

139 739

U.S. GOVERNMENT PRINTING OFFICE
1964 O - 348-000

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ	v
KISALTMA LİSTESİ	vi
ŞEKİL LİSTESİ	vii
ÇİZELGE LİSTESİ	ix
ÖNSÖZ	x
ÖZET	xi
ABSTRACT	xii
1. GİRİŞ	1
1.1 Önceki Çalışmalar ve Teknolojinin Bugün Geldiği Nokta	1
1.2 Çalışmanın Amacı ve Kapsamı	8
2. YAPAY SİNİR AĞI HAKKINDA GENEL BİLGİLER	9
2.1 Biyolojik Sinir Ağları	9
2.1.1 Biyolojik Sinir Hücresinin Anatomisi	10
2.1.2 Biyolojik Sinir Hücresinin Fizyolojisi	11
2.2 Yapay Sinir Ağının Yapısı	12
2.2.1 Giriş İşareti Sınıfları	14
2.2.2 Bağlantı Geometripleri	14
2.2.3 Ağ Tipleri	15
2.2.4 Aktivasyon Fonksiyonları	15
2.2.5 Yapay Sinir Ağlarının Kullanım Nedenleri	16
2.3 Eğitim Algoritmaları	19
2.4 Öğrenme Kuralları	20
2.4.1 YSA Konfigürasyonu	20
2.5 Çok Katmanlı Algılayıcı: ÇKA (Multi Layer Perceptron: MLP)	22
2.6 Geriye Yayılma Algoritması: GYA (Back Propagation Algorithm: BPA)	23
2.6.1 Öğrenme ve Momentum Katsayıları	23
2.7 Yapay Sinir Ağlarının Elektronik Devre Olarak Gerçeklenmesi	24
2.7.1 Giriş	24
2.7.2 Belli Başlı Tasarım Öğeleri	25
2.7.3 VLSI Uygulamalarının Kategorileri	27
2.7.4 YSA Donanımlarına Kısa Bir Bakış	30
3. FGMOS TRANSİSTORLAR	31
3.1 Tarihsel Gelişim	31
3.2 Biyolojik Nöron İle Yüzen Geçitli-Çok Girişli MOS Transistorun İşlevsel Benzerliği	32

3.3	N-Kanallı FGMOS Transistor (vMOS) Yapısı	32
3.4	FGMOS Transistorun Çalışma İlkesi ve Analitik Bağlıntılar	33
3.4.1	Doymada Çalışan n-kanallı FGMOS Transistorda Kanal Yükünün Yüzen Geçit Gerilimi Cinsinden İfadesi	35
3.4.2	Doymada Çalışan n-kanallı FGMOS Transistorun Akım Bağlıntısı	36
3.5	FGMOS Makro Modeli	38
3.5.1	FGMOS Transistor Simülasyonu İçin Makro Modelin Gerekliliği	38
3.5.2	Ochiai-Hatano Makro Modeli	38
3.5.3	(3.13) Denkleminin Dayalı Bir Makro Modelin Oluşturulması	40
3.6	Dört-Uçlu Elemanların Esnek Yapısı, İki-Uçlu Ve Üç-Uçlu Elemanlarla Karşılaştırılması	43
3.7	FGMOS Transistorun Kullanım Alanları ve Avantajları	45
3.8	İki Girişli N-Kanallı FGMOS Transistorun Simülasyonla Elde Edilen Özgeçirileri	45
4.	FGMOS NÖRONUN TASARIMI	47
4.1	Farksal Gerilim Zayıflatıcı	47
4.1.1	Farksal Gerilim Zayıflatıcısının Yapısı	48
4.1.2	İkincil Etkiler ve Eleman veya Parametre Dengesizliklerinin İncelenmesi	50
4.1.2.1	Kanal Kısılması (λ)	50
4.1.2.2	Hareket Yeteneğinin Düşey Elektrik Alan İle Değişimi (θ)	51
4.1.2.3	Girişlere Ait Kapasitif Etki Oranlarının Dengesizliği	51
4.1.2.4	Geçiş İletkenliği Dengesizliği	52
4.1.2.5	Eşik Gerilimi Dengesizliği	52
4.2	Zayıflatıcıları Kullanarak Dengeli Girişli Bir Devrenin Tek Ucu Sonlandırılmış Haline Dönüştürülmesi	53
4.3	Kare Alıcı Devre	54
4.4	FGMOS Transistorlarla Kurulmuş Dört Bölge Analog Çarpma Devresi	57
4.5	Zayıflatıcı, Kare Alıcı ve Çarpıcıya Ait Simülasyon Sonuçları	58
4.6	En Temel Nöron Yapısı	62
4.6.1	Farksal Karşılaştırıcı Yapısı	63
4.6.2	Temel Nöronun (Bir Girişli Nöron) Gerçeklenmesi	64
4.7	İki Girişli Nöron Yapısı	65
4.7.1	FGMOS Farksal Karşılaştırıcı	65
4.7.2	İki Çarpıcı ve Bir FGMOS Farksal Karşılaştırıcı Kullanılarak İki-Girişli Bir Nöronun Gerçeklenmesi	67
4.8	N-Girişli Nöron Tasarımı	68
4.9	Yapay Sinir Ağı Devresi Tasarımı	69
4.10	Uygulamalar	70
4.10.1	YSA'da Çok Bilinen XOR Problemi	70
4.10.1.1	Ağın Eğitilmesi ve Ağ Yapısının Belirlenmesi	71
4.10.1.2	XOR Problemini Gerçekleyen Devre Yapısı ve Simülasyon Sonuçları	72
4.10.2	İris Çiçeğinin Sınıflandırılması	74
4.10.2.1	MATLAB'da Ağın Eğitilmesi ve Ağ Yapısının Belirlenmesi (MLP - BPA)	74
4.10.2.2	İris Sınıflandırıcı ve Simülasyon Sonuçları	79
5.	SONUÇLAR	81
	KAYNAKLAR	83
	İNTERNET KAYNAKLARI	85

EKLER.....	86
Ek 1 İki-girişli nöron devresinin simülasyonunda kullanılan HSPICE giriş dosyası.....	87
Ek 2 XOR'un simülasyonunda kullanılan HSPICE giriş dosyası	89
Ek 3 Iris' e ait orijinal data dosyası	92
Ek 4 Iris'in simülasyonunda kullanılan normalize edilmiş girişlere ait data dosyası	95
Ek 5 Iris'in simülasyonunda kullanılan HSPICE giriş dosyası.....	98
ÖZGEÇMİŞ	102



SİMGE LİSTESİ

C_0	Geçit oksidi kapasitesi
C_{FD}	Yüzen geçit - savak örtüşme kapasitesi
C_{FS}	Yüzen geçit - kaynak örtüşme kapasitesi
C_{FSUB}	Yüzen geçit – aktif olmayan taban örtüşme kapasitesi
C_{ox}	Birim alan başına geçit oksidi kapasitesi
C_T	Yüzen geçide bağlı toplam kapasite
I_D	Savak akımı
I_{SS}	Referans akımı
K_n	NMOS geçiş iletkenliği parametresi
K_p	PMOS geçiş iletkenliği parametresi
k_i	Giriş kapasitif kuplaj ağırlığı
k_{SUB}	Aktif olmayan taban kapasitif kuplaj ağırlığı
k_D	Savak kapasitif kuplaj ağırlığı
k_S	Kaynak kapasitif kuplaj ağırlığı
L	Kanal uzunluğu
q	Birim elektrik yükü
Q_{ch}	Kanal yükü
$Q_n(y)$	Kanalın herhangi bir y noktasındaki birim alan başına yük
Q_T	Başlangıçta yüzen geçitte depolanmış yük
R_O	Çıkış direnci
V_B	Gövde gerilimi
V_D	Savak gerilimi
V_S	Kaynak gerilimi
V_T	Eşik gerilimi
V_{SUB}	Aktif olmayan taban gerilimi
V_{FG}	Yüzen geçit gerilimi
$V_c(y)$	Kanal gerilimi
V_{DD}	Pozitif besleme gerilimi
V_{SS}	Negatif besleme gerilimi
W	Kanal genişliği
β	Geçiş iletkenliği parametresi
μ_o	Gövde hareket yeteneği
μ_n	Kanaldaki ortalama elektron hareket yeteneği
λ	Kanal boyu kısılma parametresi
θ	Hareket yeteneği değişimi parametresi
γ	Gövde etkisi faktörü
η	Öğrenme katsayısı
α	Momentum katsayısı

KISALTMA LİSTESİ

BPA	Back Propagation Algorithm
FGMOS	Floating Gate MOS
MLP	Multi Layer Perceptron
VLSI	Very Large Scale Integrated
XOR	Exlusive OR
YSA	Yapay Sinir Ağı



ŞEKİL LİSTESİ

Şekil 1.1 İnsan beyni elektronik entegrasyonu [2]	3
Şekil 2.1 Biyolojik sinir hücresinin (nöron) basitleştirilmiş anatomik yapısı.....	11
Şekil 2.2 İşlem elemanı yapısı.....	13
Şekil 2.3 YSA sınıflandırıcıları	14
Şekil 2.4 Kullanılan transfer fonksiyonları	15
Şekil 2.5 Üç boyutlu uzayda ağırlık vektörünün değişimi	21
Şekil 2.6 Çok katmanlı algılayıcı (MLP) yapısı	22
Şekil 3.1 N-girişli n-kanallı FGMOS transistorun basitleştirilmiş kesidi.....	33
Şekil 3.2 N-girişli n-kanallı FGMOS transistorun sembolü ve layout gösterimi	33
Şekil 3.3 N-girişli bir vNFGMOS transistorda yüzen geçide bağlı kapasiteler ve biriktirdikleri yükler	34
Şekil 3.4 2-girişli n-kanallı FGMOS transistorun eşdeğer devresi	38
Şekil 3.5 FGMOS transistorun, Ochiai-Hatano makro modeline temel oluşturan kapasitif modeli (Ochiai, 1999)	38
Şekil 3.6 Ochiai-Hatano n-girişli FGMOS makro modeli (Ochiai, 1999).....	39
Şekil 3.7 (3.13) bağıntısını gerçekleyen makro model	41
Şekil 3.8 Temel elemanlarda iki-uçlu elemandan dört-uçlu elemana doğru işlevselliğin geliştirilmesi (Ohmi, 1997)	44
Şekil 3.9 Bir üç-uçlu ve bir dört-uçlu elemanın (FGMOS) karşılaştırılması (Ohmi, 1997).....	44
Şekil 3.10 2-girişli n-kanallı FGMOS transistorun geçiş özeğrisi	46
Şekil 3.11 2-girişli n-kanallı FGMOS transistorun çıkış özeğrisi	46
Şekil 4.1 Doğrusal gerilim zayıflatıcı yapısı (a) n-kanallı FGMOS'lu yapı, (b) p-kanallı FGMOS'lu yapı, (c) zayıflatıcının sembolü	48
Şekil 4.2 İki tane zayıflatıcı kullanarak dengeli girişin tek ucu sonlandırılmış girişe dönüştürülmesi.....	53
Şekil 4.3 Zayıflatıcıların kullanıldığı tek ucu sonlandırılmış gerilimin karesini alan devre şeması	54
Şekil 4.4 Tek ucu sonlandırılmış girişli dört-bölgeli çarpıcı	57
Şekil 4.5 Gerilim zayıflatıcının DC geçiş karakteristiği: (a) $V_o - V_1$ grafiği, V_2 parametre. (b) $V_o - V_2$ grafiği, V_1 parametre.....	58
Şekil 4.6 Zayıflatıcının transient cevabı (a) giriş işaretleri (b) çıkış gerilimi	59
Şekil 4.7 Kare alıcının DC geçiş karakteristiği	60
Şekil 4.8 Kare alıcının transient cevabı (a) giriş işaretleri (b) çıkış gerilimi	60
Şekil 4.9 Dört-bölgeli çarpıcının DC geçiş karakteristiği.....	61
Şekil 4.10 Dört-bölgeli çarpıcının transient cevabı (genlik modülatörü olarak kullanılması) (a) giriş işaretleri (b) çıkış gerilimi	62
Şekil 4.11 Farksal karşılaştırıcı devresi ve DC geçiş eğrisi	64
Şekil 4.12 (a) Temel nöron yapısı (b) Temel nöron devresinin blok şeması	64
Şekil 4.13 Temel nöron devresinin değişen ağırlık değerlerine göre DC analizi	65
Şekil 4.14 FGMOS farksal karşılaştırıcı devresi.....	66
Şekil 4.15 (a) 2-girişli nöron yapısı, (b) 2-girişli nöron yapısıyla devresi arasındaki geçiş, (c) 2-girişli nöron devresi	67
Şekil 4.16 N-girişli nöron devresi	68
Şekil 4.17 N-girişli bir-çıkışlı basit bir yapay sinir ağı devresi	69
Şekil 4.18 XOR problemini gerçekleştiren ağ yapısı	71
Şekil 4.19 XOR problemini gerçekleyen devre	72
Şekil 4.20 XOR devresinin girişleri ve çıkışı.....	73
Şekil 4.21 Iris sınıflama işlemini gerçekleyen ileri beslemeli YSA.....	75
Şekli 4.22 Ağın eğitilmesinde optimum durum ve ezberleme	76

Şekil 4.23 Matlab programında eğitime işlemi	76
Şekil 4.24 (a) Test datasının dağılım grafiği, (b) test datasının sürekli grafiği.....	77
Şekil 4.25 (a) Eğitime datasının dağılım grafiği, (b) eğitime datasının sürekli grafiği	78
Şekil 4.26 Test datasının simülasyon sonucu.....	79
Şekil 4.27 Eğitime datasının simülasyon sonucu	80
Şekil 4.28 Tüm giriş datasının simülasyon sonucu	80



ÇİZELGE LİSTESİ

Çizelge 2.1 İlk geliştirilen bazı YSA işlemcileri.....	19
Çizelge 4.1 Giriş ve hedef çıkışların devrenin giriş-çıkış aralığına göre ayarlanması	71



ÖNSÖZ

Çalışmam boyunca bana yol gösteren, her zaman olumlu yönde sabırla destek veren ve sürekli fikir alış verişinde bulunduğum değerli Hocam Doç. Dr. Tülay Yıldırım'a, bilgilerinden her an yararlandığım Ar. Gör. Mutlu Avcı'ya, fikirleriyle destekleyen ve her zaman yardımcı olan İTÜ Elektronik Mühendisliği'nden Ar. Gör. Sinem Çiftçioğlu'na, ayrıca yardımı dokunan tüm arkadaşlarıma ve tabiki gece-gündüz her an yanımda olan aileme teşekkürlerimi sunmayı bir borç bilirim.

Ayrıca kullandığım tüm kaynakların yazarlarına da teşekkürlerimi sunarım. Bu tezin de gelecek çalışmalara ışık tutması temennisiyle...



ÖZET

Bu çalışmada, Vlassis ve Siskos'un 2001 yılında önerdikleri (Vlassis, 2001) FGMOS transistörlerle kurulmuş farksal gerilim zayıflatıcı yapısına dayanan bir kare alıcı ve bunun da geliştirilmesiyle beslemeden beslemeye giriş verilebilen ve doğrusallığı çok iyi olan analog dört bölgeli çarpıcı yapısı incelenip tasarlanmış, bu önerilen çarpma devresi ve yine FGMOS transistörlere dayanan farksal karşılaştırıcı yapıları kullanılarak nöron tasarımı gerçekleştirilmiştir. Söz konusu devrenin özgün yanı, FGMOS olarak adlandırılan yüzen geçitli ve çok girişli MOS yapılarını içermesidir. FGMOS transistörlerde yüzen-geçit gerilimi, birden çok sayıdaki giriş işareti ile kutuplanan ikinci poli silisyum bölgelerinin, yüzey bırakılmış birinci poli silisyum tabakasından oluşan geçide kapasitif oranıyla belirlenir. Dolayısıyla her bir giriş geriliminin yüzen-geçit gerilimine etkisi, o girişle yüzen-geçit arasındaki kapasite oranında olacaktır.

Böylece tasarımı gerçekleştirilmiş olan FGMOS nöron yapıları kullanılarak yapay sinir ağları oluşturulabilir hale gelmiştir. MATLAB programıyla yapay sinir ağları eğitilip yapısı belirlenerek ağırlıklar elde edilmiştir. FGMOS nöron yapıları kullanılarak yapay sinir ağlarında çok bilinen problem olan XOR problemini gerçekleyen yapay sinir ağı devresi verilmiştir. Bu aşamadan sonra ise, benzer yöntemlerle FGMOS nöron yapısıyla kurulmuş olan Iris (süsen çiçeği) sınıflama işlemi yapan devre elde edilmiştir. Daha sonra, bu ağı gerçekleyen devre tasarlanıp HSPICE programında simülasyon sonuçları elde edilmiştir. Son olarak da, MATLAB ve HSPICE programlarında elde edilen simülasyon sonuçları karşılaştırmalı bir şekilde verilmiştir.

Gerçekleştirilen bütün devrelerin giriş gerilimleri, ilgili kapasitenin yüzen geçide bağlı toplam kapasiteye oranı kadar zayıflatılmakta ve bu durum giriş gerilimlerinin beslemeden beslemeye tam salınımlı olmasına olanak tanımaktadır. Tasarımı yapılan devrelerde bu özelliğin sağlandığı görülmüştür. Devrelerin içerdiği FGMOS yapılarda seri kapasitif yapılar olduğundan simule etme güçlüğü makro modellerden yararlanarak aşılmaya çalışılmıştır. Devrelerin gerçeklenmesinde TÜBİTAK-YİTAL 1.5µm çift poli – çift metal CMOS parametreleri kullanılmıştır.

Anahtar kelimeler: FGMOS, nöron, YSA, sınıflayıcı, Iris

ABSTRACT

In this study, the multiplier structure, proposed by Vlassis and Siskos in 2001 (Vlassis, 2001), which is a four-quadrant multiplier with rail-to-rail linear input and low nonlinearity obtained by improving the squarer based on the differential voltage attenuator structure composed of FGMOS transistors has been studied and a neuron model using this multiplier circuit and FGMOS based differential comparator. has been designed. The originality of the circuit in question is that, it uses floating-gate multi-input MOS devices, also called “FGMOS transistors”. The floating-gate potential of a FGMOS transistor is being determined by the capacitive coupling of the second poly areas -which biased by multi inputs- with the floating gate formed by the first poly layer. So, the influence of the any input voltage over the floating gate should be proportional to the value of the capacitor between this input and the floating gate.

