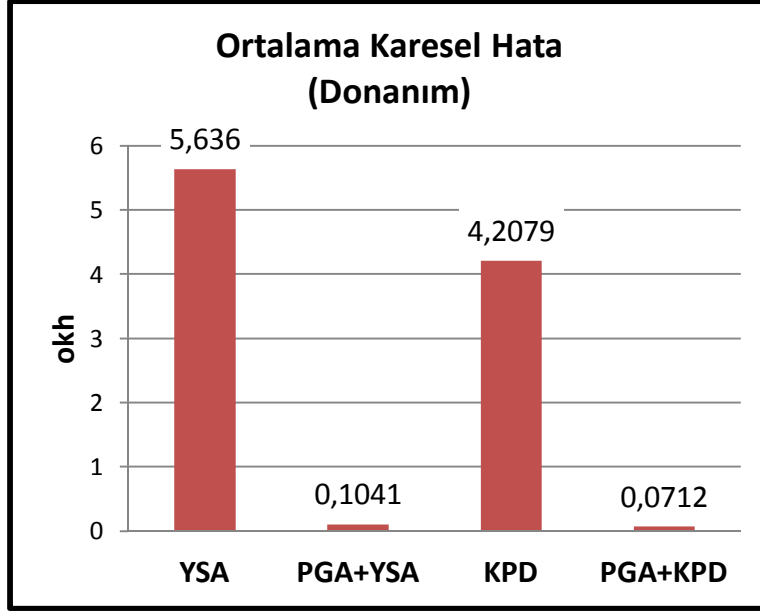
Benzetimi yapılan sistemler aynı zamanda donanım üzerinde de gerçekleşmiş ve sonuçlar ortalama karesel hata, ortalama işlem süresi ve bellek kullanımı yönünden karşılaştırılmıştır.



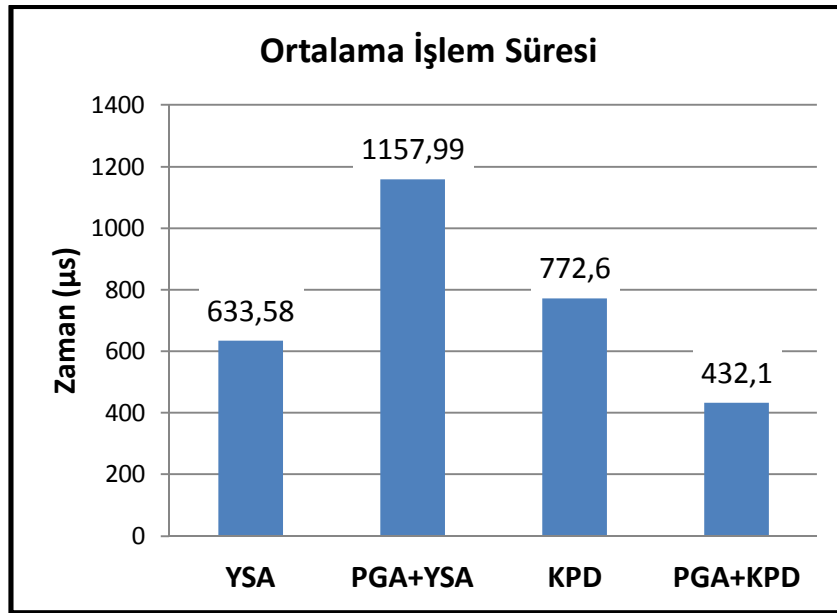
Şekil 7.37 Benzetim sonucunda elde edilen ortalama karesel hatalar

Şekil 7.38 incelendiğinde donanım uygulaması sonucunda en düşük ortalama karesel hatanın PGA+KPD yöntemi kullanılan sistemde elde edildiği görülmektedir. PGA+YSA yöntemi ortalama karesel hata yönünden ikinci sırada yer almaktadır. Benzetim sonuçlarıyla karşılaştırıldığında bu durum beklenen bir sonuçtur ancak iki yöntem arasındaki hata farkı benzetimde karşılaşılan oranda olmamıştır. KPD yöntemi kullanılan sistemde elde edilen ortalama karesel hataya bakıldığında benzetim sonucundan oldukça yüksek bir hata ile karşılaşılmıştır. Bu durum geniş bir voltaj aralığında uygulanan girdi sinyallerinin yüksek sıcaklıklarda oldukça dar bir voltaj aralığına sıkışması, ADC çözünürlüğünün kısıtlı olması, düşük voltajlarda SNR'nin (Sinyal Gürültü Oranı) düşük olması ve devredeki diğer kayıpların etkisiyle de okuma hataları artmış, sonuç itibarıyla ortalama karesel hata artmıştır. Gürültülerin filtrelenmesi, ADC çözünürlüğünün artırılması gibi iyileştirmeler, okh'nin benzetim sonucuna yaklaşmasını sağlayacaktır ancak bu durum da tasarımın karmaşıklaşmasına ve maliyetin artmasına yol açacaktır.

