

IMPLEMENTATION OF IQ-MATH BASED LINEAR ACTIVATION FUNCTIONS ON FPGA

IQ-MATH TABANLI DOĞRUSAL AKTİVASYON FONKSİYONLARININ FPGA ÜZERİNDE GERÇEKLENMESİ

Doç. Dr. İsmail KOYUNCU ¹, Mehmet Şamil AKÇAY ^{2*}, Dr. Öğr. Üyesi Murat TUNA ³, Dr. Öğr. Üyesi Murat ALÇIN ⁴

¹Afyon Kocatepe University, Technology Faculty, Department of Electrical-Electronics Engineering, Afyonkarahisar/Turkey.

^{2*}Afyon Kocatepe University, Institute of Natural and Applied Sciences, Department of Electrical-Electronics Engineering, Afyonkarahisar/ Turkey.

³Kırklareli University, Technical Sciences Vocational School, Department of Electrical, Kırklareli/Turkey.

⁴Afyon Kocatepe University, Technology Faculty, Department of Mechatronics Engineering, Afyonkarahisar/Turkey.

ABSTRACT

Nowadays, Artificial Neural Networks (ANN), which is one of the widely used fields of artificial intelligence, has been commonly used in many areas including regression, estimation, decision making, classification, image and voice recognition, nonlinear signal processing and chaotic oscillator design. ANN, implemented in two different ways as software-based and hardware-based, has features such as parallel signal processing and distributed information processing. Therefore, ANN is known as a structure that includes very intensive mathematical operations. Hardware-based ANN applications can be implemented using many different platforms. FPGA (Field Programmable Gate Array) chips, as one of these platforms, have parallel processing capacity with high operating speed. In this study, a Linear Activation Functions Library has been created by implementing 6 different linear activation functions on FPGA for real time ANN applications. The designs have been coded using VHDL (Very High Speed Integrated Circuit Hardware Description Language) in accordance with 32-bit (16I-16Q) IQ-Math number standard. All designs have been tested using Xilinx ISE Design Suite program. After the test phase, the implementations have been synthesized for Xilinx Kintex-7 FPGA chip. The chip statistics and performance analyses obtained from FPGA-based activation functions have been presented.

Keywords: Artificial Neural Networks, Activation Functions, IQ-Math number standard, FPGA.

ÖZET

Günümüzde yaygın olarak kullanılan yapay zeka alanlarından birisi olan Yapay Sinir Ağları (YSA) regresyon, tahmin etme, karar verme, sınıflandırma, görüntü ve ses tanıma, doğrusal olmayan sinyal işleme ve kaotik osilatör tasarımı gibi birçok alanda yaygın olarak kullanılmaktadır. Genellikle yazılım (software) ve donanım (hardware) olmak üzere iki farklı şekilde gerçekleştirilebilmekte olan YSA'lar paralel sinyal işleme ve dağıtılmış bilgi işleme özelliğine sahiptir. Bu nedenle YSA oldukça yoğun matematiksel işlemler içeren bir yapıdır. Donanım-tabanlı YSA uygulamaları birçok farklı platformlarda gerçekleştirilebilmektedir. Bu platformlardan birisi olan FPGA (Alan Programlanabilir Kapı Dizisi) çipleri paralel işlem yapma kapasitesine ve yüksek çalışma hızına sahiptir. Bu çalışmada gerçek zamanlı YSA uygulamaları için FPGA çipleri üzerinde 6 farklı doğrusal aktivasyon fonksiyonu gerçekleştirilerek bir doğrusal aktivasyon fonksiyonları kütüphanesi oluşturulmuştur. Tasarımlar 32-bit (16I-16Q) IQ-Math sayı standardına uygun olarak VHDL (Çok Yüksek Hızlı Tümlüşik Devre Donanım Tanımlama Dili) dilinde kodlanmıştır. Tüm tasarımlar Xilinx ISE Design Suite programı ile test edilmiştir. Test aşamasının ardından tasarımlar Xilinx Kintex-7 FPGA çipi için sentezlenmiştir. FPGA-tabanlı transfer fonksiyonlarından elde edilen çip istatistikleri ve performans analizleri sunulmuştur.

Anahtar Kelimeler: Yapay Sinir Ağları, Transfer fonksiyonları, IQ-Math sayı standardı, FPGA.