Thus the construction of the artificial neural networks has become possible using the FGMOS neuron structures that have been designed. The neural networks have been trained and the weights were obtained by determining their structure with MATLAB. An artificial neural network circuit is presented, implementing the well known XOR problem in the artificial neural networks area, using FGMOS neuron structures. After this stage, the circuit that makes Iris classification has been obtained by similar methods. Next, the circuit implementing this network was designed and the simulation results were obtained by the HSPICE program. Finally, the simulation results obtained in MATLAB and HSPICE are presented in a comparative form.

The inputs of all the implemented circuits are attenuated as much as the proportion between the related capacitor value and the total capacitance value connected to the floating gate and this gives the opportunity to use full-swing input voltages. This characteristic is being demonstrated for the designed circuits. The difficulty to simulate the circuits because of the FGMOS structures containing serial capacitance structures, is being defeated by the macromodels calculating the floating gate potentials. The TÜBİTAK-YİTAL 1.5 μ m double-poly double-metal CMOS parameters have been used for the implementation of the circuits.

Keywords: FGMOS, neuron, ANN, classifier, Iris

1. GİRİŞ

Yapay sinir ağı insan beyninin çalışma sisteminin yapay olarak benzetimi çalışmalarının bir sonucu olarak ortaya çıkmıştır. En genel anlamda bir yapay sinir ağı insan beynindeki birçok nöronun (sinir hücresi) ya da yapay olarak basit işlemcilerin birbirlerine değişik etki seviyeleri ile bağlanması ile oluşan karmaşık bir sistem olarak düşünülebilir. Önceleri temel tıp bilimlerinde insan beynindeki nöronların matematiksel modelleme çabaları ile başlayan çalışmalar geçtiğimiz on beş sene içerisinde disipline bir şekil almıştır. Yapılan bu çalışmalar sonucunda gelişen teknolojinin etkisiyle bir yapay sinir ağının elektronik devre olarak gerçekleştirilmesi ihtiyaç haline gelmiştir. Bir yapay sinir ağının temel yapı taşları nöronlar olduğuna göre 'nöronları gerçeklemek zorunlu hale gelmiştir' denilebilir. İşte bu çalışmanın amacı bir biyolojik nöronun elektronik devre olarak gerçekleştirilmesidir. Nöronun modellenmesi daha sonra bu modele ve matematiksel ifadesine uygun transistör seviyesinde tasarımın gerçekleştirilmesi çalışmanın ana konusunu oluşturmaktadır. Öncelikle bu konuyla ilişkisi olan çalışmaları, ne tür gerçeklemelerin –gerek yapay nöron, gerekse yarı-yapay nöron- olduğunu, teknolojinin hangi seviyede bulunduğunu ve geleceğe yansımaların ne şekilde olabileceği hakkında genel bir bilgi vermek gerekmektedir.

1.1 Önceki Çalışmalar ve Teknolojinin Bugün Geldiği Nokta

Bal arılarının bilişsel yeteneklerine hayranlık duyanlardan Rodney Douglas 2001 yılında Nature dergisinde bir makalesinde: "Neler yapabildiklerine bir bakın. Nasıl uçacaklarını öğreniyor, yollarını bulabiliyorlar. Karmaşık izleri tanıma ve bir dereceye kadar da iletişim kurabilme yetisine sahipler. Alıcıları ise hayli puslu. Hiç dünyayı bir arının gözlerinden gördünüz mü? Son derece ürkütücüdür." diyor. Douglas bir hayvanbilimci olsaydı, böylesi coşkulu sözler kimseyi şaşırtmazdı. Oysa onun Zürih'te bulunan Nörobilişim Enstitüsü'ndeki çalışma grubunun asıl ilgi konusu hayvanlar değil; silikon devreler. Douglas, hayvan beyinlerinin işleyiş tarzını taklit eden mikro elektronik devreler tasarlamakla uğraşan "nöromorfoloji mühendisliğinin" önde gelen isimlerindendir. Sayısal bilgisayarların, bol sıfırlı hesaplamalarda kesinlik açısından son derece başarılı oldukları bir gerçek. Bu mühendislere göre hayvanlar da gerçek yaşamın karmaşıklığıyla verimli bir etkileşime girmelerine olanak tanıyan beyin ve duyulara sahipler. O halde, bu ikisinin en iyi yönlerini birleştiren cihazlar neden geliştirilmesin? Massachusetts Teknoloji Enstitüsü'nde çalışan elektronik mühendisi Rahul Sarpeshkar, "Aptal-görünümlü organizmalar inanılması güç hesaplamalar yapıyor" diyor. Buradan da "bunu elektronikte taklit etmemiz gerekiyor" sonucu çıkar.

Beyin Taklitleri

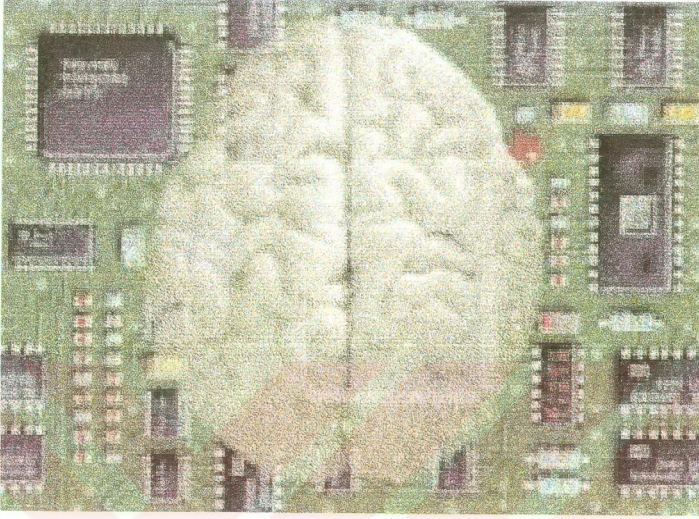
Yapay sinir ağırları yoluyla hayvan beyinlerinin taklit edilmesine yönelik girişimler 1940'lardan beri söz konusudur. Buradaki ana fikir, ağdaki diğer birimlerden gelen sinyalleri birleştirebilen ve diğer bir birimler grubuna sinyaller gönderebilen bağımsız işlemci birimlerini -yapay sinir hücrelerini- birbirine bağlamaktır. Birimlerden her biri ancak sınırlı miktarda bilgi işleyebilecek olsa da bunlar bir ağ bünyesinde toplandıklarında, en azından kuramsal olarak, gerçek bir beynin işlevlerini taklit etmeleri bekleniyor.

Sinir ağırları 1980'lerin başlarında araştırmacıların, ağırları belirli örgüleri tanıma yönünde "eğitme" lerine olanak tanıyan karmaşık öğretici "kurallar" geliştirmeleriyle gündeme gelmişti. O sıralar, programcılar sayısal bilgisayarların kalıplar veya örüntüleri ayırt etmelerini sağlamakta hayli zorlanmaktaydı. Örneğin, imzaları ele alalım. Her imza attığımızda, birbirleri arasında ufak farklar taşıyan örüntüler meydana getiririz. Bir otomatik imza tanıma sisteminin çalışması için, sistemin bu farklı örüntüleri aynı imzanın birer örneği olarak tanıması gerekir. Bu işi sayısal bilgisayarlar yoluyla başarmak güç; oysa yapay sinir ağırlarına bu teknik öğretilir. Bir kişinin imzasının çeşitli örnekleri giriş sinir hücrelerine sunulurken, çıkış sinir hücreleri de sürekli aynı çıktıyı verecek şekilde düzenlenir. Çıktı birimleriyle bir imzanın değişik biçimleri karşısında devreye giren çeşitli arabulucu sinir hücreleri arasındaki bağlantıların seçici bir şekilde güçlendirilmesiyle, ağırlar imzaları tanımayı öğrenebilir.

Silikon kullanarak bu tür bir sinir ağı inşa etmek hiç de kolay değildir. Ağda tek halde bulunan sinir hücreleri arasındaki bağlantılar, aralarındaki birleşme noktalarında saklanan yük miktarının değiştirilmesi ve korunmasıyla güçlendirilir. Nöromorfoloji mühendislerinin ilk temsilcileri için bu büyük bir sorundu. Ağların her kapatılışında yükler kayboluyor ve ağırlar öğrendikleri görevleri "unutuyordu". Geleneksel bilgisayarlarda bu tür bir sorun yaşanmaz; dolayısıyla silikondan ağırlar kurmak yerine araştırmacılar yazılımlardan yararlanarak bunlar taklit etmekte ve çalışılmış durumları bilgisayarın sayısal hafızasında saklayabilmekteler. Geçtiğimiz 30 yılda, taklitçi sinir ağırları (beyni diğerlerine göre daha iyi taklit edebilenler) üzerinde bir hayli çalışılmıştır. Bunlar, gerçek beyinlerin nasıl çalıştığına ve çok sayıda yararlı uygulamayı nasıl gerçekleştirdiğine epeyce ışık tutmuştur.

Ancak taklitçi ağırlar nöromorfoloji mühendisliğinin gerçek ruhundan farklı bir noktadadır. Biyolojide, beyinler ve duyu organları hayli hızlı bir şekilde ve fazla güç harcamadan çalışır. Taklit süreci yavaş olup aşırı güç harcayan bilgisayarlara gereksinim duyar. Ancak Pasadena'daki California Teknoloji Enstitüsü'nden Carver Mead ve ekibinin 1980'lerin

sonlarından itibaren yürüttükleri arařtırmalar sayesinde, dünya apında yayılmıř bir dzine arařtırma grubu, gnmzde gerek nromorfolojik devreler zerinde alıřmaktadır.



řekil 1.1 İnsan beyni elektronik entegrasyonu [2]

Mead, analog elektronie dayalı silikon devreler oluřturmakla ğrařmıřtır. Dijital bilgisayar iplerinde sorunlar kesin algoritmaların kullanımıyla zlr. Sayılar ikili bir kodla ayrı gerilim deėerlerine karřılık gelen '0' ve '1' ile-tems il edilir ve merkezi bir "saat" bilginin ip zerinden nasıl gnderileceėini dzenler.

Karřılařtırıldığında analog devrelerin hayli dzensiz olduėu grlr. Farklı sayılar gstermek zere bir dizi gerilim kullanılır ve sinyaller merkezi bir kontrol olmaksızın devrenin farklı blmleri arasında dolařır. Kesin algoritmalar uygulamak olanaksızdır. Bunun yerine, devre, sinyallerin "doėal" akıřının yararlı iřlemler yapacaėı řekilde tasarlanmıřtır ki bu, hayvan beyinlerinin alıřma biimine bir hayli yakındır.

Mead'in saėladıėı en byk katkılardan biri, ykn uzun sre gvenli bir řekilde saklanabildiėi cihazlar olan yzer-geit (floating gate) analog yapıların icadıdır. Bellek sorununun zlmesine yardımcı olan bu icat, retinanın, yani gzlerimizin arka blmnde bulunan ıřık-algılayıcı katmanın iřlevini taklit eden nromorfolojik aletlere giden yolun nn amıř bulunuyor.

Retina sadece bir foto alıcı (ışığa hassas alıcı sinir) hücreler toplamından ibaret değildir. Hesaplamalar yapar, nesnelerin kenarlarını daha belirginleştirmek üzere bilgileri işler (buna "kenar çıkarsama" denir), parlak veya karanlık koşullar dengelemek üzere sinyalin "kazanımını" ya da büyütülmüş halini ayarlar. Güçlü dijital makineler bu "önişleme"yi taklit edebilir, ancak sinir sistemleri bu işi basit, az güç gerektiren analog devrelerle yapar [1].

Mead bu basitliği taklit etmek için yaptığı çalışmalarda Caltech biyologlarından Misha Mahowald ile birlikte çalışmıştır. İkisinin ürettiği ilk retina, zaman içinde daha yetkin foto alıcılar ve daha karmaşık devre tasarımlarıyla geliştirildi. Ancak temel felsefe aynı kaldı. Silikon retinalar, her bir alıcının komşularına bağlandığı bir dizi foto-alıcıdan oluşur. Dirençler, yükselticiler ve diğer cihazlardan oluşan bir ağ sinyallerin alıcılar arasında gerçek zamanda akmasını sağlar. Bu devreyi gerçek retinalardaki ağlar fiziksel olarak taklit edecek şekilde tasarlamak olanaksızdır; hücrelerin ve bunlar arasındaki bağlantıların miktarı son derece büyüktür. Bunun yerine, nöromorfoloji mühendisleri retinadaki ağların önişlemleri nasıl yaptıkları üzerinde çalışmakta, ardından da aynı işi yapan daha basit analog ağlar tasarlamaktadır. Mühendisler, doğanın hesaplama yapmak için kullandığı numaralarından özgürce esinlenirler. Ancak alışveriş tek taraflı değildir, araştırmacılar da, hayvan beyinlerinin silikon modellerini yaparak aynı zamanda biyoloji hakkında bilgilerini geliştirmektedirler. Günümüzde, modern versiyonlar, örneğin Douglas ve onun Zürich'teki meslektaşları tarafından üretilen cihazlar, biyolojik retinalar kadar kenar çıkarsaması ve kazanım ayarlaması yapabilmektedir. Bu proje, Zürih'te bulunan Nörobilişim Enstitüsü araştırmacılarından Kevan Martin ile elektronik mühendisi Shih-Chii-Liu'nun insan beyninin görsel imgeleri nasıl işlediğini araştırmak üzere ortak bir projede çalışmaya başlamasıyla hayata geçmiştir. Proje, Liu tarafından yapılan, kabuksal çip adlı (insanın görme sisteminin bir bölümü üzerine yerleştirilen analog silikon devresi) bir cihaz üzerine kurulmuştur.

Mahowald 1992'de, nöromorfoloji mühendisliğinin alet kutusuna (toolbox) adres-olay-temsili (AER:address-event-representation) görünümünde yeni bir bileşen ekledi. Bu yöntem çiplerin birbirleriyle iletişim kurmasına olanak tanıyordu. Her ne kadar nöromorfolojik çipler genellikle birbirlerine bağlı yüzlerce yapay sinir nöronu içerse de, bir çipin dış dünyayla kurabileceği bağlantı sayısının bir sınırı vardır. İki nöromorfolojik çip birbirine bağlandığında, tek tek bütün nöronlar birbirine doğrudan bağlamak olanaksızdır. AER'de belirli birimlerle ortaklık kurmuş giren ve çıkan sinyaller, bir merkezi "veri yolu" ile gönderilir. Çiplerin içindeki bağlantıların yönetilmesi daha kolaylaşır, böylelikle veri yolu, "adres" lerini (her bir birime özgü sayı) kullanarak tek tek birimlerle doğrudan konuşabilir.

Her şeyi gören fare: Nöromorfoloji mühendisleri ticari başarı şansına sahip bir ürüne bile dikkat çekiyor: 1994'te, California'da bulunan Fremont Logitech tarafından ortaya çıkarılan optik bilgisayar faresi, gövdesindeki top yoluyla hareketi izleyen alışılmış farelerden farklı olarak altındaki masaya "bakarak" hareketi takip eden görsel bir alıcıyla konumundaki değişiklikleri gözlüyor. Bu tür bir soruna ilişkin sayısal yaklaşımda, alıcının ilerleyişi boyunca sahnenin çekilen ardışık enstantane görüntülerinin karşılaştırılması söz konusu; bu da cihazın ucuzluğu göz önüne alınırsa oldukça talepkar bir işlem. Bunun yerine Logitech, önce André van Schaik, ardından da Zürich'te bulunan İsviçre Elektronik ve Mikroteknoloji Merkezi tarafından geliştirilen bir nöromorfolojik çipi piyasaya sürdü. Van Schaik ucuz, az güç gerektiren bir cihaz tasarlamak için sineklerin görme sistemine ilişkin bilgilerini kullanmıştı.

Şu anda Avustralya'da Sidney Üniversitesi'nde bulunan Van Schaik, "Sineklerin zihinleri bir foto alıcıda meydana gelen yoğunluk değişikliğini komşu alıcılardaki gecikmiş versiyonlarla karşılaştırır" diyor. Sineğin gözündeki hücreler, komşu foto alıcılardan gelen çıktılarını inceleyerek, gördüğü hareketli nesnelerin hızını ve yönünü tahmin eder. Fakat tek haldeki "hareket saptayıcı" hücreler bu miktarlarla ilgili ancak kaba tahminlerde bulunur. Harekete ilişkin daha kesin tahminler, işlemlerin daha ileriki aşamalarında hücrelerin farklı hareket-saptayıcı hücrelerden gelen çıktıları karşılaştırmasıyla gerçekleştirilir. Optik farenin çalışma prensibi de bunun gibidir.

Logitech'in faresi, bilgisayarların bol sıfırlı rakamlarla şaşmaz kesinlikte işlem yapabilme becerisini, hayvan zihinlerinin hızlı, az güç gerektiren örüntü incelemesi becerisiyle kaynaştırarak, analog nöral cihazların geleneksel sayısal bilgisayarlarla nasıl birleştirilebileceğini ortaya koyar. Fakat eğer nöral cihazlar bu denli büyük bir potansiyele sahipse, neden bu alanda binlerce değil de sadece yüzlerce araştırmacı çalışıyor ? Bunun nedenlerinden biri, analog bilgisayarlığın, sayısal çip teknolojisinde yaşanan dur durak bilmez ilerlemelerin kurban oluşudur. Çip üreticileri, örneğin bir analog konuşma işleme aletine yatırım yapmakta gönülsüz davranırken, aralıksız bir şekilde gelişen bilgisayar teknolojisi, soruna sayısal bir yaklaşım getirmek üzere yeni yollar sunmaktadır.

Sinir Hücreleri

Bu dinamik karşısında, bu işle uğraşan mühendisler, biyolojiden esinlenmiş bilgisayar teknolojisinin taşıdığı avantajların göz ardı edilemeyeceği hücre uygulamaları üzerinde yoğunlaşmaktadır. Örneğin, kendi başına çalışması gereken devingen cihazlar üzerinde çalışan araştırmacılar için, nöral aletlerin güç randımanı çok caziptir.