Donanım uygulamasında elde edilen okh'ler incelendiğinde PGA+YSA ve PGA+KPD yöntemleri kullanan sistemlerin kabul edilebilir hata seviyesinde sonuç üretebildikleri görülmektedir. Bu iki yöntemin hızları karşılaştırıldığında ise (Şekil 7.39) PGA+KPD yönteminin yaklaşık 2,7 kat daha hızlı tepki verdiği görülmektedir.

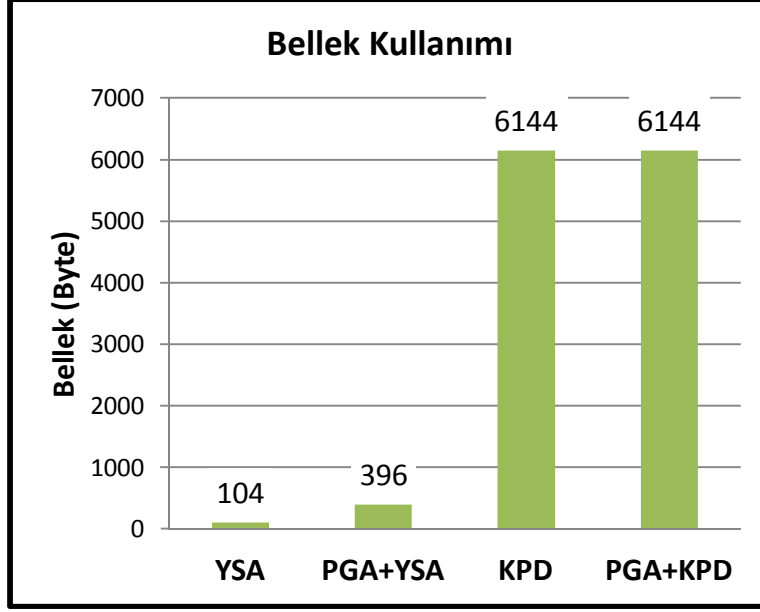


Şekil 7.38 Donanımsal uygulama sonucunda elde edilen ortalama karesel hatalar



Şekil 7.39 Donanımsal uygulama sonucunda elde edilen ortalama işlem süreleri

Yöntemlerin bellek kullanımları karşılaştırıldığında ise KPD tabanlı yöntemlerin YSA tabanlı yöntemlere kıyasla çok daha fazla bellek alanına ihtiyaç duyduğu görülmektedir. (Şekil 7.40)



Şekil 7.40 Donanımsal uygulama sonucunda elde edilen bellek kullanım miktarları

Sonuç olarak, PGA'nın kullanım hem YSA tabanlı sistemde hem de KPD tabanlı sistemde performansın önemli ölçüde artmasını sağlamıştır. PGA+YSA ve PGA+KPD yöntemleri yakın otalama hatalarda sonuç üretmişlerdir. Ancak KPD yönteminin fazla bellek kullanımı, YSA yönteminin ise eğitim süreci, yöntemlerin dezavantajları olarak karşımıza çıkmaktadır. Bunun yanısıra KPD tabanlı bir doğrusallaştırma uygulamasını gerçekleştirmek YSA yöntemine göre daha zahmetsizdir.

8. SONUÇ VE TARTIŞMA

Bu çalışmada FPGA tabanlı sayısal algılayıcı doğrusallaştırma uygulamaları gerçekleştirilmiştir. Uygulamanın temel amacı doğrusallaştırma hızını arttırmak, bellek kullanımını azaltmak ve ortalama karesel hatayı düşürmektir. Bu amaç doğrultusunda PGA ve YSA yapılarının avantajları birleştirilmiş, FPGA'nın paralel işlem kabiliyetleri kullanılmıştır. Elde edilen sonuçlar KPD, PGA+KPD, YSA gibi benzetimi ve donanım uygulaması gerçekleştirilen diğer yöntemlerle karşılaştırılmıştır.

En düşük ortalama karesel hata, PGA+KPD yönteminde elde edilmiştir. Bu sonuç hem benzetim hem de donanım çalışmalarında elde edilmiştir. Okh yönünden, PGA+KPD ve PGA+YSA yöntemlerinin kullanıldığı sistemlerin, $-55\text{ }^{\circ}\text{C} - 150\text{ }^{\circ}\text{C}$ aralığında diğer yöntemlere kıyasla daha iyi sonuç verdikleri görülmektedir. Bu iki yöntemin hızları karşılaştırıldığında ise PGA+KPD yöntemi kullanılan sistemin PGA+YSA yöntemi kullanılan sisteme göre 2,7 kat daha hızlı tepki verdiği görülmüştür.