1. GİRİŞ

İnsan beyni mükemmel şekilde çalışan bir sistemdir. Bu sistemin çalışma mükemmelliğini anlamak ve bundan faydalanmak üzerine uzun yıllar çeşitli çalışmalar yapılmıştır. İnsan beyninin çalışması referans alınarak oluşturulan matematiksel modelden ortaya çıkan bilim alanı YSA olarak tanımlanmaktadır. 1943 yılında bir sinir hekimi olan Warren McCulloch ve bir matematikçi olan Walter Pitts ilk YSA modelini gerçekleştirmişlerdir [1]. Bu modelde insan beyninin hesaplama yeteneği referans alınarak basit bir sinir ağı elektrik devreleri ile modellenmiştir. Böylece ilk yapay sinir ağı hücresinin yapısı oluşturulmuş ve yapay sinir ağlarının temeli atılmıştır. McCulloch ve Pitts tarafından modellenen yapay sinir hücreleri ile her türlü mantıksal ifadenin formülize edilebileceğini göstermişlerdir. 1949 yılında Hebb, geliştirdiği Hebb Kuralı ile öğrenebilen ve uyum sağlayabilen sinir ağı modeli için temel oluşturmuştur. Hebb Kuralı sinir ağının bağlantı sayısı değiştirilebildiği takdirde öğrenebileceğini savunmaktadır [2]. 1950'li yıllar sonrasında Hebb kuralı referans alınarak YSA'nın hesaplama gücünü arttırmaya yönelik çalışmalar yapılmıştır. 1957 yılında YSA alanındaki gelişmeler Algılayıcının (Perceptron) Frank Rosentblatt tarafından gerçekleştirilmesi ile hızlandırılmıştır [3]. 1959 yılında ADALINE ve MADALINE ağ modelleri Bernard Widrow ve Marcian Hoff tarafından gerçekleştirilmiştir [4]. MADALINE, telefon hatlarında meydana gelen yankıları yok etmede kullanılarak dünyadaki evrensel sorunlara uygulanan ilk sinir ağıdır ve halen de kullanılmaktadır [5]. Bu başarılı çalışmalar sonucunda insanlar sinir ağları hakkında yüksek beklentiler içerisine girmişlerdir. Dolayısıyla sinir ağları üzerinde yapılan araştırmalar bir duraklama dönemine girmiştir. Bu duraklama dönemi 1981 yılında sona ermiştir [6]. Bu yıllardan günümüze kadar YSA üzerine çalışmalar yapılmaya devam edilmiş ve günümüzde yaygın bir şekilde birçok farklı alandaki akademik çalışmalara konu olmuştur. Bu çalışmalarda YSA yazılım ve donanım tabanlı olarak gerçekleştirilmektedir.

Literatürde YSA ve donanım-tabanlı uygulanması ile ilgili pek çok akademik çalışma yapılmıştır. Himavathi ve arkadaşları, YSA'nın FPGA uygulaması için yeni bir teknik önermiştir. FPGA üzerinde sadece ağı en büyük katmanı uygulanarak bu katman diğer katmanlar için tekrar kullanılmıştır. Sonuçta, bu teknik ile YSA'nın FPGA çiplerindeki donanım maliyetleri etkin bir şekilde düşürülmüştür [7]. Şahin ve arkadaşları, FPGA çipinde 32-bit IEEE 754-1985 kayan noktalı sayı standardını kullanarak RadBas, LogSig ve TanSig aktivasyon fonksiyonları ile YSA nöronlarının tasarımını ve uygulamasını gerçekleştirmişlerdir. Sonuç olarak, FPGA çiplerinde 10 veya daha fazla nöron içeren YSA'nın kolayca uygulanabileceği gösterilmiştir [8]. Çavuşlu ve arkadaşları, FPGA üzerinde parçacık sürü optimizasyonu algoritması (PSO) ile YSA eğitimini donanımsal olarak uygulamışlardır. Tasarım Altera firmasına ait EP2C35F672C6 FPGA çipinde gerçekleştirilmiştir [9]. Adetiba ve arkadaşları, çalışmalarında FPGA çipinde Çok Katmanlı Perceptron Yapay Sinir Ağı (Multi-Layer Perceptron-ANN) nöronunu donanımsal olarak gerçekleştirerek TanSig ve LogSig aktivasyon fonksiyonları ile bu nöron test edilmiştir. Elde edilen sonuçlara göre, TanSig aktivasyon fonksiyonunun önerilen nöronun donanım uygulaması için daha iyi bir sigmoid aktivasyon fonksiyonu olduğu gözlenmiştir [10]. Alçın ve arkadaşları, FPGA'de Pehlivan-Uyaroglu Kaotik Sistemini (PUKS) modelleyebilmek için yeni bir YSA donanım uygulaması geliştirmişlerdir. Çalışmada elde edilen sonuçlara göre, kaotik sistemlerin FPGA çiplerinde YSA kullanılarak başarılı bir şekilde modellenebileceği gösterilmiştir [11]. Hajduk yaptığı çalışmada YSA için yüksek hassasiyetli FPGA aktivasyon fonksiyonu uygulamasını gerçekleştirmiştir. Çalışma hiperbolik tanjant ve sigmoid aktivasyon fonksiyonlarının FPGA'de uygulanmasını sunmaktadır. Sunulan çalışmada, e^x üstel fonksiyonunun gerçekleştirilmesi için McLaurin serisi ve Padé polinomları kullanılmış ve maksimum mutlak hata karşılaştırıcısı yapılmıştır [12]. Nambiar ve arkadaşları, geliştirilebilir blok tabanlı sinir ağlarının (Block-based Neural Networks (BbNNs)) donanım uygulamasını FPGA ortamında gerçekleştirmiştir. Çalışmada yeni ve düşük maliyetli sigmoid benzeri bir aktivasyon fonksiyonu kullanılmıştır. BbNNs, çip üzerinde bir sistem (System on Chip (SoC)) olarak tasarlanmış ve çeşitli durum çalışmalarında işlevsel olarak doğrulanarak test edilmiştir. Çalışmanın sistem performansı gerçek zamanlı sınıflandırma için yeterli olmakta ve gömülü yazılımdan 410 kata kadar daha hızlı çalışmaktadır [13]. Yukarıda literatürde sunulan YSA uygulamaları Uygulamaya Özel Tümlük Devre (Application Specific Integrated Circuit (ASIC)), Sayısal Sinyal İşlemci (Digital Signal