"Biyonik" yerleřtirim cihazlar üretmek üzere nöromorfoloji teknolojisini kullanmanın altında yatan mantık budur. Günümüzün, doğuřtan sağır kiřilerin duyma yetilerini bir ölçüde onarmak üzere kulak salyangozuna yerleřtirilen cihazları , ses dalgalarını algılayan saç hücrelerinin mekanik versiyonlarını içerir. 30.000'den fazla kıl hücresinin iřitme sinirlerine girdilerini 10 ila 20 elektrot kullanarak taklit eden yapay salyangozlar görece kabadır; yine de sonuçlar hayli etkileyici olabilir. En iyi durumda, yerleřtirilen cihazlar kullanıcısının telefon görüşmesi yapmasına olanak tanımıřtır.

Ancak günümüzün cihazları hacimli, aşırı güç harcayan ve harici kullanımlı sayısal sinyal işlemcisiinden yararlanır. Ayrıca, yerleřtirilen cihazın birkaç haftada bir yeniden doldurulması gerekir ki bu da kullanıcının uzun saatlerini bir dolum istasyonunda geçirmesi anlamına gelir. Londra'daki Imperial College'in bir yan kuruluşu olan Toumaz Technologies, son yıllarda sayısal sinyal işlemcisinin daha küçük, daha az güç harcayan bir analog versiyonunu üretmekle uğrařmaktadır. Bu projenin beyinlerinden elektronik mühendisi Chris Toumazou'a göre, hem yeni, az güç harcayan elektrotlar hem de bunun işlemcisini kapsayan cihaz, kulağına içine yerleřtirilecek ve yılda sadece bir kez yeniden doldurulması gerekecektir. Klinik denemeler 2001 yılının sonunda bařlamıřtır.

Toumazou ayrıca Mead ile Mahowald'ın öncülüğünü yaptığı analog retinaların işlevsel tıbbi cihazlara dönüřtürülmesi üzerinde çalışmaya bařlamıřtır. Bu, salyangoza yerleřtirilen cihazların geliştirilmesinden daha zorlu bir iş, çünkü retinadaki öniřlemler çok daha karmařık ve işin içinde olan sinir sayısı çok daha fazladır Her bir görme sinirimizde 1 milyon lif bulunur, iřitme sinirindeyse bu sayı 30,000'dir.

Öte yandan, Baltimore'da bulunan Johns Hopkins Üniversitesi elektronik mühendislerinden Ralph Etienne-Cummings, insan omuriliğinin yürüme sırasında bacaklarda yařanan kas gerilmesini ayarlama biçimini taklit eden bir devre üzerinde çalışmaktadır. Bunun bilincinde olmasak da, yürüme karmařık ve kesintisiz gerçek zamanlı hesaplamalar gerektirir. Omuriliğimiz, doğru kasılma dizgesini hesaplamak üzere dengemize ve bacak konumlarımıza ilişkin bilgileri bir araya getirir. Etienne-Cummings ve Mahomet, Illinois'de bulunan Iguana Robotics ile birlikte, belden ařağısı felçli kiřilerin omuriliklerine yerleřtirilerek onların yeniden yürümelerine yardımcı olacak bir çip gerçekleřtirmek üzere çalışıyor.

En cořkulu nöromorfoloji mühendisleri bile mevcut ürün sayısının pek de etkileyici olmadığını farkında. Ancak bir dizi bařarılı biyolojik yerleřtirim cihaz durumu bambařka bir noktaya taşıyacak. Toumazou'nun çalışanlarından, řu sıralar Johns Hopkins'te bulunan

Andreas Andreou, "Yolumuz engbelerle dolu, fakat řu an için durum hayli heyecan verici" diyor (Giles, 2001).

Bu bölümde ayrıca řu ilginç 'yarı-yapay nöron' olarak adlandırdığımız yapıdan bahsetmeden geçemeyeceğiz.

Yarı-yapay nöron

Mikro elektronik ve nöronları (sinir hücreleri) bir araya getirme, bilim kurgunun uzun yıllar konusu olmuş ve nörobilgisayarlar, beyine eklenen yongalar filmlerdeki yerini almış durumdadır. Bugün nöronlarla mikro elektroniğı birleřtirmenin (nöroelektronik) iki yaklaşımı var. Birincisi, nöron ağıının silikon üzerinde bir mikro elektronik halkaya yerleřtirilmesi ve geleneksel halkanın sinyalinii işlemlmeyi içeriior. İkincisi, elektronik ile nöron halkasının, biyolojik işlevleri arařtırmak ve desteklemek için mikro pipetler yoluyla birleřtirilmesi şeklindedir. Elektronik araçlar ve nöron ağlarının mikroskobik seviyede birbiriyle etkileşimini gerektiğinden, bu iki yapıyı bir araya getirmek çok gelişmiş teknik gerektirir. Bir nöron ağı, altta yatan mekanizma bilinmediğı halde elektronik veri işlemlmesini destekleyebilir. Bir gün nöroelektronik yongaların hasarlanan sinirdokusu yerine nöroprotez olarak yerleřtirilmeleri umulabilir. Nöron ağlarının tam bir elektronik kontrolünün yapılabilmesi, biyolojik nörodinamiklerin çözülmemiş doğasını ortaya koymaya da yardım edebilir. Bu türden hibrid halkaların uzun süreli çalışmaları, biyolojik bir yapı ile elektro kimyasal tepkimeye girmeyen elektronik arabirime gerek duyar [2].

Almanya'da Max Planck Biyokimya Enstitüsü 'nün Hücre zarı ve nörofizik bölümünden Peter Fromherz ve Gunther Zeck adlı iki arařtırmacı, salyangoz nöronlarını bir silikon yonga üzerinde mikroskobik sabitleřtiricilerle bir araya getirerek nöroelektronik devrelerin ilk adımını attılar. Bu çalışmalarını Proceedings of the National Academy of Sciences dergisinin Ağustos 2001 sayısında yayımladılar. Yaptıkları ağıdaki sinir hücreleri, bir diğeri sinir hücresi ve transistorla bağlantılar oluşturmak üzere büyüdüler. Bir yonga uyarıldığında bir nörondan diğeriine ve silikon bir anahtarla da yongaya geriye doğru akım geçebileceğini gösterdiler. Yani, silikon ve nöronlardan oluşan devre canlanıyordu. 1999 'da, Peter Fromherz, Stefano Vassanelli ile The Journal of Neuroscience 'da bu çalışmalarının öncülerini yayımlamışlardı. Fare hipokampusundan (denizati bölgesi) elde ettikleri nöronları yongalar üzerinden kültüre ederek, hücrelerin birbirine bağlantı yerlerinde, hücre içinden dışına olan potasyum akımını transistor üzerinden ölçmüşlerdi. Canlanan devreler daha sonraki çalışmalarla da hayat bulmaya başladı (Zeck, 2001).

1.2 Çalışmanın Amacı ve Kapsamı

Tüm yukarıda anlatılanlara göre teknolojinin eriştiği seviye ortadır. Belirtilen nedenlerden dolayı bir analog yapay nöronun elektronik devre olarak tasarlanması ne kadar önemli bir yer tuttuğu gayet açıktır. Bu noktada, en başta değindiğimiz gibi bu çalışmanın amacını analog bir nöron tasarlayabilmek oluşturmıştır. Yapay nöronun transistörler seviyesinde gerçekleştirilmesi ve bu nöronlarla gerçekleştirilen yapay sinir ağlarının uygulamaları ise çalışmanın içeriğini oluşturur.

Bu gerçeklemler yapılırken transistör bazında devre gerçekleştirme yöntemleri kullanılmış, devre tüm devre olarak tasarlandığından sadece transistör içerecek şekilde kurulmuştur. Aynı zamanda hem FGMOS hem de MOS transistörlerinin kombinasyonları kullanılmıştır.

Önceki nöron tasarımlarında genelde rastlanan yapı (Bayraktaroğlu, 1998) akım çıkışlı bir çarpıcı, sonra bu çarpımların toplama işlemi için bir işlemsel kuvvetlendirici (bu yapı aynı zamanda I-V çevirimi yapar), daha sonra aktivasyon fonksiyonunu gerçekleştiren örneğin sigmoid fonksiyon üreticiler kullanılıyor. Fakat bu çalışmada bütün bunların yerine toplama işlemini ve aktivasyon fonksiyonundan geçirme işlemini (çarpıcının çıkışı gerilim olduğundan I-V çevirimine gerek kalmıyor), ikisini bir arada yapabilen bir yapı olan FGMOS transistörlerle gerçekleştirilmiş bir farksal karşılaştırıcı kullanılarak bu sorun da daha az eleman ve yapı kullanılarak halledilmiştir.

Sonuç olarak tezde, Vlassis ve Siskos'un 2001 yılında önerdikleri (Vlassis, 2001) FGMOS transistörlerden oluşan dört bölgeli çarpıcı yapısı ve yine FGMOS transistörlere dayanan karşılaştırıcı yapısı kullanılarak nöron tasarımı gerçekleştirilmiştir. Oluşturulan yapay sinir ağları MATLAB programında eğitilip yapısı belirlenerek ağırlıklar elde edilmiştir. Daha sonra bu tasarımı gerçekleştirilmiş olan FGMOS nöron yapıları kullanılarak yapay sinir ağlarında çok bilinen problem olan XOR problemini gerçekleyen devre verilmiştir. Bu aşamadan sonra ise, FGMOS nöron yapısıyla kurulmuş olan Iris (süsen çiçeği) sınıflama işlemi yapan devre elde edilmiştir. Sonra, bu ağı gerçekleyen devre tasarlanıp HSPICE programında simülasyon sonuçları elde edilmiştir. Son olarak da, MATLAB'de ve HSPICE'da elde edilen simülasyon sonuçları karşılaştırmalı bir şekilde verilmiştir.

2. YAPAY SİNİR AĞI HAKKINDA GENEL BİLGİLER

Yapay sinir ağları, teorik hale getirilmiş zeka ve beyin faaliyetlerinin matematiksel modelleridir. Ancak biyolojik sistemler o kadar karmaşıktır ki yapay sinir ağı için kullandığımız modeller biyolojik modellerin fazlaca basite indirgenmiş biçimleri şeklindedir.

Beyin çok kompleks, doğrusal olmayan ve paralel bir (bilgi işleme sistemi) bilgisayardır. Beynin nöronları organize ederek şekil tanıma, algılama ve motor kontrolü gibi belirli hesaplamaları bugün mevcut olan en hızlı bilgisayarlardan daha hızlı yapabilme kapasitesine sahiptir. Örneğin bilgi işleme yolu olan görmeyi düşünelim. (Churcland, 1992, Marr,1982) Görme sisteminin fonksiyonu çevremizi tanımlamayı sağlama, daha da önemlisi çevreyle ilişkiyi sağlayacak bilgiyi sağlamaktır. Daha belirli olarak, beyin rutin bir şekilde 100-200 ms mertebesinde somut hatırlama, algılama işlevi görür (örneğin tanıdık olmayan bir resim içinde bilinen (tanınan) bir yüzü algılayıp tanıyabilir), daha basit karmaşıklıkta problemi en ileri bilgisayarlarla çözmek için günler gerekir (Churcland, 1986).

Yapay sinir ağları ya da kısaca YSA insan beyninin çalışma sisteminin yapay olarak benzetimi çalışmalarının bir sonucu olarak ortaya çıkmıştır. En genel anlamda bir YSA insan beynindeki birçok nöronun ya da yapay olarak basit işlemcilerin birbirlerine değişik etki seviyeleri ile bağlanması ile oluşan karmaşık bir sistem olarak düşünülebilir. Önceleri temel tıp bilimlerinde insan beynindeki nöronların matematiksel modelleme çabaları ile başlayan çalışmalar geçtiğimiz on beş sene içerisinde disipline bir şekil almıştır. YSA bugün fizik, matematik, elektronik ve bilgisayar mühendisliği gibi çok farklı bilim dallarında araştırma konusu haline gelmiştir. Yapay sinir ağları ile devre gerçekleştirme ve bu devreyi eğiterek istenen fonksiyonun sentezini elde etme bilimin hedef aldığı ilerlemelerdendir. YSA'nın pratik kullanımı genelde çok farklı yapıda ve formlarda bulunabilen bilgi verilerini hızlı bir şekilde tanımlama ve algılama üzerinedir. Aslında mühendislik uygulamalarında YSA'nın geniş çaplı kullanımının en önemli nedeni klasik tekniklerle çözümü zor problemler için etkin bir alternatif oluşturmasıdır (Haykin, 1999).

2.1 Biyolojik Sinir Ağları

Yapay sinir ağlarını açıklamadan önce doğal sinir ağı olan biyolojik sinir ağları hakkında bazı temel bilgileri vermekte fayda vardır. Tipik olarak, nöronlar lojik kapılara göre çok daha yavaştır. Silikon bir kırılgın (çip) içinde olaylar nano saniyeler (10^{-9}) mertebesinde, nöronlar içinde olaylar mili saniye (10^{-3}) mertebesinde oluşur. bununla birlikte beyin oldukça yavaş bir hızda çalışmasına rağmen çok fazla sayıda nöronlara (sinir hücresi) ve birbirleri

arasında çok geniş bir bağlantı ağına sahiptir. Gerçektende en karmaşık sinir ağı 'Cerebral Cortex' denen beyindir. İnsan korteksinde yaklaşık 10 milyar sinir hücresi olduğu ve 60 trilyon bağlantı veya sinaps olduğu tahmin edilmektedir. Net sonuç şudur ki insan beyninin çok etkin bir yapısı vardır. Yine hücre başına bağlantı sayısı ise 10^4 mertebesinde. Beyin için çalışma frekansı 100 Hz'dir. İnsan beyni ortalama 1.3 kg ağırlığında ve 0.15 m^2 'lik alana sahip en karmaşık sinir ağıdır.

Beynin temel yapısı sinir hücrelerinden oluşur. İç sinir hücreleri kendi öz giriş ve çıkış bağlantılarına sahiptir. Çıkış hücreleri ise beynin bir bölgesini diğerine, beyni kaslara ve duyu organlarını beyine bağlar.

2.1.1 Biyolojik Sinir Hücresinin Anatomisi

Bir biyolojik sinir ağını oluşturan temel öğeler şunlardır (Şekil 2.1'de gösterilmiştir).

Soma: Sinir hücresinin gövdesi.

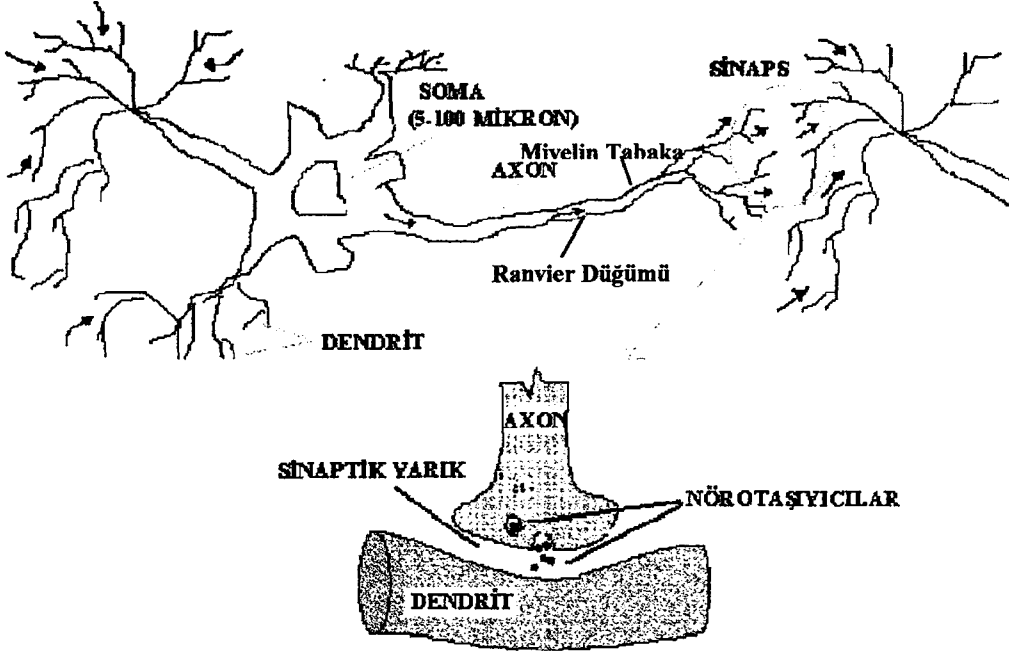
Akson: Çıkış darbelerinin üretildiği elektriksel olarak aktif gövde uzantısı.

Dendrit: Diğer hücrelerden gelen işaretleri toplayan elektriksel olarak pasif hücre kolları.

Sinaps: İki farklı hücrenin dendritlerinin bağlantı noktaları.

Miyelin tabaka: Sitoplazma ile hücreler arası sıvı arasındaki kapasiteyi düşürerek işaretlerin yayılma hızını arttırmaya yarayan yalıtım maddesi.

Ranvier düğümü: Miyelin tabaka ile kaplı aksın üzerinde her birkaç milimetrede yer alan ve işaretleri periyodik olarak yeniden üretmeye yarayan boğum.



Şekil 2.1 Biyolojik sinir hücresinin (nöron) basitleştirilmiş anatomik yapısı

2.1.2 Biyolojik Sinir Hücresinin Fizyolojisi

Hücre, sitoplazma dengede iken hücreler arası sıvıya göre -85 mV 'luk bir gerilim sınırına kadar polarizedir. Bu potansiyel farkı 50 Å^0 kalınlığındaki fosfolipid moleküllerini iki katmanlı olarak düzenlemesinden oluşmuş hücre zarı üzerindedir.

Akson tarafından bir elektriksel darbenin üretilmesi için -40 mV 'luk eşik gerilimini aşacak ve hücre zarını depolarize edecek yönde sitoplazmaya akım enjekte edilmelidir. Bu darbenin biçimi darbeyi üreten sistemden bağımsızdır.