Bellek kullanımına bakıldığında, KPD tabanlı yöntemlerin YSA tabanlı yöntemlere oranla çok daha fazla bellek alanı kullandığı görülmektedir. YSA tabanlı yöntemler arasında ise YSA+PGA yönteminin YSA yöntemine kıyasla yaklaşık 3 kat fazla bellek kullandığı sonucu ortaya çıkmıştır. Önerilen PGA+YSA yöntemi diğer yöntemlere kıyasla bellek kullanımı yönünden daha iyi sonuç vermiştir. Bu sebeple bellek kullanımının azlığının önem kazandığı uygulamalarda PGA+YSA yönteminin kullanılması, hızın ve hata azlığının önem kazandığı uygulamalarda ise PGA+KPD yönteminin kullanılması önerilmektedir.

Donanım uygulamalarının hata grafiklerine bakıldığında, PGA kullanımının algılayıcının ölçüm aralığını önemli oranda genişlettiği görülmektedir. PGA kullanılmayan yöntemlerde efektif ölçüm aralığının yaklaşık $-55^{\circ}\text{C} - 55^{\circ}\text{C}$ olduğu görülürken, PGA kullanılan yöntemlerde bu aralığın yaklaşık $-50^{\circ}\text{C} - 140^{\circ}\text{C}$ olduğu görülmektedir. Geniş sıcaklık aralığında ölçüm yapmanın gerekli olduğu uygulamalarda PGA'nın kullanılması tavsiye edilmektedir. Bu çalışmada kazanç

dirençleri deęiřtirilebilen, paralel arayüzle kontrol edilen PGA yapısı kullanıldığından hız yükselmiş ancak maliyet artmıştır.

Önerilen PGA+YSA yönteminde MLP aę yapısı kullanılmıştır. Bu yöntemin performansını arttırmak üzere tek girişli ve tek çıkışlı aęlarda daha iyi performans verdiği ifade edilen RBF (Radial Basis Function) yapısındaki YSA'lar da kullanılabilir.

KAYNAKLAR LİSTESİ

- [1] HABERLI, A., PAUL, O., MALCOVATI, P., FACCIO, M., MALOBERTI, F., BALTES, H., CMOS integration of a thermal pressure sensor system, IEEE ISCAS'96, vol.1, s.377-80 ,1996.
- [2] ERDEM, Hamit, Implementation of software-based sensor linearization algorithms on low-cost microcontrollers, ISA Transactions, vol. 49, s.552, 2010.
- [3] ATTARI, M., Methods for linearization of non-linear sensors, CMMNI-4, vol.1, s.344-50, 1993.
- [4] KRASKA, M.P., Digital Linearization & Display of Non-linear Analog (Sensor) Signals, IEEE Transactions on Instrument and Measurement, vol.38, s.45-51, 1988.
- [5] BRIGNELL, J.E., Software techniques for sensor compensation, Sensors Actuators, vol.25-27, s.29-35, 1991.
- [6] PATRANABIS, D., GHOSH, D., BAKSHI, C., Linearizing transducer characteristics, IEEE Trans Inst Meas, vol.37, no.1, s.66-9, 1988.
- [7] MAHANA, P.N., TROFIMENKOFF, F.N., Transducer output signal processing using an eight-bit microcomputer, IEEE Trans Inst Meas, vol.35, no.2, s.182-6, 1986.
- [8] FLAMMINI, A., MARIOLI, D., TARONI, A., Transducer output signal processing using an optimal look-up table in microcontroller based systems, Electron Lett, vol. 33, no.14, s.1197-8, 1997.
- [9] CATUNDA, S.Y.C., SAAVEDRA, O.R., FONSECANETO, J.V., MORAIS, M.R.A., Look-up table and breakpoints determination for piecewise linear approximation functions using evolutionary computation. IEEE IMTC'03, vol.1, s.435-40, 2003.
- [10] BAKER, B.C., Advances in measuring with nonlinear sensors, Sensors Mag, 2005.
- [11] FREEMAN, J.A., Neural Networks: Algorithms, Applications, and Programming Techniques, Addison-Wesley, Massachusetts, 1992.
- [12] AGGOUNE, M.E., BOUDJEMA, F., BENSENOUCI, A., HELLAL, A., VADARI, S.V., EL MESAI, M.R., Design of an adaptive-structure voltage regulator using artificial neural networks, 2nd IEEE Conference on Control Applications, Vancouver, Canada, s.337-343, 1993.
- [13] PAU, L.F., JOHANSEN, F.S., Neural Network Signal Understanding for Instrumentation, IEEE Trans on Inst & Meas, vol.39, no.4, s.558-564, 1990.
- [14] ATTARI, M., BOUDJEMA, F., HENICHE, M., Linearizing a Thermistor Characteristic in the Range of Zero to 100 Degree C with Two Layers Artificial

Neural Networks, IEEE Instrumentation and Measurement Technology Conference, 1995.