Processor (DSP)), Grafik İşlemci Ünitesi (Graphic Processor Unit (GPU)) ve FPGA gibi platformlar kullanılarak gerçekleştirilmektedir. Bu platformlardan birisi olan FPGA çipleri tekrar tekrar programlanabilmesi, paralel çalışma, düşük güç tüketimi ve hızlı ilk prototipleme gibi özellikleri sayesinde diğer platformlara göre önemli avantajlar sağlamaktadır. Bu çalışma, FPGA çiplerinde PosLin, HardLim, HardLims, SatLin, SatLins, TriBas doğrusal aktivasyon fonksiyonların 32-bit (16I-16Q) IQ-Math sayı standardında gerçekleştirilmesini sunmaktadır. Tasarımlar VHDL dili kullanılarak kodlanarak Xilinx ISE Design Suite programı ile test edilmiştir. Daha sonra tüm tasarımlar Xilinx Kintex-7 FPGA çipi için sentezlenmiştir. Tasarımlardan elde edilen FPGA çip istatistikleri ve performans analizleri sunulmuştur. Bu çalışmada ikinci bölümde, YSA ve FPGA çipleri hakkında kısaca bilgiler verilmiştir. Üçüncü bölümde, FPGA tabanlı aktivasyon fonksiyonları ve yapıları sunulmuştur. Dördüncü bölümde, FPGA tabanlı tasarımların Xilinx ISE Design Suite ortamındaki test sonuçları ve çip istatistikleri verilmiştir. Son bölümde ise çalışmalardan elde edilen sonuçlar üzerinde değerlendirme yapılmıştır.

2. GENEL BİLGİLER

2.1 Yapay Sinir Ağları

Yapay sinir ağları, insan beyninin çalışma yapısı referans alınarak matematiksel modellenmesinden meydana gelen ağ yapılarıdır. Bu sinir ağları elektrokimyasal yöntem ile çalışan sinir hücrelerine göre çok daha hızlı çalışabilmekte ve odaklanmış olduğu problemin çözümü için oldukça başarılı sonuçlar üretebilmektedir. Günümüzde YSA elektrik motorları [14], kontrol [15], tıbbi [16], optimizasyon [17], sinyal ve görüntü işleme [18], tahmin [19, 20], karar verme, sınıflandırma [21] ve kaotik osilatör tasarımı gibi alanlarda yaygın olarak kullanılmaktadır.

YSA yapısı giriş katmanı, gizli katman ya da katmanlardan ve çıkış katmanından oluşmaktadır. Gizli katmanda bulunan katman sayısı ve gizli katmandaki hücre sayısı tasarlanan ağ yapısına göre farklılık göstermektedir. Hücre sayısının artması hesaplama karmaşıklığına ve ağıın sonuç üretme süresinin artmasına sebep olmasına rağmen ağıın genelleme özelliği kazanması için önemli bir etkidir [22]. YSA'da dış ortamdan veya diğer hücrelerden alınan girdiler ağırlıklar ile çarpılmakta ve toplama fonksiyonu yardımı ile net girdi hesaplanmaktadır. Net girdinin aktivasyon fonksiyonundan geçirilmesi ile net çıktı hesaplanmaktadır. YSA'da aktivasyon fonksiyonlarının kullanılmasının temel nedeni çıkış sinyalinin genlik aralığını bazı sonlu değerlerle sınırlandırmaktır [23]. Aktivasyon fonksiyonları sonuca direkt etki ederek doğrusal ve doğrusal olmayan formlarda yaygın olarak kullanılmaktadır.

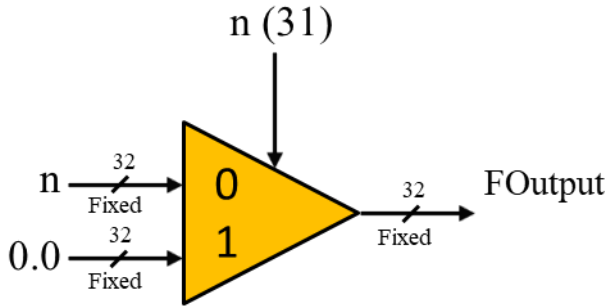
2.2 FPGA

FPGA çipleri günümüzde yaygın olarak kullanılan entegre devrelerdir. FPGA çipleri farklı firmalar (Xilinx, Altera, Actel vb.) tarafından üretilmektedir. Bu firmalar arasında Xilinx firması günümüzde çoğunlukla tercih edilmektedir. Üretildikleri firmalara göre FPGA'ler farklılık göstermekte ve kodlama dilleri (VHDL, Verilog vb.) ile programlanarak simüle edilmektedir. Simülasyon sonucunda sistemden beklenen sonuçlar elde edildikten sonra sentez aşamasına geçilmektedir. Sentezleyici ile kullanıcının tasarladığı sistem seçilen FPGA elemanına göre sentezlenmektedir. Sentezleme yapıldıktan sonra yerleştirme ve yönlendirme işlemlerinin yapılması ile FPGA'in programlanması için gerekli olan bit dizisi üretilmektedir. Son olarak gerçekleştirme işleminin tamamlanması için FPGA uygun donanımlar kullanılarak programlanmaktadır [24]. FPGA çipleri defalarca programlanabilme, paralel işlem yapabilme gibi üstünlüklere sahiptir. Bu üstünlükleri sayesinde günümüzde sinyal ve görüntü işleme [25], motor kontrolü [26], birçok algoritmaların hızlandırılması [27], yapay sinir ağları [28], güvenli iletişim [29], kaotik osilatör tasarımı [30], hidrojen üretim sistemleri [31], rasgele sayı üretimi [32] ve robotik sistemler [33] gibi birçok alanda yaygın olarak kullanılmaktadır.