Nöron, soma adı verilen hücre gövdesi, dentrit denilen kıvrımlı uzantılar ve somanın dalları sayesinde nöronu dallarına bağlayan tek sinir fiberli aksondan oluşur. Dentritler hücreye gelen girişleri toplarlar. Dentrit tarafından alınan işaretler hücrede birleştirilerek bir çıkış darbesi üretilip üretilmeyeceğine karar verilir. Eğer bir iş yapılacaksa üretilen çıkış darbesi aksonlar tarafından taşınarak diğer nöronlarla olan bağlantılara veya uç organlara iletilir. Beyindeki korteksde her nöronun karşılığı vardır. Bir nöronun çıkışı ona bağlı olan bütün nöronlara iletilir. Fakat korteks işin yapılabilmesi için hangi nöron harekete geçirilecekse sadece ona komut gönderir.

Somanın içinde ve çevresinde Na^{+1} , K^{+1} , Ca^{+2} ve Cl^{-1} iyonları vardır. Potasyum yoğunluğu nöronun içinde, Sodyum yoğunluğu dışındadır. Somanın zarı elektriksel olarak uyarılınca (Söz konusu uyarı genellikle bir gerilim düşmesidir) zar, Na^{+1} ve Ca^{+2} gibi diğer iyonların

içeri girmesine izin verir ve somanın iç durumunu değiştirir. Nöronlar arası bağlantılar hücre gövdesinde veya sinaps adı verilen dentritlerdeki geçişlerde olur. Yardımcı bir benzetme aksonlarla dentritlerin elektrik sinyallerini nörona ileten değişik empedansdaki yalıtılmış iletken olmasıdır. Sinir sistemi milyarlarca nöron ile tek bir nörondan çıkan aksonu 10000 kadar diğer nöronları bağlayan bir ağdır. Sinapslarla düzeltilen işaretleri taşıyan aksonlar ve dentritlerle iç içe geçmiş nöronlar bir sinir ağı oluştururlar.

Eşik birimi çıkışları toplayan ve sadece girişin toplamı iç eşik değerini aştığında bir çıkış üreten işlem elemanıdır. Bir eşik birimi olarak nöron sinapslarındaki işaretleri alır ve hepsini toplar, eğer toplanan işaret gücü eşiği geçecek kadar güçlü ise diğer nöronları ve dendritleri uyaran akson boyunca bir işaret gönderilir. Kesişen dendritlerden gelen sinapslarla kapılanan bütün işaretleri soma toplar. Toplam işaret daha sonra nöronun iç eşik değeri ile karşılaştırılır ve eşik değeri aşmışsa aksona bir işaret yayar. YSA bu basit nöronların (düğümünün ya da işlem elemanlarının) bağlanarak bir ağa dönüştürülmesi ile meydana getirilir.

2.2 Yapay Sinir Ağının Yapısı

Burada biyolojik sinir ağları hakkındaki kısa bilgilerden sonra, bir yapay sinir ağının ne olduğu ve çalışma prensipleri hakkında, konuyu anlamak bakımından bazı bilgileri fazla teorisine girmeden verilmeye çalışılacaktır.

Bir YSA'nın temel ögesi hücreler yani işlem birimleridir (şekil 2.2). Bir YSA işlem birimleri (elemanları) ile bağlantı geometrisinden oluşan eylem birimi ile bunlara ait bir öğrenme kuralı açıklanarak tanımlanır.

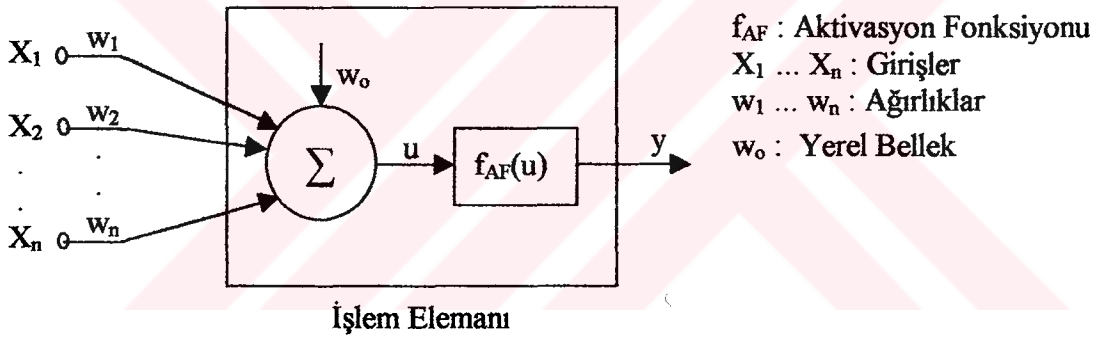
YSA'nın anatomisi yönlü bir graf biçimindedir. Her düğüm hücre denilen n. dereceden doğrusal olmayan dinamik bir devredir.

YSA'ların elektrik devreleri bakımından tanımını özet olarak aşağıdaki gibi sıralanabilir:

- YSA'nın düğümleri birer işlem elemanlarıdır.
- Düğümler arasındaki bağlantılar tek yönlü işaret iletim yollarıdır.
- Her işlem elemanına istenildiği kadar girişe bağlanabilir.
- Her işlem elemanında tek bir çıkış elemanı olabilir (Fakat bu bağlantı kopyalanabilir).
- İşlem elemanları yerel bellek taşıyabilir.
- Her işlem elemanının bir transfer (Aktivasyon) fonksiyonu vardır. Kabul edilebilir ki giriş işaretleri yerel bellekte bulunan işaretlerden biri olabilir. Transfer fonksiyonu sürekli ya

da kısmen sürekli olabilir. Kısmen sürekli çalışma konumunda aktif halde eleman bir çıkış işareti üretir.

- Giriş işaretleri YSA'na bilgi taşır sonuç ise çıkış işaretlerinden alınır.
- Genel işlem elemanı ancak işlem elemanı 'aktif' konumunda ise giriş işaretleri ve yerel bellekteki işarete göre çıkış işareti üretir.
- YSA bir takım aynı transfer fonksiyonuna sahip alt gruplara ayrılabilir. Bunlara katman denir.
- Pek çok YSA giriş katmanı içerir, ancak bu katmanın bir transfer fonksiyonu yoktur.
- YSA'nın transfer fonksiyonu, yerel bellek elemanları ve bağlantı aralıkları, öğrenme kuralı ile giriş çıkış işareti arasındaki bağıntıya göre ayarlanır.
- Aktif yapma girişi için bir zamanlama girişi gerekir.
- Bağlantılarda, taşınan işaretin cinsi tanımlanmalıdır. Bağlantıların geometrisi YSA için önemlidir. Bağlantı işareti her cinsten olabilir.



Şekil 2.2 İşlem elemanı yapısı

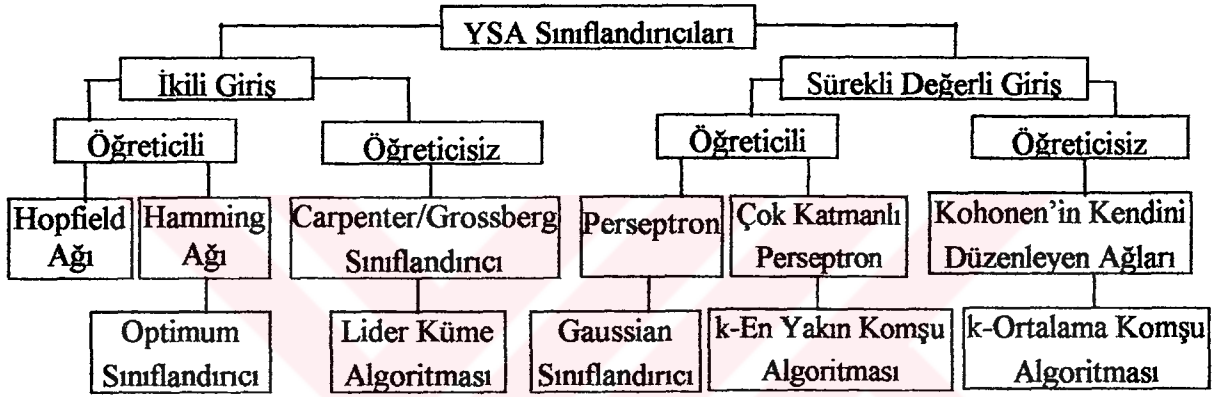
Bağlantı ağırlık katsayıları ağ geometrisi ve hücrelerin iç yapısı ve parametreleri belirli bir öğrenme kuralı ile her bir giriş olasılığına göre değiştirilmesi 'on-line process' ya da bütün giriş olasılıkları değerlendirildikten sonra değiştirilmesi 'off-line process' olarak adlandırılır.

Bir YSA'dan beklenen görev gerçek dünyadaki nesnelerle biyolojik sinir ağının yaptığı işlemi benzer bir yolla yerine getirmesidir. YSA'nın giriş veri tipleri ikili (binary) 0-1 veya sürekli değerlerdir. Bu giriş tiplerinden başka işlem elemanların ait girişleri matematiksel olarak da sınıflandırmak gerekmektedir. Çünkü bir işlem elemanın gelen girişlerin bir kısmı azaltıcı uyarma girişleri olmaktadır. Bu artırıcı ve azaltıcı giriş sınıflarını oluşturur.

2.2.1 Giriş İşareti Sınıfları

İşlem elemanının (nöron) transfer fonksiyonu gelebilecek bütün giriş işaretleri için tanımlanır. Bazen değişik katmanlardan gelen işaret tiplerinin ve katman davranışlarının farklı olması doğaldır. İşaretlerin hangi bölgelerden geldiğinin bilinmesi gerekir. Değişik bölgelere göre işaretlerin sınıflarını tanımlayabiliriz.

Bir işlem elemanına gelen girişler matematiksel tiplerine göre sınıflandırılırlar. YSA giriş veri tiplerine göre ikili giriş (0,1) ve sürekli değerli giriş olmak üzere Şekil 2.3'deki gibi sınıflandırılır.



Şekil 2.3 YSA sınıflandırıcıları

2.2.2 Bağlantı Geometrileri

Bağlantılarda taşınan işaret verisinin cinsi tanımlanmalıdır. Bağlantı geometrisi YSA için çok önemlidir. Bağlantının nereden başlayıp nerede bittiğinin bilinmesi işlem elemanı kümesinin bağlantıları aşağıda tanımlandığı gibi $n \times n$ boyutlu matris biçiminde gösterilebilir.

$$[w_{ij}] = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \dots & w_{nn} \end{bmatrix} \quad \begin{aligned} w_{ij} = w_{ji} = 1 &\Rightarrow i. \text{ işlem elemanı } j. \text{ işlem elemanına bağlı} \\ w_{ij} \neq w_{ji} = 0 &\Rightarrow \text{bağlı değil} \end{aligned} \quad (2.1)$$

En fazla N^2 bağlantı olur. Bağlantılar çeşitli geometrik bölgeler arasında demetler halinde düşünülebilir. Bu bağlantı demetlerinin uyması gereken kurallar şunlardır.

- Bağlantı demetini oluşturan işlem elemanları aynı bölgeden çıkmalıdır.
- Bağlantı demetinin işaretleri aynı matematiksel tipten olmalıdır.
- Bağlantı demetinin işaretleri aynı sınıftan olmalıdır.

➤ Bağlantı demetinin bir seçim fonksiyonu (σ) olmalıdır.

$\sigma : T \rightarrow 2^S$ T: Hedef Bölgesi S: Kaynak bölgesi

Hedef bölgesindeki her işlem elemanı kaynak bölgesindeki her elemana giderse 'full' tam bağlıdır.(Örnek: Çok katmanlı algılayıcı) Eğer her hedef bölgesi elemanı N kaynak bölgesi elemanına bağlı ise 'düzgün dağılma' olasıdır. Ayrıca her bir elemana yine bir kaynak elemanı bağlı ise buna 'bire-bir' bağlı denir.

2.2.3 Ağ Tipleri

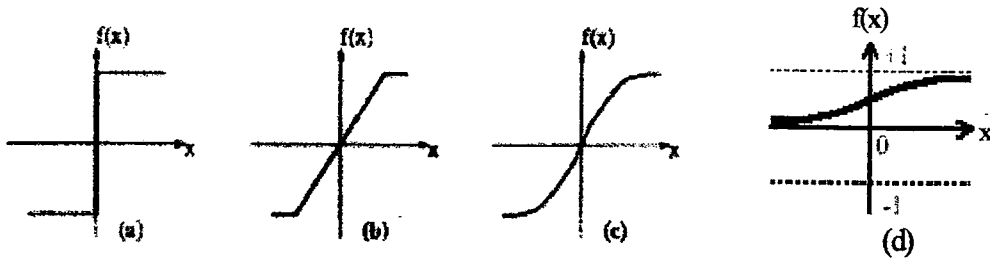
Üç çeşit ağ tipi vardır.

- 1) İleri beslemeli ağ: Her bir katmandaki hücreler sadece bir önceki katmanın hücrelerince beslenir.
- 2) Kaskat bağlantılı ağ: Hücreler sadece önceki katmanlardaki hücrelerce beslenir.
- 3) Geri beslemeli ağ: En az bir hücre sonraki katmanlardaki hücrelerce de beslenir.

Çalışmada kullanılan YSA yapısı ileri beslemeli ve kaskat bağlantılı bir ağ yapısına sahiptir.

2.2.4 Aktivasyon Fonksiyonları

Transfer, eşik fonksiyonu olarak da adlandırılan aktivasyon fonksiyonları muhtemel sonsuz düzlem girişli işlem elemanlarını önceden belirlenmiş sınırdaki çıkış olarak düzenler. Çeşitli aktivasyon fonksiyonları vardır. Bunlardan yaygın olarak kullanılan dördü; basamak, rampa, tanjant-hiperbolik ve sigmoid tip aktivasyon fonksiyonlarıdır. Şekil 2.4'te bu fonksiyonlar gösterilmiştir. Çalışmalarımızda rampa ve tanjant-hiperbolik (tanjant-sigmoid) tip aktivasyon fonksiyonları üzerinde durulmuştur. Denemeler neticesinde ağırlıkların belli değerler arasında kalmasını sağlayan rampa ve tanjant-sigmoid fonksiyonlarının kullanılması uygun bulunmuştur.



Şekil 2.4 Kullanılan transfer fonksiyonları

$+\ell$ ($-\ell$) işlem elemanının maksimumu (minimumu) çoğu zaman doyma seviyesi olarak adlandırılan çıkış değeridir, yani basamak tipi aktivasyon fonksiyonunda giriş değeri pozitif ise işlem elemanı pozitif doymaya gider, giriş negatif ise çıkış negatif doymadadır. Şekil2.4a'da basamak tipi aktivasyon fonksiyonu gösterilmiştir ve denklemi;

$$f(x) = \begin{cases} +\ell, & x \geq 0 \\ -\ell, & x < 0 \end{cases} \quad (2.2)$$

şeklini alır. Şekil 2.4b'de gösterilen kısmi lineer fonksiyonun denklemi $f(x) = ax$ gibidir. a , işlem elemanının x aktivitesini ayarlayan gerçek değerli bir sabittir. Lineer fonksiyon $[-\ell, +\ell]$ sınırları arasında kısıtlandığında şekildeki gibi rampa aktivasyon fonksiyonu olur ve denklemi;

$$f(x) = \begin{cases} +\ell, & x \geq \ell \\ x, & |x| < \ell \\ -\ell, & x \leq -\ell \end{cases} \quad (2.3)$$

şeklindedir. Şekil 2.4c-d'de gösterilen tanjant-hiperbolik ve sigmoid tip aktivasyon fonksiyonu seviyeli, lineer olmayan çıkış veren, sınırlı, monoton artan fonksiyondur. Denklemleri sırasıyla aşağıdaki gibidir:

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad f(x) = \frac{1}{1 + e^{-x}} \quad (2.4)$$

Her işlem elemanı kendisine verilen yerel veriye göre kendisini ayarlayacak bütün YSA'nın enformasyon bölgesinin öğrenmesini sağlar (enformasyon bölgesi olasılık-yoğunluk fonksiyonu ile de tanımlanabilir). Enformasyon bölgesi bir çok uygulamada tanjant-sigmoid için gerçek değerler $'1'$ ile $'-1'$ arasında normalize edilmesi gerekir. Normalize etmek: örneğin normalizasyon katsayısı yüz ise gerçek değeri 85 olan bir girişi 0.85 şeklinde ağa uygulamaktır. Normalizasyon aynı anda bütün girişlere uygulanabilir.

2.2.5 Yapay Sinir Ağlarının Kullanım Nedenleri

Öncelikle resim tanıma (Pattern Recognition) tekniğinin gerekliliği, gerçek dünya ile bilgisayar ilişkisinin başlamasıyla ortaya çıkmıştır. Bu önem YSA'nın çok güçlü örnek tanıma tekniği olarak ortaya çıkmasına ve gelişmesine neden olmuştur. Uygulamada YSA'larının

kıymetini anlayabilmek için öncelikle bu konudaki algoritmaları ve geliştirme tekniklerini iyi kavramak ve özümsemek gerekir:

YSA'larının bazı uygulamalardaki kullanımı için yapılan geliştirmeler bunların bu uygulamalar için ne kadar güçlü ya da yetersiz olduklarının bilinmesine bağlıdır.

YSA'ları gerçek dünyaya ait ilişkileri tanıyabilir. Sınıflandırma, tahmin ve fonksiyon uydurma gibi görevleri yerine getirebilirler. Bunların genel ilişkilerle, ayrık örnekler arası boşluğu birbirlerine bir köprü gibi bağladığını da söylemek yanlış olmaz. YSA'larının ani değişen değerlerden örnek olarak, doğrudan doğruya örneklerin birbirleri arasındaki benzer ilişkileri öğrenme yetenekleri vardır. YSA'ları bir konuda veya olaydaki verilerden hareketle önceki bilgileri bilmese bile sonuç çıkarma yetenekleri de vardır. Çünkü YSA'ları klasik programlama ile çalışmazlar. Bir YSA'nın geliştirilmesi bildiğimiz yazılım geliştirmeye benzemez. Zaten aralarındaki en büyük fark ağların bir işi yapmaları için eğitime gereksinim duymalarıdır. Bildiğimiz gibi klasik programlama böyle değildir.

Bu eğitime işlemleri değişkenlerin tanımlanması, çevrimlerin oluşturulması, şartların tespiti ve bunların testi ile bir derleyici üzerinde çalıştırma veya yazılan kodun hata denetiminden (debug) geçirilmesi gibi işlemleri gerektirmezler. İşte bütün bunların yerine eğitime yöntemleri seçim, analiz ve verilerin bir araya getirilmesi ile başlar. Bunlardan başka istatistik ve işaret işlemeden de bazı teknikleri kullanır. Bir YSA'nın geliştirilmesindeki ilk ve en kritik adım budur.