- [15] MEDRANO-MARQUES, N.J., MARTIN-DEL-BRIO, B., A General Method for Sensor Linearization Based on Neural Networks, IEEE International Symposium on Circuits and Systems, s.497-500, 2000.
- [16] Medrano-Marques, N.J., Implementing Neural Networks onto Standard Low-Cost Microcontrollers for Sensor Signal Processing, IEEE Conference, ETFA, vol.2, s.967-72, 2005.
- [17] DOĞAN, Ömer, Sensör Lineerleştirme Amaçlı Mikrodenetleyicili Tasarım, s.16-30, 2006.
- [18] MEDRANO-MARQUÊS, N.J., MARTIN-DEL-BRIO, B., A General Method for Sensor Linearization Based on Neural Networks, IEEE International Symposium on Circuits and Systems, s. 497-500, 2000.
- [19] BUCCI, Giovanni, New ADC with Piecewise Linear Characteristic: Case Study – Implementation of a Smart Humidity Sensor, IEEE Transactions on Instrument and Measurement, vol.49, s.1154-1166, 2000.
- [20] YILMAZ, Necla, Alan Programlamalı Kapı Dizileri (FPGA) Üzerinde Bir YSA'nın Tasarlanması ve Donanım Olarak Gerçekleştirilmesi, Yüksek Lisans Tezi, Selçuk Üniversitesi, Konya, s.18-19 2008.
- [21] DOĞAN, Recep, Face Recognition Using Neural Networks on Field Programmable Gate Array, Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi, İzmir, s.21-31, 2011.
- [22] AKGÜN DAR, Kadriye, Sinir Dokusu ve Sinir Sistemi, İstanbul Üniversitesi Fen Fakültesi Biyoloji Bölümü, 2009.
- [23] SIRMAÇEK, Beril, FPGA ile Mobil Robot İçin Öğrenme Algoritması Modellenmesi, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, İstanbul, s.3, 2007.
- [24] http://andreeasanatomy.blogspot.com/2011/04/you-need-to-step-up-on-step-to-reach_23.html, 2012.
- [25] DUMAN, Nihal, Yapay Sinir Ağları ve Bir Uygulama, Yüksek Lisans Tezi, s.2-18, 2006.
- [26] KORKUT, Murat, Yapay Sinir Ağları Tabanlı Fonksiyonel Yaklaşım ve Mühendislik Uygulamaları, Yüksek Lisans Tezi, Gebze Yüksek Teknoloji Enstitüsü, s.20, 2007.
- [27] SARITEKİN, Namık Kemal, Yapay Sinir Ağlarının Otomatik Olarak FPGA'YA Uygulanması İçin Veri Yolu Tasarım Aracı, Yüksek Lisans Tezi, Düzce Üniversitesi, s14-16, 2011.

- [28] DURMUŞ, Ali, Yapay ve Bulanık Sinir Ağları ile Sistemlerin Modellenmesi, Yüksek Lisans Tezi, Erciyes Üniversitesi, Kayseri, s.25, 2005.
- [29] NEGNEVITSKY, Michael, Artificial Intelligence: A guide to Intelligent Systems, Addison Wesley, ISBN: 0-321-20466-2, Ch.6, 2005.
- [30] JUNG, Walt Op Amp Applications Handbook, s.2.31-2.33, Newness, 2006.
- [31] SERRANO, J., Introduction to FPGA Design, CAS – CERN Accelerator School: Course on Digital Signal Processing, s.231-247, 2007.
- [32] <http://www.terasic.com.tw/cgi-bin/page/archive.pl?No=364>, Altera De0 Board, 2012.
- [33] BAKER, Bonnie C., Thermistors in Single Supply Temperature Sensing Circuits, s.1-6, Microchip Technology Inc., 1999.
- [34] ÇAVUŞLU, Mehmet Ali ve KARAKUZU, Cihan, Nöral ve Bulanık Sistem Hücre Aktivasyon Yaklaşımları ve FPGA'da Donanımsal Gerçeklenmesi, SAÜ. Fen Bilimleri Dergisi, 15. Cilt, 1. Sayı, s.8-16, 2011