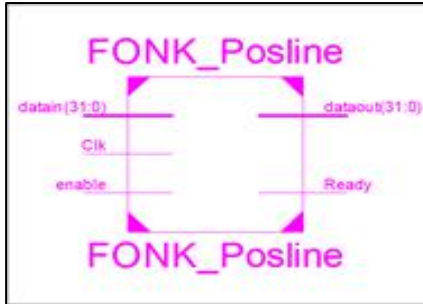
3. AKTİVASYON FONKSİYONLARININ FPGA-TABANLI TASARIMLARI

Bu çalışmada, PosLin, HardLim, HardLims, SatLin, SatLins, TriBas fonksiyonları FPGA çipleri için VHDL dili kullanılarak 32 bit (16I-16Q) IQ-Math sayı standardında kodlanmıştır. Tasarımlar Xilinx ISE Design Suite 14.7 ortamında test edilmiştir. Daha sonra tüm tasarımlar Xilinx firmasına ait Kintex-7 FPGA çipi için sentezlenmiştir. Denklem 1’de PosLin fonksiyonuna ait denklem verilmiştir. Şekil 1’de PosLin fonksiyonun 32 bit IQ-Math sayı standardına uygun olarak tasarlanmasına ait blok şeması görülmektedir. Bu şemada 32 bit n giriş sinyalinin işaret biti lojik 0 ya da 1 olması durumuna göre çıkıştan n ya da 0 çıktı sinyali alınmaktadır. Şekil 2’de PosLin fonksiyonun FPGA en üst seviye blok diyagramı verilmiştir.

$$\text{PosLin} = \begin{cases} n, & \text{if } n \geq 0 \\ 0, & \text{if } n \leq 0 \end{cases} \quad (1)$$



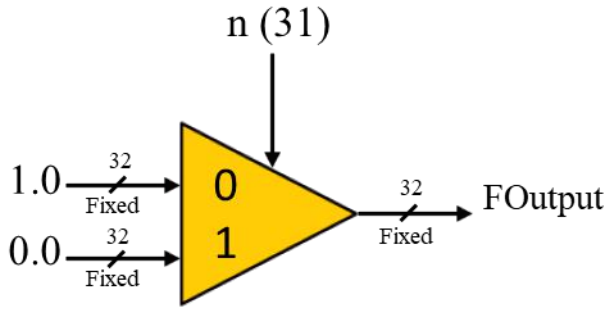
Şekil 1. PosLin fonksiyonun blok şeması



Şekil 2. PosLin fonksiyonun en üst seviye blok diyagramı

Denklem 2’de HardLim fonksiyonuna ait denklem verilmiştir. Şekil 3’de HardLim fonksiyonun 32 bit IQ-Math sayı standardına uygun olarak tasarlanmasına ait blok şeması görülmektedir.

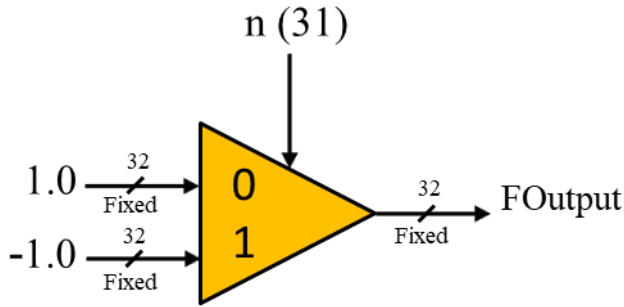
$$\text{HardLim} = \begin{cases} 1, & \text{if } n \geq 0 \\ 0, & \text{in other cases} \end{cases} \quad (2)$$



Şekil 3. HardLim fonksiyonun blok şeması

Denklem 3’de HardLims fonksiyonuna ait denklem verilmiştir. Şekil 4’de HardLims fonksiyonun 32 bit IQ-Math sayı standardına uygun olarak tasarlanmasına ait blok şeması görülmektedir.

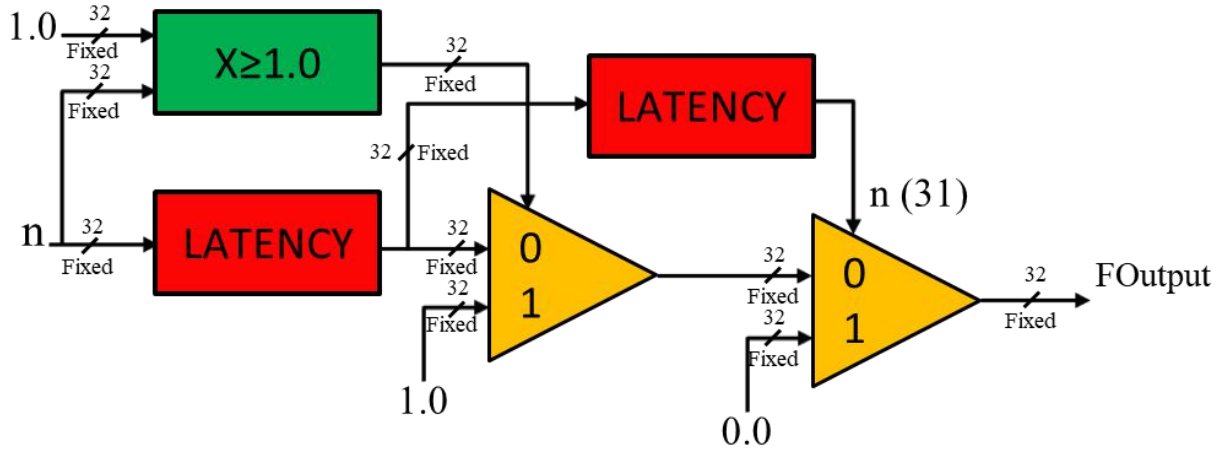
$$\text{HardLims} = \begin{cases} 1, & \text{if } n \geq 0 \\ -1, & \text{in other cases} \end{cases} \quad (3)$$