Bir ağ gerekli ilişkileri oluştururken ağı geliştiren kişi bunun farkına varmaz. Ağ bu ilişkileri bulup yapar ve eğer bir örnek sonuç çıkarılacaksa bunu da bulup ortaya çıkarır. Oysa klasik programlamada kodun derlenmesi sırasında kaynak programın adım adım izlenmesi mümkündür, hatta yerine göre gereklidir. Bir YSA verilerle uğraşırken yineleme çevrimi olarak bunların nasıl öğrenileceğinin kontrolünü yapmak için de bir takım parametrelere sahiptir. İşte bu parametreler seçilen değerlerle iyi eğitilmiş bir ağı ne kadar etkili olacağını belirler. Deneyler yaparak uygun parametre değerlerinin bulunması özellikle bazı algoritmaların çok fazla hesaplama zamanı gerektirmesi nedeniyle bu ağların geliştirilmesine engel olmuştur. Öyle ki bir tek öğrenme işlemi çok güçlü bir iş istasyonunda bile saatler veya günlerce sürebilir. Bu yüzden paralel işlemcili mimarilerden yararlanmak yoluna gidilmiştir. Bilgisayar teknolojisindeki baş döndürücü gelişme bu problemi de sorun olmaktan çıkarma yolundadır.

YSA'ların yeni ortaya çıkmış bir disiplin olmasına rağmen uygulama alanları bir çok konuda hızla artmaktadır. Bazı problemlerin çözümünde oturup bir program yapmaktan daha etkin ve hızlı olabilmektedirler. Ancak bütün bunların yanında YSA'ların uygun olmadığı durumlarda söz konusudur. Bunun en büyük sebebi eğitime işleminin hedef uzayının çok geniş olması durumunda, zor olması ve uzun zaman almasıdır.

YSA kullanmak istememizin sebepleri:

1- YSA'lar verilerden hareketle, bilinmeyen ilişkileri akıllıca hemen ortaya çıkarabiliyorlar. Bu özellikleri uygulama açısından son derece önemlidir. Ayrıca veri toplama için bir ön sorgulama ya da açıklama gerekmiyor.

2- Ağlar geliştirilebilir. Bir örnekten hareketle diğer örneklerdeki benzerleri doğru olarak anlayabilirler. Genelleştirme yapılabilmesi çok önemli bir özelliğidir. Çünkü gerçek dünya verilerinde sürekli olarak gürültü ve bozucu etkiler mevcuttur.

3- YSA'ları doğrusal değildir. Bu özellikleri nedeni ile daha karmaşık problemleri lineer tekniklerden daha doğru çözerler. Doğrusal olmayan davranışlar hissedilir, algılanır, bilinebilir ancak bu davranışları ya da problemleri matematiksel olarak çözmek zordur.

4- YSA'ları son derece paralellığa sahiptir. Bağımsız işlemleri aynı anda çok hızlı yürütebilirler. Paralel donanımlar yapıları gereği YSA'ları uygun olduğundan alternatiflerinden daha elverişlidir. Piyasada elektronik olarak gerçekleştirilmiş YSA işlemcileri mevcuttur. YSA'nın belli başlı uygulama alanları olarak görüntü sınıflandırma, hedef tanıma, robot uygulamaları, risk analizi, sonar işaret sınıflandırma, metinden sese dönüştürme, modelleme, işlem izleme, kelime veya ses tanıma, uyarlamalı kanal denkleme sayılabilir.

İlk geliştirilen bazı YSA işlemcileri Çizelge 2.1'de sıralanmıştır. 2.7.4'de son yıllarda geliştirilen YSA işlemcileri üzerinde de durulacaktır.

Çizelge 2.1 İlk geliştirilen bazı YSA işlemcileri

İŞLEMCİ	YILI	İŞLEM ELE. SAYISI	AĞ SAYISI	TÜRÜ
PERCEPTRON	1957	8	1	DENEYSEL
ADALINE	1962	8	1	TİCARİ
MARK III	1985	8000	1	TİCARİ
ODYSSEY	1986	8000	1	TİCARİ
CROSSBAR	1986	256	1	DENEYSEL

2.3 Eğitim Algoritmaları

Eğitim algoritmaları YSA'nın ayrılmaz bir parçasıdır. Eğitim algoritması eldeki problemi özelliğine göre öğrenme kuralını YSA'na nasıl uyarlayacağımızı belirtir. Üç çeşit eğitim algoritması yaygın olarak kullanılmaktadır (Haykin, 1999).

- 1- Öğreticili eğitim
- 2- Skor ile eğitim
- 3- Kendini düzenleme ile eğitim

Öğreticili eğitimde elimizde doğru örnekler vardır. Yani $(x_1, x_2, \dots, x_n)$ şeklindeki giriş vektörünün, $(y_1, y_2, \dots, y_n)$ şeklindeki çıkış vektörü, tam ve doğru olarak bilinmektedir. Her bir $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ çifti için ağ doğru sonuçları verecek şekilde bir öğrenme kuralıyla beraber eğitilir.

Skor ile eğitimde giriş işaretlerine karşılık gelen çıkış işaretleri tam olarak bilinmektedir. Çıkış işareti yerine skor verilir ve alın değerlendirilmesi yapılır. Özellikle kontrol uygulamaları için idealdir.

Kendini düzenleyen ağ giriş işaretine göre kendini düzenleyerek organize eder. Olasılık yoğunluk fonksiyonlarının sınıflandırma ve şekil tanıma problemlerine uygulanabilir. Ne tür eğitim yöntemi kullanılırsa kullanılsın herhangi bir ağ için gerekli karakteristik özellik ağırlıkların verilen eğitim örneğine nasıl ayarlanacağını belirtilerek öğrenme kuralının oluşturulmasıdır. Öğrenme kuralının oluşturulması için bir örneğin ağa defalarca tanıtılması gerekebilir. Öğrenme kuralı ile ilişkili parametreler ağın zaman içinde gelişim kaydetmesiyle

değişebilir. Hangi YSA algoritmasında ne tür bir eğitime kullanıldığı bu bölümün giriş işaretlerinin sınıflandırılması kısmında gösterilmiştir.

2.4 Öğrenme Kuralları

Eşitliklerle gösterilen bu kurallar transfer fonksiyonları ve yerel bellek için kullanılır. Çeşitli YSA yapılarına göre bu kurallar seçilerek uygulanabilir.

2.4.1 YSA Konfigürasyonu

Bir yapay sinir ağı konfigürasyonunu belirleyen temel unsurlar şunlardır:

- Düğüm davranışlarının tanımlanması
- Eğitim prosedürü
- Ağ topolojisi
- Eğitim parametreleri (öğrenme kuralı)

Sonuçta öğrenme kuralları kendi başlarına bir şey ifade etmezler. Ancak bilgi bölgesi ve eğitim yöntemiyle anlam kazanırlar. Eşitliklerle gösterilen öğrenme kuralları transfer fonksiyonu ve yerel bellek için kullanılırlar. Düğüm davranışlarını açıklayabilmek için önce bilgi bölgesi yani YSA'nın veri kümesinin ve ağırlık uzayının açıklamasını yapmamız gerekir.

Bilgi Bölgesi (Hedef Uzayı): (YSA veri kümesinin oluşturulması) Her öğrenme kuralının bir bilgi bölgesi vardır. Bir YSA geliştirmede ilk ve en önemli adım ağ için gerekli olan veri kümesinin oluşturulmasıdır. Buradaki amaç ham verilerden toplanarak elde edilen verileri analiz edip değişkenleri seçerek veriyi ön işlemlerden geçirip ağın etkili bir şekilde öğrenmesini sağlamaktır. Bir çok uygulamada işin belki de 10'da 9'u veri toplamakla ilgilidir. Sonuçta her öğrenme kuralının bir bilgi bölgesi vardır.

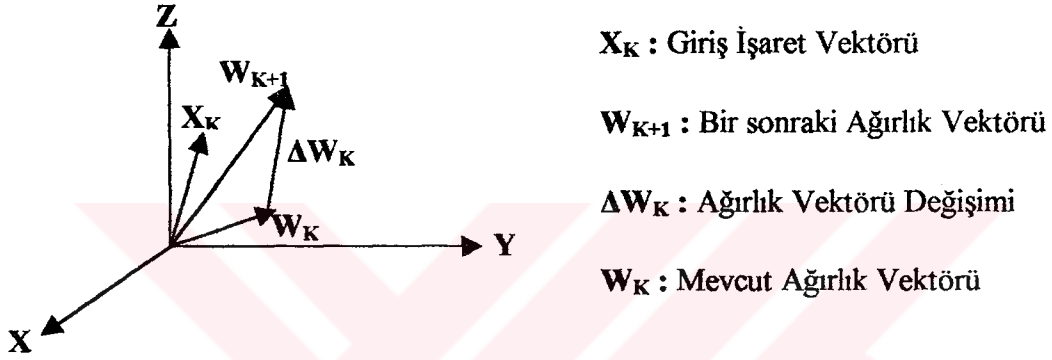
Yapay sinir ağı paralel dağılmış parametrelili bir sistem olduğundan her bir işlem elemanı izole edilmiş bir ada gibi düşünülebilir. İşlem elemanı girişine uygulanan işaretleri transfer fonksiyonuna göre değerlendirip çıkış işareti üretir.

Her işlem elemanı kendisine verilen yerel veriye göre kendisini ayarlayarak, bütün yapay sinir ağının tüm bilgi bölgesini öğrenmesini sağlar. Bilgi bölgesi olasılık yoğunluk fonksiyonu olarak tanımlanabilir. YSA paralel dağılmış parametrelili bir sistem olduğundan her bir işlem elemanı izole edilir. İşlem elemanı girişine uygulanan işaretleri transfer fonksiyonuna göre değerlendirip çıkış işareti üretir.

Ağırlık Uzayı: Bir çok YSA öğrenme işlemi, işlem elemanlarının ağırlıkları değiştirilerek sağlanır. Böylece tanımlanan ağırlıklar tanımlanacak bir hedef kritere göre değiştirilerek ağ son şeklini alır. Basit bir matematiksel model olarak, her bir işlem elemanının “n” adet gerçek ağırlığı olduğu düşünülerek N adet işlem elmanı göz önüne alınırsa

$w_1^T = (w_{11}, w_{12}, \dots, w_{1n}) \dots w_N^T = (w_{N1}, w_{N2}, \dots, w_{Nn})$ olmak üzere, $w = (w_1^T, w_2^T, \dots, w_N^T)$ 'dir. $w_1, w_2, \dots, w_N$: İşlem elemanlarının ağırlık vektörleridir.

YSA ağırlık vektörü N, n boyutlu uzayda yayılır. YSA'nın bilgi işleme performansı ağın ağırlık vektörünün belirli bir değeri ile bulunacaktır. Şekil 2.5'de ağırlıkların düzeltilmesinin vektörel çizimi verilmiştir.



Şekil 2.5 Üç boyutlu uzayda ağırlık vektörünün değişimi

Şekil 2.5'de görüldüğü gibi ΔW_K , X_K ile aynı doğrultuda olduğunda istenen hata düzeltilmesinin en küçük ağırlık değişimi ile elde edilmesi mümkündür. Böylece yeni bir giriş dizisi uygulandığında önceki eğitim dizilerinin cevabı en az bozulmuş olur. Hata değişimini inceleyen iki çeşit kural vardır.

1- Hata düzeltme kuralları

2- Gradyant kuralları

Hata düzeltme kuralları her bir giriş dizisinde ağırlıkları yeniden ayarlayarak çıktı hatasını en aza indirmeye çalışırlar. Gradyant kurallarında ise ağırlık yeniden ayarlanarak ortalama karesel hatayı (MSE) en aza indirilmeye çalışılır.

Ağırlık vektörü ile çalışan YSA'da önemli noktalardan birisi bir öğrenme kuralı geliştirip bilgi bölgesi kullanarak (Aktivasyon fonksiyonu ile) ağırlık vektörü “w” yi istenilen YSA performansı verecek noktaya yöneltmektir. Genellikle öğrenme kuralları için bir performans fonksiyonu tanımlanır. Optimizasyon ile “w” vektörü bulunur.

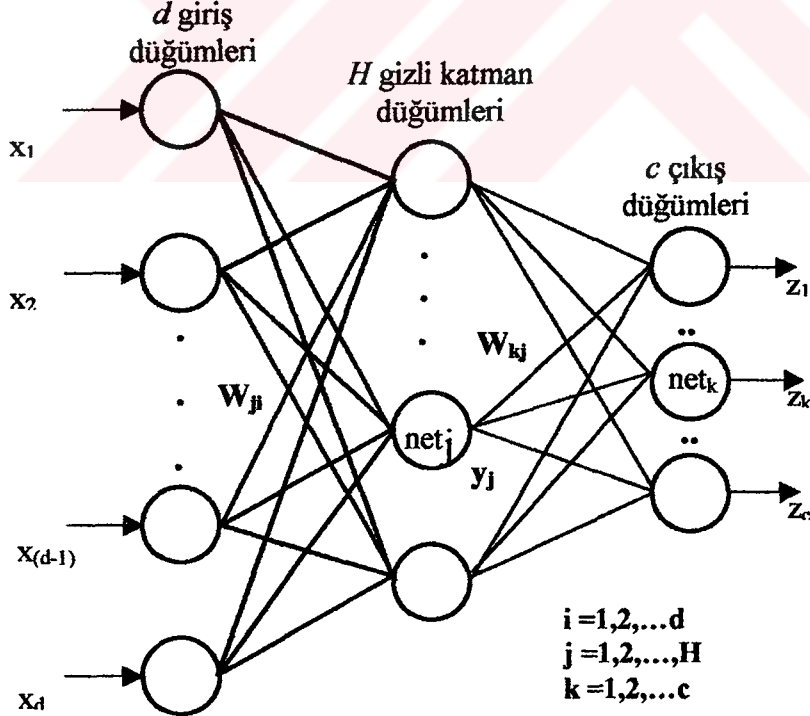
Günümüzde kullanılan bir çok öğrenme kuralı vardır. En çok kullanılan öğrenme kuralları şunlardır (Haykin, 1999):

- Rastlantısal (Hebb) öğrenme kuralı
- Performans (Widrow ve Adaline) öğrenme kuralı
- Yarışmalı (Kohonen) öğrenme
- Filtreleme (Grossberg)
- Spatiotemporal öğrenme
- Genelleştirilmiş delta kuralı öğrenme

Burada bütün öğrenme kurallarına değinilmeyecektir. Sadece tezde kullanılan “Genelleştirilmiş delta kuralı” öğrenmesi kısaca açıklanacaktır.

2.5 Çok Katmanlı Algılayıcı: ÇKA (Multi Layer Perceptron: MLP)

Çok katmanlı algılayıcı giriş ve çıkış katmanları arasında birden fazla katmanın kullanıldığı YSA sistemleridir. Gizli katman (hidden layer) olarak isimlendirilen bu katmanlarda, düğümleri aracısız giriş olmayan ve aracısız çıkış veremeyen üniteler vardır. Şekil 2.6’da çok katmanlı algılayıcı genel yapısı verilmiştir.



Şekil 2.6 Çok katmanlı algılayıcı (MLP) yapısı

2.6 Geriye Yayılma Algoritması: GYA (Back Propagation Algorithm: BPA)

Bu çalışmada kullanıldığı için üç katmanlı bir YSA'nın geriye yayılma algoritması (GYA) ile eğitilmesini göz önüne alalım. Ağdaki her bir düğüm tek bir işlem elemanıdır. Genelde daha önce de belirttiğimiz gibi düğümler katmanlarda düzenlenir. Veri girişleri giriş düğümlerinden yapılır. Giriş düğümleri pasiftir ve hesaplanabilir değildir. Giriş ile çıkış katmanları arasında kalan gizli düğümlere ağırlıklı veri düğümleri bu katmandan iletilir.

Her bir düğüm için ayrıca bir eşik (bias) değer girişi daha mevcuttur. Bu girişler düğümün kutuplaması ya da yerel belleği olarak adlandırılır. Bütün gizli katman düğümleri tüm giriş verilerini alır. Fakat her biri farklı bir ağırlık kümesine sahip olduğu için değerler kümesi de farklıdır. Her bir gizli katman düğümü kendisine gelen girişleri işleyerek sonucu çıkış katmanı düğümlerine aktarırlar. Çıkışlarda farklı ağırlık kümesine sahiptir. Geri yayılma algoritması için gizli ve çıkış katmanı düğümleri kendisine giren girişleri iki aşamada işlerler. Her giriş kendisinin ağırlığı ile çarpılır, çarpımlar toplanır. Sonra bu toplam bir fonksiyon üzerinde (Aktivasyon fonksiyonu) çıkış elde edebilmek üzere, bir sonraki katmana geçirilir. Bu transfer fonksiyonuna aktivasyon fonksiyonu denir. Aktivasyon fonksiyonları çeşitlidir. Bunlar Şekil 2.4'de gösterilmiştir. Aktivasyon fonksiyonu bir nonlinearlik oluşturur. Bu nonlinearlik ağırlık yeteneğine daha fazla güvenilmesini sağlar.

GYA ile eğitilme sırasında ağ her giriş örneğini çıkış düğümlerinde sonuç üretmek üzere gizli katmanlardaki hataları bulmak için amaçlanan hedef sonuçtan gerek sonucu elde eder. Bundan sonra ağ çıkış hatalarının türevini çıkış katmanından geriye doğru saklı katmanlara geçirirler. Bunu yaparken de orijinal ağırlık bağıntılarını kullanırlar. GYA'na ismini veren de işte bu hataların geriye doğru yayılması özelliği olmuştur.

Her çıkış ve gizli düğümlerin hata değerleri bulunduktan sonra düğümler kendi hatalarını azaltmak için kendi ağırlıklarını ayarlar. Ağırlık değiştirme denklemleri ağdaki hataların karelerinin toplamını minimize edecek şekilde düzenler. Genelleştirilmiş delta kuralı eğitime için kullanılmıştır.