Şekil 4. HardLims fonksiyonun blok şeması

Denklem 4’de SatLin fonksiyonuna ait denklem verilmiştir. Şekil 5’de SatLin fonksiyonun 32 bit IQ-Math sayı standardına uygun olarak tasarlanmasına ait blok şeması görülmektedir. Bu şemada giriş sinyalinin 1’den küçük, büyük ya da eşit olması durumları karşılaştırıcı ünitesinde değerlendirilmektedir. Eğer giriş sinyali 1’den büyük ya da eşit ise lojik 1, diğer durumlarda lojik 0 üretilmektedir. Bu şemada tasarımın paralel çalışması için 2 adet gecikme ünitesi kullanılmıştır. Burada giriş sinyali karşılaştırıcı ünitesinin işlem süresi kadar geciktirilmiştir. Daha sonra gecikme ünitesinden çıkan sinyale multiplexer ünitesinde gerçekleşen işlem süresi kadar gecikme uygulanmıştır.

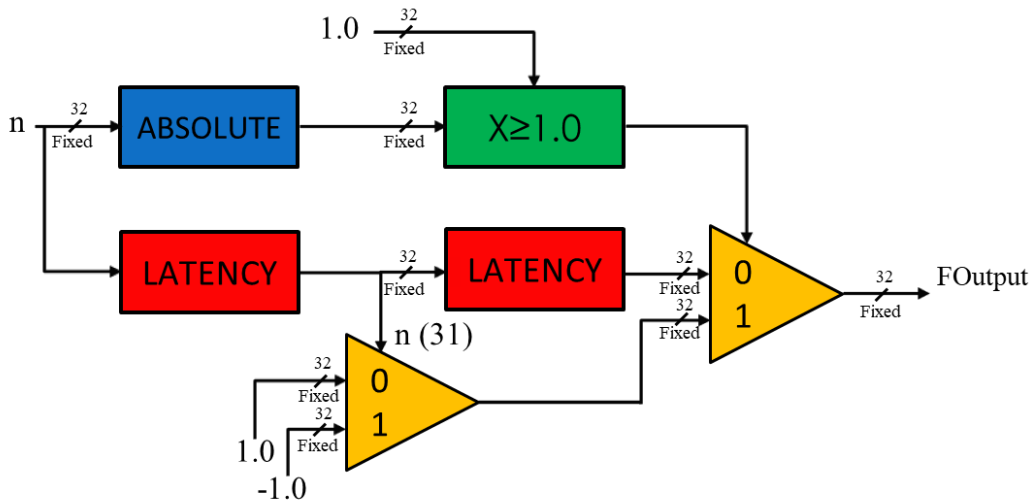
$$\text{SatLin} = \begin{cases} 0, & \text{if } n \leq 0 \\ n, & \text{if } 0 \leq n \leq 1 \\ 1, & \text{if } n \geq 1 \end{cases} \quad (4)$$



Şekil 5. SatLin fonksiyonun blok şeması

Denklem 5’de SatLins fonksiyonuna ait denklem verilmiştir. Şekil 6’da SatLins fonksiyonun 32 bit IQ-Math sayı standardına uygun olarak tasarlanmasına ait blok şeması görülmektedir. Bu şemada giriş sinyali mutlak değer ünitesinde uygulanan işlem süresi kadar geciktirilmiştir. Daha sonra bu sinyalin işaret biti multiplexer ünitesinde seçici sinyal olarak kullanılmıştır. Aynı zaman diliminde bu sinyal karşılaştırıcı ünitesinin işlem süresi kadar geciktirilerek diğer multiplexer ünitesine gönderilmiştir.

$$\text{SatLins} = \begin{cases} -1, & \text{if } n \leq -1 \\ n, & \text{if } -1 \leq n \leq 1 \\ 1, & \text{if } n \geq 1 \end{cases} \quad (5)$$

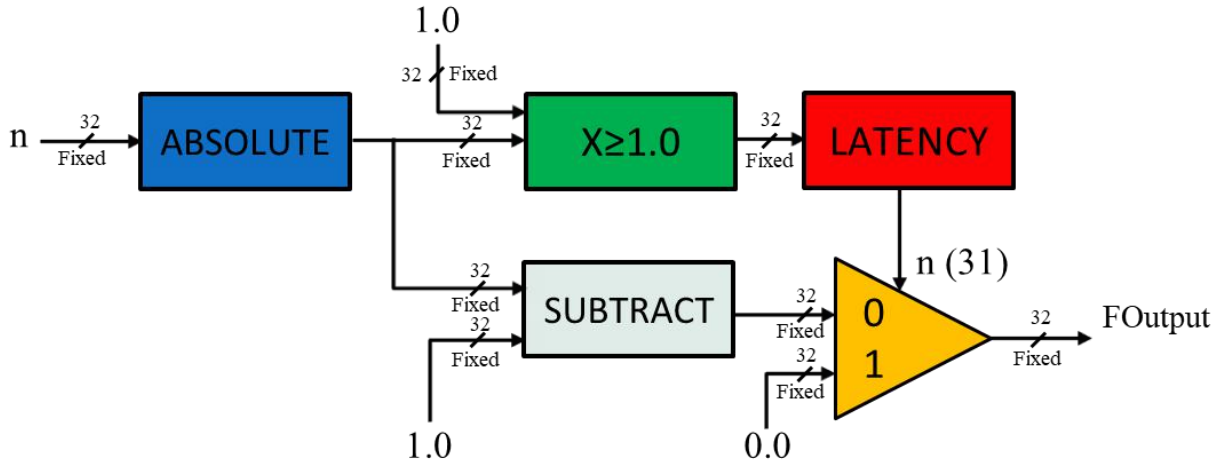


Şekil 6. SatLins fonksiyonun blok şeması

Denklem 6’da TriBas fonksiyonuna ait denklem verilmiştir. Şekil 7’de SatLins fonksiyonun 32 bit IQ-Math sayı standardına uygun olarak tasarlanmasına ait blok şeması görülmektedir. Bu şemada çıkarıcı ünitesi 5 saat darbesi sonucunda sonuç üretmektedir. Bu yüzden karşılaştırıcı ünitesinin çıkış sinyaline

gecikme uygulanmıştır. Böylece çıkarıcı ve karşılaştırıcı ünitelerinin çıkış sinyalleri eş zamanlı olarak multiplexer ünitesine iletilmiştir.

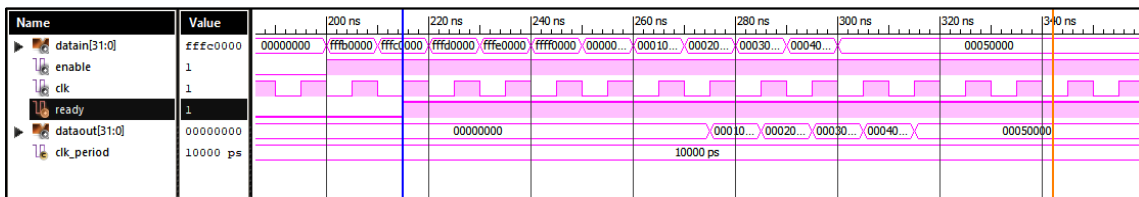
$$\text{TriBas} = \begin{cases} 1 - \text{abs}(n), & \text{if } -1 \leq n \leq 1 \\ 0, & \text{in other cases} \end{cases} \quad (6)$$



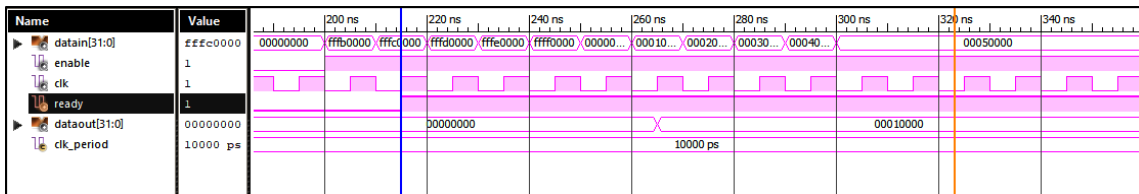
Şekil 7. TriBas fonksiyonun blok şeması

4. FPGA-TABANLI AKTİVASYON FONKSİYONLARIN TEST SONUÇLARI

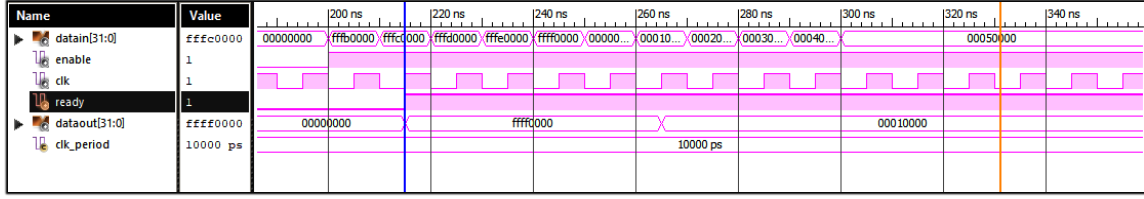
FPGA'lerden üretilen sonuçların doğrululuğunu test etmek için çeşitli programlar kullanılmaktadır. Bu çalışmada, tüm tasarımlar Xilinx ISE Design Suite 14.7 programında (-5,5) aralığı için test edilerek simülasyon gerçekleştirilmiştir. Şekil 8'de PosLin, Şekil 9'da HardLim, Şekil 10'da HardLims, Şekil 11'de SatLin, Şekil 12'de SatLins ve Şekil 13'de TriBas fonksiyonlarının Xilinx ISE Design Suite 14.7 ortamındaki simülasyon sonuçları verilmiştir.



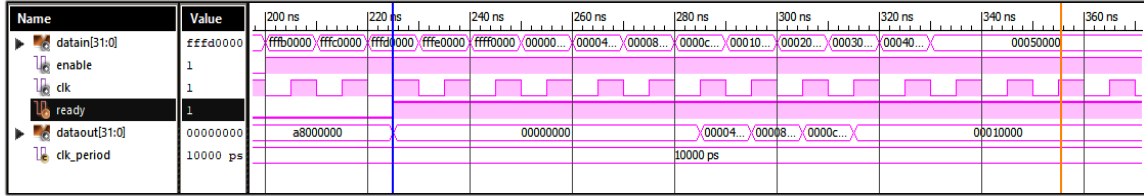
Şekil 8. PosLin fonksiyonun Xilinx ISE 14.7 simülasyon sonuçları