2.6.1 Öğrenme ve Momentum Katsayıları

GYA'nda ele alınacak bir konu da düzgün bir öğrenme katsayısının (η) ayarlanmasıdır. Ağırlıkları çok yüksek tutmak davranışın bozulmasına neden olabilir. O nedenle öğrenme katsayısını böyle bir davranışı önlemek için küçük tutmak gereklidir. Öğrenme katsayısı, genelde $0 < \eta < 1$ aralığında seçilen sabit bir sayıdır. Öte yandan çok küçük bir öğrenme oranı da, öğrenme işleminin yavaşlamasına yol açar.

Momentum (α) fikri bu noktadan hareketle ortaya atılmıştır. Momentum mevcut delta ağırlığı üzerinden önceki delta ağırlığının belli bir kısmını besler. Böylece daha düşük öğrenme katsayısı ile daha hızlı öğrenme elde edilir. Momentum katsayısı genellikle $0 < \alpha < 1$ aralığında değişen sabit bir sayıdır.

2.7 Yapay Sinir Ağlarının Elektronik Devre Olarak Gerçeklenmesi

2.7.1 Giriş

Yapay sinir ağları şekil sınıflandırma, modelleme, işaret işleme ve kontrol gibi mühendislik problemlerini şu iki yoldan biriyle çözer.

Geleneksel yöntemlerle mukayese edildiğinde, YSA'larının kullanımı gerçek-dünya uygulamalarında sistem performansında önemli derecede bir fark sağlar veya uygulamanın maliyetinde önemli bir miktarda azalmayı performansında bir bozulma olmadan sağlar.

YSA'ları başka çözüm yolu olmayan problemleri çözme imkanı verir.

Bu şekilde YSA yaklaşımı ile mühendislik problemine başarı ile uygulanabilecek çözüm verilmiştir. İkinci basamak olarak YSA'nın donanım olarak gerçekleştirilmesi gerekir ve çalışan düzeneğin içine bir parça donanım yerleştirilmelidir. Ancak çalışan bu sistemin çalışan bir modeli olduğunda onu tamamiyle anlayabildiğimizi söyleyebiliriz. Bu noktada düşünülmesi gereken anahtar soru bir YSA'nın donanım olarak uygulanabilmesi için bu noktada fiyat olarak en ucuz ve efektif ortam nedir? Tamamiyle sayısal (dijital) yaklaşım denince akla RISC işlemcisi gelir, RISC Reduced Instruction Set kelimesinin baş harflerinden türetilmiştir (Cocke, 1990). Bu işlemci birkaç basit talimatı işletecek şekilde tasarlanmıştır (her bilgisayar saykılında –adımında- bir işlevi –komutu- yürütecek). Gerçekten günümüzün modern RISC işlemcileri çok yüksek hıza sahip olmalarından dolayı bazı uygulamalarda YSA'larının simülasyonu için yeteri kadar hızlıdır. Ancak ses tanıma, optik karakter tanıma gibi RISC işlemcileri ile sağlanması mümkün olmayan performans seviyesi gerektiren bazı karmaşık uygulamalarda yetersiz kalır. Buna ilaveten proses kontrolü adaptif-ışın şekillendirme ve adaptif gürültü yok etme gibi istenilen öğrenme hızı standart işlemcilere göre çok hızlı olan uygulamalar da vardır. Bu tip uygulamalarda VLSI (Very Large Scale Integration) devreleri kullanılmak zorundadır. Hızla gelişen bir teknolojiye sahip olması nedeniyle YSA'larının donanım olarak gerçekleştirilmelerinde ideal bir ortamdır.

VLSI teknolojisinde tek bir yarıiletken yongaya 10' milyonlarca transistor sığdırılabilmesi kapasitesi vardır. Bu kapasite gelecekte fiziksel kanunların gerçekleştirdiği limite

(Hoeneisen, 1972, Keyes, 1987) ulaşmadan önce bu miktar 100 kat daha artırılabilir. VLSI teknolojisi iki temel nedenden dolayı YSA'larına uygun bulunur.

- Yüksek yoğunluğa (birim yüzeye sığdırılan eleman sayısı) VLSI teknolojisinin imkan vermesi nedeniyle tek bir yongaya çok sayıda birbirinin aynısı eş zamanlı çalışan nöronların uygulanmasına olanak tanır. Böylelikle YSA'larının doğal paralellliğini etkin olarak tamamıyla kullanmayı mümkün kılar.
- YSA'larını düzenli topolojileri ve oldukça az sayıda iyi tanımlanan aritmetik işlemlerin eğitime algoritmalarına katılmış olması VLSI devrelerini tasarım ve üstten görünüş planı büyük ölçüde basitleştirir.

Bundan dolayı YSA'larının VLSI uygulamaları dünyada yaygın olarak büyük miktarda bir araştırma gayreti vardır. Bugün çok katmanlı algılayıcıların Boltzman makinesinin ortalama-alan teorisi makinelerinin ve kendi kendine organize olan YSA'larının gerçekleştirilmesi için genel amaçlı kırmıklara ulaşabilmektedir. Buna ilaveten özel bilgi işleme fonksiyonları için özel maksatlı çeşitli kırmıklar gerçekleştirilmiştir. VLSI teknolojisinin başarılı kullanımı enformasyon bilimleri ile nörobiyoloji arasında bir köprü kurulması şimdi sıralanacak olan yararlı etkilere sahip olacaktır. Bilgi işlemeyi daha derinden anlamayı ve geleneksel bilgisayar teknikleri ile çözülmesi çok zor olan mühendislik problemlerinin çözümü için başlangıç metodu olabilir.

2.7.2 Belli Başlı Tasarım Öğeleri

Kolay kullanım ve düşük maliyeti ve yüksek yoğunluklu olması nedeniyle CMOS transistörler YSA'larının VLSI uygulamalarında (Mead, 1989) kullanılır. YSA için geliştirmek için kullanılan kırmıkların genel amaçlı veya özel amaçlı olduğuna bakılmadan, bu teknolojiye göz önüne alınması gereken belli başlı tasarım öğeleri vardır. Özellikle aşağıda sıralanacak olan öğeler tanımlanmalıdır (Hammerstrom, 1992).

1- Çarpımların toplamının hesaplanması: Bu bütün nöronlar için gerekli fonksiyonel bir gereksinimdir. Bu YSA'ya gelen giriş data vektörünün ağırlıklarla ayrı ayrı çarpıldıktan sonra toplanmasını ifade eder. Matematiksel olarak

$$y_j = \sum_i^p w_{ij} x_i \quad (2.5)$$

şeklinde ifade edilebilir. w_{ij} , i. nöronu j. nörona bağlayan ağırlık değeridir. x_i giriş vektörü, p ağırlıkların (sinaps) sayısı, y_j , j. nöronun çıkışında oluşan aktivasyon potansiyelidir.

2- *Verinin temsili*: Genel olarak söylemek gerekirse, YSA'ların düşük hassasiyet gereksinimleri vardır. Tam bir tanımlama kullanılan algoritmaya ve uygulamaya bağlıdır.

3- *Çıkışın hesaplanması*: En çok kullanılan aktivasyon fonksiyonu yumuşak geçişli doğrusal olmayan bir fonksiyon olan sigmoid fonksiyonudur. Aşağıdaki gibi tanımlanabilir:

$$\varphi(y_j) = \frac{1}{1 + e^{-(y_j)}} \quad (2.6)$$

veya tanjant hiperbolik aktivasyon fonksiyonu;

$$\varphi(y_j) = \tanh(y_j) = \frac{e^{(y_j)} - e^{-(y_j)}}{e^{(y_j)} + e^{-(y_j)}} \quad (2.7)$$

Bu iki sigmoid tip aktivasyon fonksiyonu arasında doğrusal bir ilişki vardır. Eşik fonksiyonun da;

$$\varphi(y_j) = \begin{cases} 1 & y_j > 0 \\ 0 & y_j < 0 \end{cases} \quad (2.8)$$

yeterli olduğu durumlar söz konusu olabilir.

4- *Öğrenme karmaşıklığı*: Her öğrenme algoritmasının kendine ait hesaplanma gereksinimleri vardır. Birçok popüler öğrenme algoritmaları lokal hesaplamaları kullanarak YSA'na ait ağırlıkları değiştirip yenilemesi gerekir. Bu gerçekleşmesi açısından çok arzulanan bir uygulamadır. Bazı algoritmaların da geriye yayılma algoritmasında kullanılan hata terimleri gibi ilave gereksinimleri vardır. Bu da YSA'nın uygulamasına çok katmanlı algılayıcıların GYA ile eğitildiği durumda olduğu gibi ek bir külfet getirir.

5- *Ağırlıkların saklanması*: Eğitim esnasında ağırlıkların değiştirilmesi sırasında eski ağırlık değerlerinden hareketle yeni ağırlıkların tayin edildiğinden dolayı eski ağırlıkların da saklanması ihtiyacından kaynaklanır.

6- *İletişim*: Silikon alanında metal yolların kullanımı, kullanılan iletişim (bağlantı) linklerinin band genişliği dar olması durumunda önemli kararsızlıklara sebebiyet verir. Bağlantılar özellikle çok geniş YSA devrelerini analog ve sayısal teknolojiye uygulamak istediğimizde silikon kırmağın fabrikasyonunda belki de en ciddi sınırlamaya sebebiyet verir. Gerçekten, silikon kırmağlara çok geniş YSA devrelerine uygulamak için iletişim biçimlerine önemli yenilikler getirilmelidir. Bailey ve Hammerstrom (1988) tarafından hazırlanan bir makalede

VLSI uygulamalarındaki özellikle bağlantı problemi üzerinde durulmuştur. Bu makalede çoğullamalı (multiplexing) ana bağlantıların gerekliliği gösterilmiştir.

7- *Uygulama maliyeti*: Tüm sistemin maliyeti, YSA uygulama kırılganının gerçekleştirme aşamasında göz önüne alınmalıdır. Hesaba katılması gereken faktörler aşağıdaki gibi sıralanabilir.

- Giriş / çıkış band genişliği gereksinimleri
- Güç tüketimi
- Uygulamaların kademesi ve esnek kullanım
- Analog ve sayısal teknolojinin kullanımı

2.7.3 VLSI Uygulamalarının Kategorileri

Analog uygulamalarda giriş işaretlerinin sürekli bir genliği vardır. Sayısal uygulamalarda ise, işaretler belirli bir aralıktaki ayrık genliklere kuantalanmıştır. Bu iki yaklaşımın karışımı da (YSA'larının inşasında) kullanılan diğer yöntemin temelidir (Hibrit tekniği).

Analog Teknikler:

Analog tekniklerin hassasiyet bakımından eksikliği olmasına rağmen klasik devre teorisine ve fizik kanunlarına dayandırılan verimli, etkin hesaplamalar bu eksikliği dengeler. Analog devreler geleneksel sayısal uygulamalarda zor veya fazla zaman tüketen belirli hesaplamaları kolaylıkla yapabilir. Buna ilaveten daha az güç harcar (Mead, 1989).

Bilindiği gibi, yük taşıyıcıları olarak elektron ve delikleri kullanan, n-kanal ve p-kanal MOS (Metal Oxide Silicon) transistörlerle dayandırılan teknoloji tamamlayıcı MOS, CMOS (Complementary MOS) olarak anılır. MOS transistörünün işleyişi savak akımı I_d 'nin hesaba alınması ile anlaşılabilir. Bu da zaten geçit-kaynak geriliminin V_{gs} fonksiyonu olarak tanımlanmıştır. Savak gerilimi sabit tutulursa (örneğin 2V), iki bölge aşağıda belirtilen fonksiyonel bağımlılıkla tanımlanabilir.

-*Eşik üstü bölgesi*: Bu bölgede savak akımı I_d , V_{gs} geçit-kaynak geriliminin ikinci dereceden polinomu şeklindedir.

-*Eşik altı bölgesi*: Bu bölgede transistör düşük geçit-kaynak geriliminde çalışır ve savak akımı I_d geçit-kaynak geriliminin üstel fonksiyonudur.

Bu bölgeler hemen hemen aynı işi görür, üstel nonlinearlik tercih sebebidir. Çünkü birim akım başına daha çok iletkenliği sağlar.

MOS transistorların çok düşük güç tüketimi, YSA'ların donanım olarak VLSI teknolojisi ile analog devreler kullanılarak gerçekleştirilmesinde çok önemli bir faktördür.

Andreou tarafından tanımlanan analog yaklaşımında, en küçük yer kaplayan tasarım stili uyarlanmıştır (Andreou, 1992). Bu yaklaşımda, tek bir transistorun güçlü bir hesaplama elemanı olduğu inancı üzerine motive edilmiştir. Tasarım metodolojisi, akım-modu eşik altı CMOS devreleri üzerine dayandırılmıştır. İlgilenilen sinyaller akımlarla temsil edilir ve gerilimler daha az önemli rol üstlenirler. Akım-modu yaklaşımı mümkün olan en büyük band genişliğinde işaret işleme sağlar.

Mead tarafından tanımlanan bir analog yaklaşımda ise temel inşa bloğu olarak geçiş iletkenliği kuvvetlendiricisi kullanılmıştır (Mead, 1989).

“Toplamların çarpımını” hesaplama analog paradigmaya çok uygundur. CMOS teknolojisinde, sinaps (ağırlığın)’ın doğal seçeneği doyuma sürülmüş transistordur (Bohen, 1989). Özel olarak, transistorun geçidine uygulanan giriş gerilimi çıkışta düşük iletkenlik üretir. Bu ayarlama çok geniş fan-out’a izin verir. Son adım olarak “çarpımların toplamı” hesaplaması, Kirchhoff’un akım yasası; bu da bir sinir hücresinin çeşitli ağırlıklarına karşılık gelen çıkış savak akımlarının toplamını gerektirir.

Göz önüne alınması gereken bir diğer önemli konu, saklama (kaydetme) meselesidir. Bu gereksinim, örneğin, geçitler arası gerilim farkını kullanarak sinaptik ağırlığın kaydedilmesi (saklanması) halledilebilir.

Sonuç olarak, analog devreler, geleneksel CMOS teknolojisi veya eşik altı CMOS teknolojisine dayandırılan, yapay sinir ağlarının uygulanmasında hesaplama açısından çok etkin bir tekniktir.

Sayısal Teknikler:

Analog yaklaşıma göre sayısal yaklaşımın iki önemli avantajı vardır (Hammerstrom, 1992).

-Kolay Tasarım ve Üretim: VLSI teknolojisinin kullanımı yüksek hassasiyet, ağırlıkların kolay saklanması ve fiyat-performans avantajı sağlar. Analog VLSI’ya göre “programlanabilirlik” avantajı vardır. Buna ilaveten analoga göre sayısal silikon işleme daha da kolayca elde edilebilir.

-Esneklik: Sayısal yaklaşımın ikinci ve en önemli üstünlüğü çok daha fazla esnek olmasıdır. Bu da birçok daha kompleks algoritmanın kullanılmasına müsaade etmek suretiyle mümkün uygulama alanlarını çoğaltır. Bazı durumlarda, kompleks problemlerin çözümü yapay sinir ağı mimarisinde önemli derecede esneklik, ayarlanabilirlik gerektirir. Esneklik eksikliği analog sistemlerin gerçekten temel bir sınırlamasıdır. Özellikle karmaşıklık derecesi (teknolojinin üstesinden gelebileceği) analog teknoloji ile çözülebilecek problemlerin kademesini ve faaliyet alanını sık sık sınırlar.

Bununla birlikte, VLSI teknolojisinin çarpma işleminin sayısal uygulamalarında fazla güç tüketimine ve alana ihtiyaç gösterir. Alan ihtiyacı çoğullamalı iç bağlantı (Hammerstrom, 1992) kullanılarak giderilebilir.

Sayısal ve analog teknolojiden hangisinin kullanılacağına son kararı, yapay nöron ağı uygulaması için hangi özel algoritmanın kullanılacağı bilinmeden verilemez. Eğer genel amaçlı bir hedefimiz varsa, sayısal VLSI teknolojisine göre belirgin bir üstünlüğü vardır.

Hibrid Teknikler:

Analog hesaplamaların kullanımı YSA amaçlı VLSI'lar için, potansiyel hızları, az yer kaplamaları ve kuantalama etkilerinden uzak olmaları nedeniyle caziptirler. Diğer tarafından sayısal tekniklerin kullanımı uzun mesafeli haberleşmede sayısal işaretler kolayca iletebilmesi ve yeniden üretilmesi nedeniyle, tercih edilir. Bu düşünceler YSA uygulamalarının VLSI tekniği ile uygulanmasında hibrid yaklaşımının (hem sayısal hem analog yaklaşımı kullanan) kullanımını cesaretlendirmiştir. Zaten bu yaklaşım da her iki yaklaşımın yararlı tarafları üzerinde bina edilmiştir (Murray, 1991).

YSA'larının VLSI tekniğinde kullanılan hibrid tekniğine örnek olarak çarpıcı olarak kullanılan MDAC (Multiplying digital-to-analog converters) dir. Bu teknikte, analog durum (giriş sinyali gibi) Belcore kırmığında (çipinde) olduğu gibi (Alspector, 1991) sayısal ağırlıkla veya sayısal durum AT&T ANNA kırmığında olduğu gibi (Sackinger, 1992) analog ağırlıkla çarpılabilir. Bu nedenle MDAC'lar YSA tasarımcılarına belirli bir hesaplama problemini, analog ve sayısal teknolojileri optimal bir şekilde kombine ederek çözmeye izin verir.