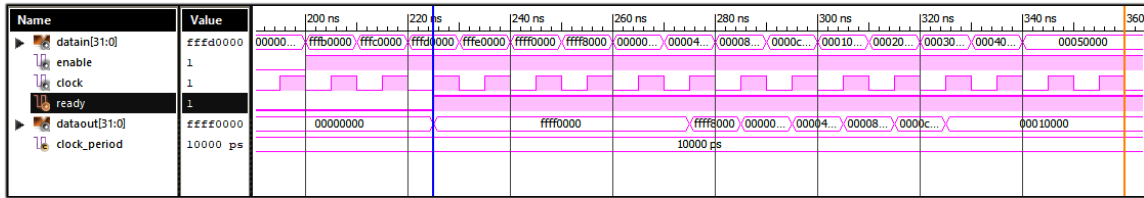
Şekil 9. HardLim fonksiyonun Xilinx ISE 14.7 simülasyon sonuçları



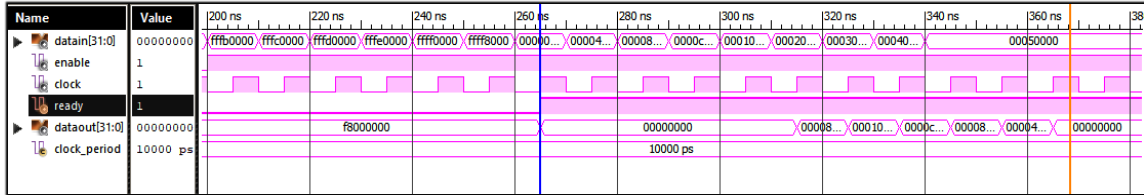
Şekil 10. HardLims fonksiyonun Xilinx ISE 14.7 simülasyon sonuçları



Şekil 11. SatLin fonksiyonun Xilinx ISE 14.7 simülasyon sonuçları



Şekil 12. SatLins fonksiyonun Xilinx ISE 14.7 simülasyon sonuçları



Şekil 13. TriBas fonksiyonun Xilinx ISE 14.7 simülasyon sonuçları

Xilinx ISE Design Suite 14.7 ortamında tüm tasarımların test işleminden beklenen sonuçlar alındıktan sonra tasarımlar Xilinx firmasına ait Kintex-7 FPGA çipi için sentezlenmiştir. Tablo 1’de 6 farklı aktivasyon fonksiyonun FPGA çip istatistikleri gösterilmektedir.

Tablo 1. Doğrusal Aktivasyon Fonksiyonların FPGA Çip İstatistikleri

FPGA Çip İstatistikleri	PosLin	HardLim	HardLims	SatLin	SatLins	TriBas
Mak. Çalışma Frekans (MHz)	956.023	956.023	956.023	1379.172	825.355	822.504
Slice Registers Sayısı	63	3	3	102	86	151
Slice LUTs Sayısı	2	2	2	68	116	206

IOBs Sayısı	67	36	36	67	67	67
BUFG/BUFGCTLs Sayısı	1	1	1	1	1	1

5. SONUÇLAR

Bu çalışmada 6 farklı (PosLin, HardLim, HardLims, SatLin, SatLins, TriBas) doğrusal aktivasyon fonksiyonu FPGA çiplerinde gerçekleştirilmiştir. Tasarımlar 32-bit IQ-Math sayı standardına uygun olarak VHDL dilinde kodlanmıştır. Bu tasarımlar Xilinx ISE 14.7 programı kullanılarak test edilmiş ve istenen sonuçlar elde edilmiştir. Tüm tasarımlar Kintex-7 FPGA çipi için sentezlenmiş ve FPGA çip istatistikleri sunulmuştur. Böylece 6 farklı doğrusal aktivasyon fonksiyonunun FPGA çiplerinde başarılı bir şekilde gerçekleştirilebileceği gösterilmiştir.

TEŞEKKÜR

Bu çalışma 19.FEN.BİL.14 proje numarası ile Afyon Kocatepe Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi tarafından desteklenmiştir.

KAYNAKLAR

- [1] McCulloch, W. S., & Pitts, W. A., A logical calculus of the ideas immanent in nervous activity, *Buttetin of Mathematics and Biophysics*, 5, 115-133, 1943.
- [2] Hebb, D. O., The organization of behaviour. The first stage of perception: growth of the assembly, 4, 60-78, 1949.
- [3] Rosenblatt, F., The perceptron: A probabilistic model for information storage and organization in the brain. *Psychoanalytic Review*, 65, 386-408, 1958.
- [4] Widrow, B., & Hoff, M. E., Adaptive switching circuits. WESTCON Convention, Record Part IV, 96-10, 1960.
- [5] Keskenler MF., Keskenler EF., Geçmişten Günümüze Yapay Sinir Ağları ve Tarihçesi, *Takvim-i Vekayi*, Cilt.5, No. 2, Sayfa. 8-18, 2017.
- [6] Ç. Elmas, “Yapay zeka uygulamaları”, Ankara, Seçkin Yayıncılık, 2018.
- [7] S. Himavathi, D. Anithaand A. Muthuramalingam, Feedforward neural network implementation in FPGA using layer multiplexing for effective resource utilization, *IEEE Trans. Neural Netw.* 18, 880–888, 2007.
- [8] M. Alçın, “FPGA üzerinde YSA tabanlı gerçek rasgele sayı üreticinin tasarımı ve gerçekleşmesi”, *Sakarya- Türkiye: Sakarya Üniversitesi (Doktora Tezi)*, 2017.
- [9] M. A. Çavuşlu, C. Karakuzu ve S. Şahin parçacık sürü optimizasyonu algoritması ile yapay sinir ağı eğitiminin FPGA üzerinde donanımsal gerçekleşmesi, *Politeknik Dergisi* cilt 13, sayı 2, s. 83-92, 2010.
- [10] Adetiba, E., Ibikunle, F., Daramola, S., ve Olajide, A., Implementation of Efficient Multilayer Perceptron ANN Neurons on Field Programmable Gate Array Chip, *International Journal of Engineering & Technology IJET-IJENS*, Vol:14, No:01, 2014.
- [11] M. Alcin, I. Pehlivan, I. Koyuncu, Hardware design and implementation of a novel ANN-based chaotic generator in FPGA, *Optik*, Vol:127, No:13, 2016.
- [12] Z. Hajduk, High accuracy FPGA activation function implementation of neural networks,