2.7.4 YSA Donanımlarına Kısa Bir Bakış

YSA'nın VLSI uygulamalarına analog, sayısal ve hibrid yaklaşımlar açısından baktıktan ve avantaj ve dezavantajlarını belirledikten sonra artık YSA donanımlarına göz atabiliriz. Genel ve özel amaçlı nörobilgisayarlar kırmık ve sistemlerinin halihazırda elimizin altında olan listesi oldukça çeşitlilik göstermekte ve halen genişlemektedir. Bu da YSA'larının kullanıcı topluluğu tarafından ne kadar hızlı bir kabul gördüğünün göstergesidir. Genel amaçlı kırmık ve sistemleri; ETANN analog kırmık (Holler, 1989), Edinburgh Üniversitesi EPSILON hibrid kırmığı (Murray, 1991), Adaptif çözümler CNAPS sayısal sistemi (Hammerstrom, 1990), Siemens Sayısal sinyal işlemcisi (Ramacher, 1991), Mitsubishi BNU sayısal kırmığı (Arima, 1991), Hitachi sayısal kırmığı (Watanabe, 1993), Bellcore Boltzman ortalama-alan öğrenme kırmığı (Alspector, 1992), ve AT&T Bell Labaratuarı ANNA kırmığı (Boser, 1992) olarak sayılabilir. Özel amaçlı kırmıklar Sinaptik OCR kırmığı, Kohonen'in kendi kendine organize olan haritası yaklaşımının analog uygulaması olan ve kırmık üzeri öğrenme imkanı da veren kırmık (Macq, 1993), Video makineleri algılaması için VLSI işlemcileri (Lee, 1993), Haberleşme alıcıları için programlanabilir analog VLSI YSA (Choi, 1993) ve analog-sayısal dönüştürücüler için çok seviyeli YSA kırmığı (Yuh,1993) şeklinde sıralanabilir.

3. FGMOS TRANSİSTORLAR

3.1 Tarihsel Gelişim

-1967 yılında Khang ve Sze ilk “yüzen-geçit” (floating-gate) yapısını değişken olmayan bilgiyi saklamaya yarayan bir mekanizma olarak ortaya atmıştır (Khang, 1967).

Bu zamandan sonra FGMOS transistörler, uzun süre boyunca dijital bilgiyi saklamada EPROM’lar, EEPROM’lar ve Flash Memory’ler gibi yapılarda yaygın olarak kullanılmıştır. FGMOS transistörler aynı zamanda analog uygulamalarda sabit bilgiyi uzun süre saklamak için de kullanılmıştır.

-1983’de Wada ve arkadaşları “dual-control-gate EEPROM” olarak bilinen yeni bir EEPROM adresleme yapısı tanımladılar. Burada iki kontrol geçidi transistörün yüzen-geçit’ine eşit ağırlıklı olarak bağlıdır (Hieda, 1983).

-1992’ye gelindiğinde ise, Shibata ve Ohmi FGMOS transistörler hakkında yeni bir müthiş düşünce tarzı ortaya atmışlardır -Tohoku Üniversitesi-(Shibata, 1992).

Burada çoklu kontrol girişleri yüzen geçit’e kapasitif etkiyle bağlanarak, bir kapasitif gerilim bölücü üzerinden giriş gerilimlerinin ağırlıklı toplamaları şeklinde yüzen geçit gerilimi sağlanmış olur. Bu yüzen geçit gerilimi, yüzen geçit’in altındaki silikonun içindeki kanaldan akan akımı yani bir MOS transistörün içinden akan akımı ayarlar. Shibata ve Ohmi, bu birleşik elemana **neuron MOS (neuMOS veya vMOS) transistör** adını takmışlardır. neuMOS transistörler ve sinir sistemindeki hücreler tarafından gerçekleştirilen fonksiyonların benzerliğinden dolayı bu ad verilmiştir.

Yang, Andreou ve Boahen böyle elemanlara çok-girişli FGMOS adını vermektedir (Yang, 1994). Ramirez-Angulo, bunlara çok-girişli yüzen-geçitli (MIFG) transistörler demektedir (Ramirez, 1996).

Shibata ve Ohmi tarafından FGMOS transistörler hakkında ortaya atılan bu düşünce tarzı, çok sayıda ilginç analog ve dijital bilgi işleme devrelerinin gerçekleştirilmesine olanak sağlamıştır. Bunlardan rastlanan örnekler şöyle sıralanabilir; basit D/A çeviriciler, dört bölgeli çarpıcılar, eşik lojik birimlerine dayanan yeniden biçimlendirilebilen sayısal lojik, sinir ağları, ‘winner-take-all’ devreleri, FGMOS geçiş iletkenliği devreleri. Özetle, Shibata ve Ohmi’nin kavramsal geliştirmesi demidir ki; FGMOS transistörler sadece bilgi depolamaktan ibaret değildir, aynı zamanda bilgi işleme aygıtları olarak da kullanılabilir.

3.2 Biyolojik Nöron İle Yüzen Geçitli-Çok Girişli MOS Transistorun İşlevsel Benzerliği

McCulloch-Pits'in basitleştirilmiş biyolojik nöron modelinde nöron, girişlerine (sinapslarına) uygulanan işaretlerin ağırlıklandırılmış toplamı belli bir "t" eşik değerini aştığında, çıkışını "lojik 0" kabul edilen seviyeden "lojik 1" olarak belirlenmiş seviyeye anahtarlayabilen birim olarak tanımlanmıştır. Yani, x_i yalnızca "0" ve "1" değerlerini alabilecek n-adet giriş, w_i bu girişlere ait ağırlıklar ve y de çıkış olmak üzere, bir nöronun işlevi aşağıdaki bağıntıyla modellenmiştir (Werner, 1996):

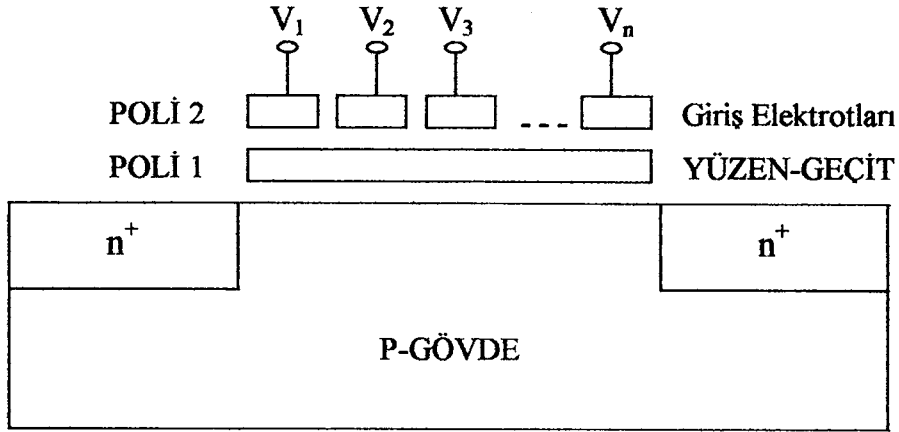
$$y = \begin{cases} 1, & \sum_{i=1}^n x_i w_i > t \text{ ise} \\ 0, & \text{aksi halde} \end{cases} \quad (3.1)$$

Bunu izleyen bölümlerde, bir yüzen geçitli-çok girişli MOS transistorda, (3.1) denklemindeki "y" büyüklüğüne transistorun savak akımının, " x_i " büyüklüğüne transistorun i'nci giriş geriliminin, " w_i " ağırlığına i'nci girişle yüzen geçit arasındaki kapasitenin yüzen geçide bağlı toplam kapasiteye oranının, "t"ye ise transistorun yüzen geçidinden görülen ve V_T ile sembolize edilen eşik geriliminin karşılık geldiği görülecektir. Açıktır ki analog uygulamalarda herhangi bir değer alabilecek giriş işaretlerinin ağırlıklandırılmış toplamı, transistorun "açık" ve "kapalı" konumlarını kontrol etmenin dışında, savak akımının miktarını da belirleyecektir.

İşte çalışma biçiminin, (3.1) bağıntısına uygunluğu dolayısıyla yüzen geçitli-çok girişli MOS transistora "nöron MOS transistor" veya kısaca "vMOS" adı verilmiştir (Shibata, 1992).

3.3 N-Kanallı FGMOS Transistor (vMOS) Yapısı

Basitleştirilmiş kesidi Şekil 3.1'de verilmiş olan n-kanallı FGMOS transistor, geçidi "yüzen" bırakılmış bir NMOS transistor ve bu yüzen geçit üzerinde, ikinci polisilisyum tabakasıyla oluşturulmuş ve yüzen geçitle kapasitif etki oluşturan birden fazla girişten oluşur. Giriş ağırlıklarını gerçekleyen poli1-poli2 kapasitelerinin bir kısmı veya tamamı alan oksidi üzerinde de oluşturulabilir (Werner, 1996). N-kanallı FGMOS transistorun sembolü ve layout gösterimi Şekil 3.2'de verilmiştir.



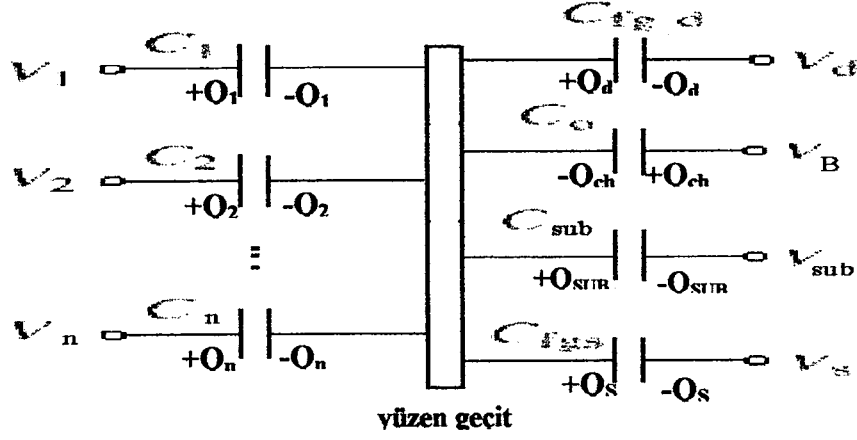
Şekil 3.1 N-girişli n-kanallı FGMOS transistorun basitleştirilmiş kesidi



Şekil 3.2 N-girişli n-kanallı FGMOS transistorun sembolü ve layout gösterimi

3.4 FGMOS Transistorun Çalışma İlkesi ve Analitik Bağntılar

Bir FGMOS transistorda yüzen geçit gerilimi, giriş uçları ile transistorun yüzen geçit altında kalan kısmının, yüzen geçide olan kapasitif oranı ile belirlenir. FGMOS transistorun çalışması, geçidin bu şekilde kutuplanması dışında, sıradan bir MOS transistorla aynıdır. N-girişli bir n-kanallı FGMOS transistorda, bir ucu yüzen geçitte olan kapasiteler ve üzerlerinde biriken yükler Şekil 3.3'te gösterilmiştir.



Şekil 3.3 N-girişli bir vNFGMOS transistorda yüzen geçide bağlı kapasiteler ve biriktirdikleri yükler

Şekil 3.3'te, $C_1, C_2, \dots, C_n$ poli2 ile poli1 arasındaki giriş kapasitelerini; C_{FS} , C_{FD} ve C_{FSUB} yüzen geçidin kaynak, savak ve *aktif olmayan* taban bölgesiyle örtüşme kapasitelerini; C_o da geçit oksit kapasitesini göstermektedir.

Başlangıçta yüzen geçitte depolanmış yük Q_T olsun. N-kanallı FGMOS transistora $V_1, V_2, \dots, V_n$ giriş gerilimleri ve V_D, V_S, V_B ve V_{SUB} savak, kaynak, gövde ve aktif olmayan taban gerilimleri uygulandığında yüzen geçide bağlı kapasitelerde biriken yükler Şekil 3.3'te belirtildiği gibi olacak ve yükün korunumu ilkesi gereği, bu yükün ilk değeri (Q_T) ve son değeri birbirine eşit olacaktır:

$$Q_T = \left(\sum_{i=1}^n -Q_i \right) + Q_d + Q_s + Q_{sub} - Q_{ch} \quad (3.2)$$

Yüzen geçit gerilimi V_{FG} ile gösterilir ve Q_{ch} dışındaki yüklerin yerine, ilgili kapasite değeri ile kapasite üzerindeki gerilimin çarpımı yerleştirildiğinde (3.3) denklemi elde edilir.

$$Q_T = \left(\sum_{i=1}^n -C_i(V_i - V_{FG}) \right) + C_{FD}(V_{FG} - V_D) + C_{FS}(V_{FG} - V_S) + C_{FSUB}(V_{FG} - V_{SUB}) - Q_{ch} \quad (3.3)$$

(3.3) denkleminde görüldüğü üzere eğer kanal yükü Q_{ch} de V_{FG} 'nin bir fonksiyonu olarak ifade edilebilirse, yüzen geçit gerilimi V_{FG} transistorun uç gerilimleri ve kapasite değerleri cinsinden belli olacaktır. Buradan da, bir n-kanallı FGMOS transistorun akımını, sıradan bir NMOS transistorun akım denkleminde, geçit gerilimi V_G 'nin yerine V_{FG} 'nin ifadesini koyarak bulmak mümkün hale gelecektir.

3.4.1 Doymada Çalışan n-kanallı FGMOS Transistorda Kanal Yükünün Yüzen Geçit Gerilimi Cinsinden İfadesi

Bir n-kanallı FGMOS transistorda, kaynak noktası orijin kabul edilerek kaynaktan savağa kanal boyunca uzanan bir y eksenini göz önüne alınsın. Transistörün lineer bölgede kutuplandığı, dolayısıyla savaktan kaynağa kadar, iletken bir kanalın oluştuğu kabulü ile, kanalın herhangi bir y noktasındaki birim alan başına yük $Q_n(y)$, -fakirleşmiş bölge yükünün kanal boyunca değişimi ihmal edilir ve yalnızca kaynaktaki değeri hesaba katılırsa- birim alan başına geçit oksidi kapasitesi C_{ox} ile, bu kapasitenin uçlarındaki gerilimin, yük birikimine neden olan “efektif” değeriyle çarpımına eşittir (Sin, 1992).

$$Q_n(y) = -C_{ox}[(V_{FG} - V_T) - (V_S + V_c(y))] \quad (3.4)$$

Burada $V_c(y)$ kanal gerilimi olup sınır değer koşulları “ $V_{c(y=0)} = V_S = 0$ ” ve L transistörün kanal uzunluğu olmak üzere, “ $V_{c(y=L)} = V_{DS}$ ” olarak seçilmiştir. Ayrıca analiz boyunca gövde gerilimi için de $V_B = 0$ alınacaktır.

Kanal toplam yükü, $Q_n(y)$ 'nin kanal boyunca integralinin, kanal genişliği W ile çarpımına eşittir.

$$Q_{ch} = W \int_0^L Q_n(y) dy \quad (3.5)$$

(3.5) integralini alabilmek için, integral değişkeni y'yi (3.4)'teki V_c cinsinden ifade etmek gerekir. Bu amaçla önce dy uzunluğundaki kanal bölgesinin direnci (dR) bulunacak, bunun savak akımı I_D ile çarpımı dV_c 'yi verecektir. Bir iletkenin direnci, iletkenin özgül direnci ile iletkenin uzunluğunun çarpımının kesit alanına oranına eşittir:

$$dR = \frac{1}{q \mu_n n} \times \frac{dy}{W x} \quad (3.6)$$

Bu denklemde q, birim elektrik yükünü, μ_n kanaldaki ortalama elektron mobilitesini, n birim hacimdeki yük adedini, x ise y noktasında kanalın derinliğini göstermektedir. Öte yandan birim hacimdeki yük adedi, birim alan başına elektron yükü yoğunluğu cinsinden;

$$n = -\frac{Q_n(y)}{q x} \quad (3.7)$$

ile gösterilir. n'nin (3.7)'deki eşitini (3.6)'ya yerleştirip I_D ile çarpımı alındığında;

$$dV_C = -\frac{I_D}{W \mu_n Q_n(y)} dy \quad (3.8)$$

elde edilir. (3.8)'de dy yalnız bırakılıp (3.4) ile beraber (3.5)'e yerleştirildiğinde aşağıdaki integrale ulaşılır:

$$Q_{ch} = -\frac{W^2 \mu_n C_{ox}^2}{I_D} \int_0^{V_{DS}} (V_{FG} - V_S - V_T - V_C)^2 dV_C \quad (3.9)$$

Bu integral alındığında;

$$Q_{ch} = -\frac{W^2 \mu_n C_{ox}^2}{I_D} \left[(V_{FG} - V_S - V_T)^2 V_{DS} - (V_{FG} - V_S - V_T) V_{DS}^2 + \frac{V_{DS}^3}{3} \right] \quad (3.10)$$

bulunur. Transistor doymaya girdiği anda –ki bu anda kaynaktan savağa kadar kanal halen mevcuttur- aşağıdaki denklemler geçerlidir:

$$V_{DS} = V_{FG} - V_S - V_T \quad (3.11a)$$

$$I_D = \frac{1}{2} \frac{W}{L} \mu_n C_{ox} (V_{FG} - V_S - V_T)^2 \quad (3.11b)$$

Bu iki denklem (3.10)'da yerlerine konduğunda, transistor doymaya girdiği andaki kanal yükünü yüzen geçit gerilimine doğrusal olarak bağlayan denklem elde edilmiş olur (Sin, 1992):

$$Q_{ch} = -\frac{2}{3} W L C_{ox} (V_{FG} - V_S - V_T) \quad (3.12)$$

Transistor doymada çalışırken kanalın, $y = L' (< L)$ 'deki ucunun gerilimi V_{Dsat} 'a eşit olacağından ve yine (3.11a) bağıntısını sağlayacağından (3.12) denklemini transistorun tüm doyma rejimi için geçerlidir.