- Neurocomputing, 247, 59-61, 2017.
- [13] V.P. Nambiar, M. Khalil-Hani, R. Sahnoun, M.N. Marsono, Hardware implementation of evolvable block-based neural networks utilizing a cost efficient sigmoid-like activation function, *Neurocomputing*, 140, 228-241, 2014.
- [14] X. Yang, J. Cao and D. W. Ho, Exponential synchronization of discontinuous neural networks with time varying mixed delays via state feedback and impulsive control, *Cogn. Neurodyn.*, 9, 113–128, 2014.
- [15] J. Fei and H. Ding, Adaptive sliding mode control of dynamic system using RBF neural network, *Nonlinear Dyn.*, 70, 1563–1573, 2012.
- [16] D. Avci, M. K. Leblebicioglu, M. Poyraz and E. Dogantekin, A new method based on adaptive discrete wavelet entropy energy and neural network classifier (ADWEENN) for recognition of urine cells from microscopic images independent of rotation and scaling, *J. Med. Syst.*, 38, 1–9, 2014.
- [17] C. J. Lin and H. M. Tsai, FPGA implementation of a wavelet neural network with particle swarm optimization learning, *Math. Comput. Model.*, 47, 982–996, 2008.
- [18] Koyuncu I., Sahin I., Design and Implementation of Neural Networks Neurons with RadBas, LogSig, and TanSig Activation Functions on FPGA, *Electronics and Electrical Engineering*, Kaunas: Technologija, No. 4(120), P. 51–54, 2012.
- [19] H. Papadopoulos and H. Haralambous, Reliable prediction intervals with regression neural networks, *Neural Netw.*, 24, 842–851, 2011.
- [20] M. Kanayama, A. Rohe and L. A. Paassen, Using and improving neural network models for ground settlement prediction, *Geotech. Geol. Eng.*, 32, 687–697, 2014.
- [21] İ. Şahin, G. Temür, Yapay sinir ağlarının otomatik olarak FPGA çipine uygulanması için otomatik denetleyici tasarım aracı, *SDU International Journal of Technological Science*, vol.8, no.1, pp. 34-52, 2016.
- [22] Ersoy E., Karal Ö., Yapay sinir ağları ve insan beyni, *Journal of the Human and Social Science Researches*, Volume.1, Issue: 2, 2012.
- [23] I. Koyuncu, I. Sahin, C. Gloster, N. K. Saritekin, A neuron library for rapid realization of artificial neural networks on FPGA: A case study of rössler chaotic system, *Journal of Circuits, Systems, and Computers*, vol. 26, no. 1, 1750015 (21 pages), 2017.
- [24] İ. H. Topçu, Sahada Programlanabilir Kapı Dizileri Kullanılarak Sayısal Tasarım Kartı Gerçeklenmesi, İstanbul: İstanbul Teknik Üniversitesi, 2002.
- [25] Paukštaitis, V. and Dosinas A., Pulsed Neural Networks for Image Processing. *Electronics and Electrical Eng.*, No. 7(95), 2009.
- [26] Monmasson, E., Idkhajine, L., Cirstea, M.N., Bahri, I., Tisan, A., and Naouar, M.W., FPGAs in Industrial Control Applications. *Industrial Informatics*, *IEEE Trans. on*, vol. 7, no.2, pp.224-243, 2011.
- [27] Sahin I., A 32-bit floating-point module design for 3D graphic transformations. *SRE*, vol. 5(20), pp. 3070-3081, 2010.
- [28] Song, Y. and Gao, L., Incremental Battery Model Using Wavelet-Based Neural Networks. Components, Packaging and Manufacturing Technology, *IEEE Trans. on*, vol.1, no.7, pp. 1075-1081, 2011.
- [29] Tuna M., Alçın M., Koyuncu İ., Fidan CM., Pehlivan İ., High speed FPGA-based chaotic oscillator design, *Microprocessors and Microsystems*, Vol. 66, pp. 72–80, 2019.
- [30] Koyuncu, İ., & Şeker, H. İ., Implementation of Dormand-Prince based chaotic oscillator designs in different IQ-Math number standards on FPGA. *Sakarya University Journal of Science*, 23(5),

859-868, 2019.

- [31] Yilmaz, C., Koyuncu, I., Alcin, M., & Tuna, M., Artificial Neural Networks based thermodynamic and economic analysis of a hydrogen production system assisted by geothermal energy on Field Programmable Gate Array. *International Journal of Hydrogen Energy*, 2019.
- [32] Alcin, M., Koyuncu, I., Tuna, M., Varan, M., & Pehlivan, I., A novel high speed Artificial Neural Network-based chaotic True Random Number Generator on Field Programmable Gate Array. *International Journal of Circuit Theory and Applications*, 47(3), 365-378, 2019.
- [33] Tuntas R., A new intelligent hardware implementation based on field programmable gate array for chaotic systems, *Applied Soft Computing*, Vol. 35, pp. 237-246, 2015.