3.4.2 Doymada Çalışan n-kanallı FGMOS Transistorun Akım Bağıntısı

(3.12) bağıntısı (3.3)'te yerine konarak V_{FG} uç gerilimleri, kapasiteler ve yüzen geçidin başlangıç yükü cinsinden ifade edilebilir:

$$V_{FG} = \frac{Q_T + \left(\sum_{i=1}^n C_i V_i \right) + C_{FSUB} V_{SUB} + C_{FD} V_D + \left(C_{FS} + \frac{2}{3} W L C_{ox} \right) V_S + \frac{2}{3} W L C_{ox} V_T}{C_T} \quad (3.13)$$

Burada C_T :

$$C_T = \left(\sum_{i=1}^n C_i \right) + C_{FSUB} + C_{FD} + C_{FS} + \frac{2}{3} W L C_{ox} \quad (3.14)$$

ile belirlidir. (3.13), (3.11b)'de yerine konduğunda, doymada çalışan n-kanallı FGMOS transistorun akım bağıntısı –kanal kısılması ihmal edilerek- aşağıdaki formülle verilmiş olur:

$$I_D = \frac{1}{2} \frac{W}{L} \mu_n C_{ox} \left\{ \frac{Q_T}{C_T} + \frac{\sum_{i=1}^n C_i V_i}{C_T} + \frac{C_{FD}}{C_T} V_D + \frac{C_{FSUB}}{C_T} V_{SUB} - \left[1 - \left(\frac{C_{FS}}{C_T} + \frac{2WLC_{ox}}{3C_T} \right) \right] V_S - \left(1 - \frac{2WLC_{ox}}{3C_T} \right) V_T \right\}^2 \quad (3.15)$$

Başlangıç yükü Q_T sıfır alınır ve transistorun çalışması sırasındaki yük enjeksiyonu ihmal edilirse, $(W / L) \mu_n C_{ox} = K_n$, $C_i / C_T = k_i$, $C_{FSUB} / C_T = k_{SUB}$, $C_{FD} / C_T = k_D$, $[1 - (C_{FS} / C_T) - (2 / 3)(WLC_{ox} / C_T)] = k_S$, $[1 - (2 / 3)(WLC_{ox} / C_T)] = k_T$ alındığında (3.15) bağıntısı aşağıdaki şekli alır:

$$I_D = \frac{1}{2} K_n \left[\left(\sum_{i=1}^n k_i V_i \right) + k_{SUB} V_{SUB} + k_D V_D - k_S V_S - k_T V_T \right]^2 \quad (3.16)$$

Başlangıç yükü Q_T 'nin sıfır olduğu ve örtüşme kapasitelerinin ihmal edilebildiği yani, $C_{FSUB} \cong 0$, $C_{FS} \cong 0$, $C_{FD} \cong 0$ alınabilen durumlarda, yüzen geçide bağlı toplam kapasite C_T ;

$$C_T = \left(\sum_{i=1}^n C_i \right) + \frac{2}{3} W L C_{ox} \quad (3.17)$$

olmak üzere ve k_i 'nin yine yukarıdaki eşitliği sağlamasıyla yüzen geçit potansiyel;

$$V_{FG} = \left[\sum_{i=1}^n k_i (V_i - V_S - V_T) \right] + V_S + V_T \quad (3.18)$$

denklemleriyle belirli olup bunun (3.11b)'de yerine konmasıyla doymada çalışan n-kanallı FGMOS'un akım bağıntısı;

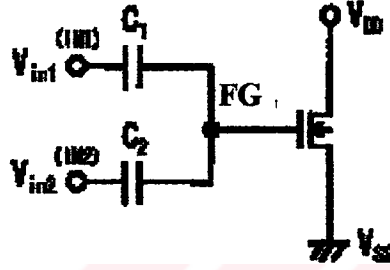
$$I_D = \frac{1}{2} K_n \left[\sum_{i=1}^n k_i (V_i - V_S - V_T) \right]^2 \quad (3.19)$$

ifadesine dönüşür.

3.5 FGMOS Makro Modeli

3.5.1 FGMOS Transistor Simülasyonu İçin Makro Modelin Gerekliliği

Girişlerine V_1 ve V_2 işaretleri, savağına besleme gerilimi V_{DD} uygulanmış, kaynak ucu toprakta olan 2-girişli bir n-kanallı FGMOS transistorun eşdeğeri Şekil 3.4’de verilmiştir (Ochiai, 1999). Poli1-poli2 arası giriş kapasiteleri C_1 ve C_2 ile, yüzen geçidin bulunduğu düğüm de FG ile gösterilmiştir.

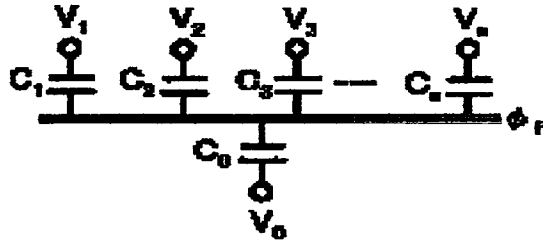


Şekil 3.4 2-girişli n-kanallı FGMOS transistorun eşdeğer devresi

FG düğümünden toprağa DC bir yol bulunmadığından SPICE, FG düğümünü yüzen bir düğüm olarak algılamakta ve Şekil 3.4’deki eşdeğer devreyi simüle edememektedir. Dolayısıyla yüzen geçit potansiyelini girişler ve transistorun diğer uç gerilimleri cinsinden hesaplayan bir modele ihtiyaç vardır.

3.5.2 Ochiai-Hatano Makro Modeli

Ochiai ve Hatano’nun önerdiği makro model (Ochiai, 1999), n-girişli n-kanallı FGMOS transistorun Şekil 3.5’teki basitleştirilmiş kapasitif modeline dayanır.



Şekil 3.5 FGMOS transistorun, Ochiai-Hatano makro modeline temel oluşturan kapasitif modeli (Ochiai, 1999)

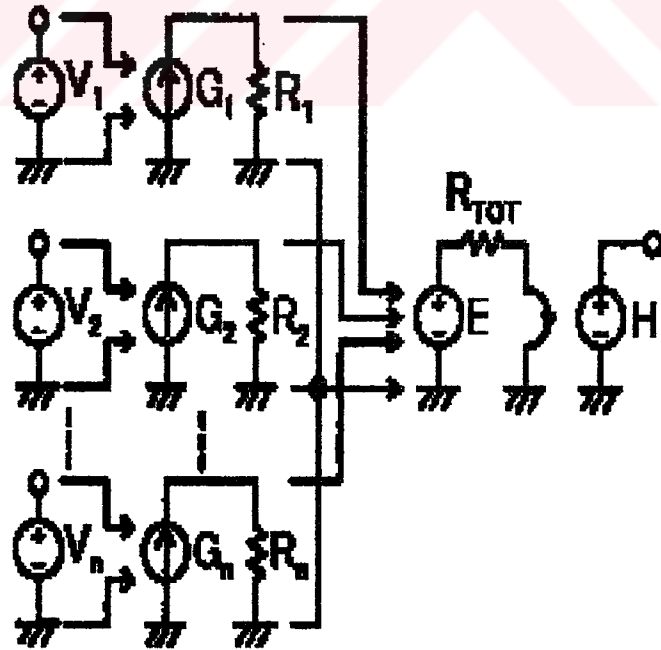
Burada, Şekil 3.3'den farklı olarak, örtüşme kapasiteleri (C_{FSUB} , C_{FS} , C_{FD}) ihmal edilerek gösterilmemiştir. Transistorun çalışma rejimine göre yüzey geriliminde meydana gelecek değişimlerin ihmal edilerek yüzey geriliminin her zaman gövde gerilimine eşit olduğu kabulü (ki bu aslında sadece, transistor kesimdeyken böyledir) ve gövde geriliminin de sıfır alınması ile, -başlangıç yükü de sıfır alınarak- yük denklemi yazılır ve buradan kapasite ve uç gerilimlerine geçilirse yüzen geçit gerilimi için (3.20) bağıntısı elde edilir.

$$V_{FG} = \frac{\sum_{i=1}^n C_i V_i}{C_T} \quad (3.20)$$

Burada C_T ;

$$C_T = \sum_{i=0}^n C_i \quad (3.21)$$

ile ifade edilir ve transistor doymadaysa kanal kısılmasının ihmalı ile (3.17) denklemini sağlar. N-girişli FGMOS transistorun yüzen geçit potansiyelini simüle eden Ochiai-Hatano makro modeli Şekil 3.6'da verilmiştir (Ochiai, 1999).



Şekil 3.6 Ochiai-Hatano n-girişli FGMOS makro modeli (Ochiai, 1999)

Şekil 3.6'da G_i , gerilim kontrollü akım kaynağı olup,

$$G_i = V_i \quad (i = 1, 2, \dots, n) \quad (3.22a)$$

ile verilir. E , gerilim kontrollü polinomsal gerilim kaynağıdır ve dirençlerin uçlarındaki gerilimlerin toplamı kadar gerilim sağlar:

$$E = \sum_{i=1}^n G_i R_i \quad (3.22b)$$

H , akım kontrollü gerilim kaynağı olup R_T direncinden akan akıma göre gerilim üretir:

$$H = \frac{E}{R_T} \quad (3.22c)$$

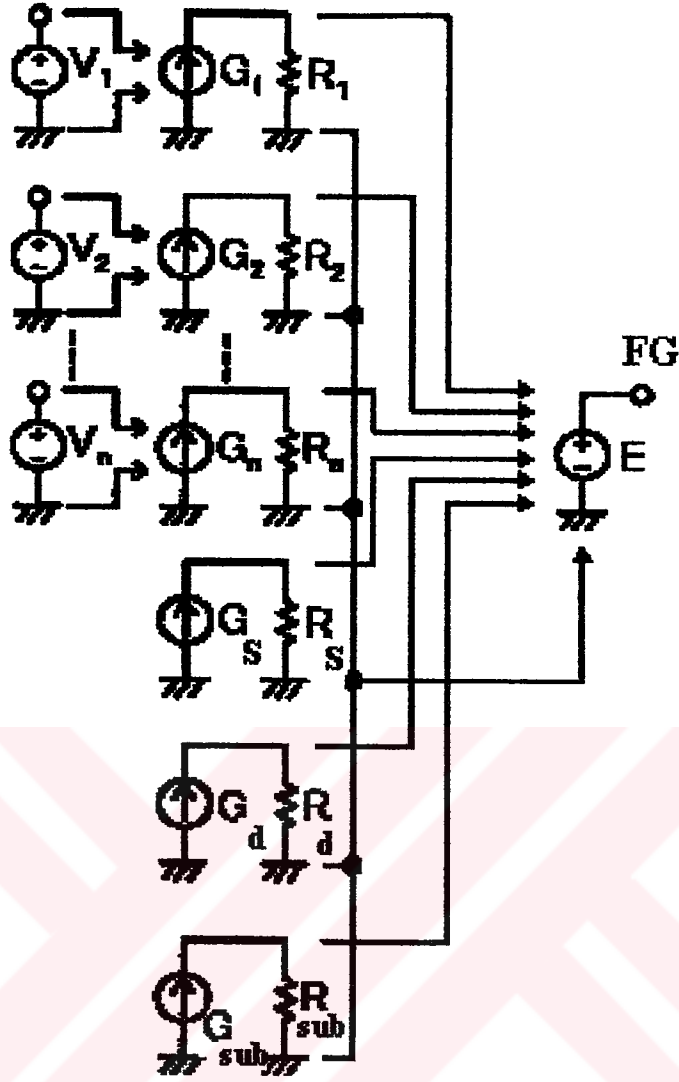
(3.21) denklemlerini bir araya getirip H 'nin yüzen geçit gerilimini sağladığını da göz önüne alarak:

$$H = \frac{\sum_{i=1}^n R_i V_i}{R_T} \quad (3.22d)$$

bağıntısına ulaşılır. SPICE kodu girilirken R_i yerine C_i kapasite değerleri, R_T yerine C_T toplam kapasitesinin değeri yazılıp, H kaynağı da, Şekil 3.6'da gösterildiği gibi sıradan MOS transistorun geçidine bağlandığında (3.22d) denkleminin (3.20) denklemiyle aynı sonucu vermesi sağlanmış olur.

3.5.3 (3.13) Denklemine Dayalı Bir Makro Modelin Oluşturulması

Yüzen geçit gerilimini veren (3.13) ve (3.20) denklemleri karşılaştırıldığında, n -kanallı FGMOS transistorun geometrisine ve kutuplama şartlarına bağlı olarak, (3.20) denkleminde ihmal edilen terimlerin, yüzen geçit geriliminin hesabında önemli hatalara yol açabileceği görülür. Böylesi durumlar için (3.13) denklemini temel alan bir makro modele ihtiyaç vardır. Bu amaca yönelik bir makro model Şekil 3.7'de verilmiştir.



Şekil 3.7 (3.13) bağıntısını gerçekleyen makro model

Burada V_s , V_D ve V_{SUB} ; kaynak, savak ve aktif olmayan taban gerilimleri olmak üzere; gerilim kontrollü akım kaynakları ve dirençler için;

$$G_i = V_i \quad (i = 1, \dots, n) \quad (3.23a)$$

$$G_s = V_s \quad (3.23b)$$

$$G_D = V_D \quad (3.23c)$$

$$G_{SUB} = V_{SUB} \quad (3.23d)$$

$$R_i = \frac{C_i}{C_T}, \quad (i = 1, \dots, n) \quad (3.24a)$$

$$R_S = \frac{C_{FS} + \frac{2}{3} W L C_{ox}}{C_T} \quad (3.24b)$$

$$R_D = \frac{C_{FD}}{C_T} \quad (3.24c)$$

$$R_{SUB} = \frac{C_{FSUB}}{C_T} \quad (3.24d)$$

eşitlikleri geçerlidir.

C_{FS} , C_{FD} ve C_{FSUB} kapasiteleri Bölüm 3.4 Şekil 3.3'de tanımlandığı gibi olup C_T (3.14) denklemiyle verilmiştir. Yüzen geçit gerilimine eşit olan E polinomsal kaynağının gerilimi;

$$E = P_0 + \sum_{i=1}^n R_i V_i + R_S V_S + R_D V_D + R_{SUB} V_{SUB} \quad (3.25)$$

bağıntısını sağlar. Polinomun sabit katsayısı P_0 , gövde etkisi olmadığı varsayımıyla;

$$P_0 = \frac{2/3 W L C_{ox}}{C_T} V_T + \frac{Q_T}{C_T} \quad (3.26)$$

şeklinde seçilirse yüzen geçit gerilimi için de (3.13) gerçekleşmiş olur. Bu arada, (3.24) denklemlerinin sonucu olarak direnç değerleri arasında

$$\left(\sum_{i=1}^n R_i \right) + R_S + R_D + R_{SUB} = 1 \quad (3.27)$$

ilişkisi vardır.

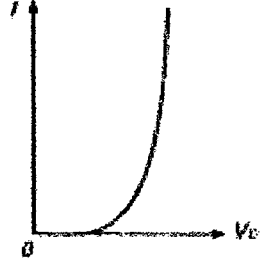
3.6 Dört-Uçlu Elemanların Esnek Yapısı, İki-Uçlu Ve Üç-Uçlu Elemanlarla Karşılaştırılması

Daha esnek bir donanım yapısına sahip olmak için temel eleman seviyesinde fonksiyonel geliştirme şarttır. Bu nedenle, üç-uçlu elemanlar olarak bilinen MOS transistörlerden ve bipolar transistörlerden daha üstün işlevsel özelliklere sahip bir dört-uçlu eleman kavramı ileri sürülmüştür (Ohmi, 1997).

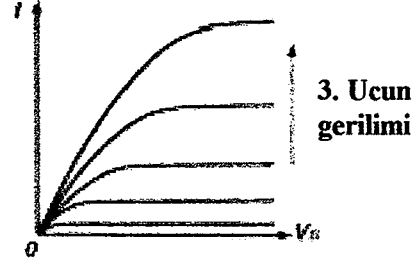
İki-uçlu elemanlardan dört-uçlu elemanlara kadar olan fonksiyonel gelişmeyi Şekil 3.8 özetler. İki-uçlu eleman olan bir diyotta, üzerinden geçen akımı sadece uçları arasındaki gerilim farkı belirler. Böyle ilkel bir çalışma biçiminden ancak gelişmemiş, karmaşık olmayan bir uygulama ortaya çıkar. MOS transistörler ve bipolar transistörler gibi üç-uçlu elemanlarda ise, iki ana uçtan akan akım üçüncü bir ucun gerilimi tarafından kontrol edilebilir. Bunun sonucu olarak, akımın kontrol edilebilirliği iki-boyutlu bir ortama genişletilmiştir. Bu ise, kuvvetlendiricileri, osilatörleri ve sayısal hesaplama sistemleri gibi sistemleri içeren elektronik devreler kavramını geliştirmeye izin verir. 20. yüzyıldaki elektronik üç-uçlu elemanlar tarafından oluşturulduğundan çok fazla bir şey söylemeye gerek kalmaz.

Biliyoruz ki; üç-uçlu elemanda (MOS transistör) savak akımı, şekil3.9'da gösterildiği gibi savak geriliminin sabitlenmesiyle sadece üçüncü ucun gerilimi (geçit-kaynak gerilimi) tarafından belirlenir. Doğrusal olmayan akım-gerilim karakteristiği lojik bir yapı oluştururken bir akım anahtarlama gibi kullanılabilir. Bu, katı ikili sayısal algoritma için çok uygundur fakat esneklikten çok uzaktır. Bir dört-uçlu elemanda akım aynı koşullar altında tek başına belirlenmez çünkü üçüncü uç tarafından kontrol edilen akımın davranış şekli ek olarak dördüncü uçla daha fazla kontrol altına alınabilir. Başka bir deyişle, dört-uçlu elemanda akımı kontrol altına almada serbestlik derecesi şekil 3.8'de gösterildiği gibi üç boyutlu bir ortama taşınmıştır. Donanım üzerinde esnek bir bilgi işleme tasarımı gerçekleyebilmek için temel elemanda böyle bir fonksiyonel geliştirme şarttır.

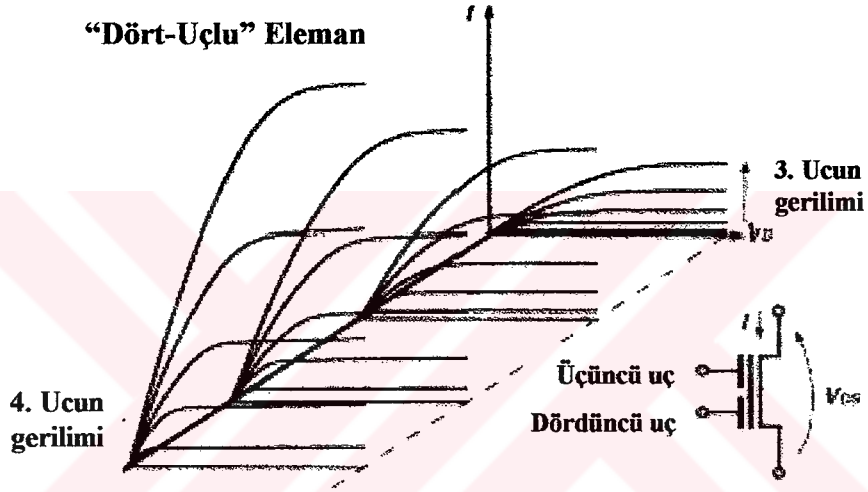
“İki-Uçlu” Eleman



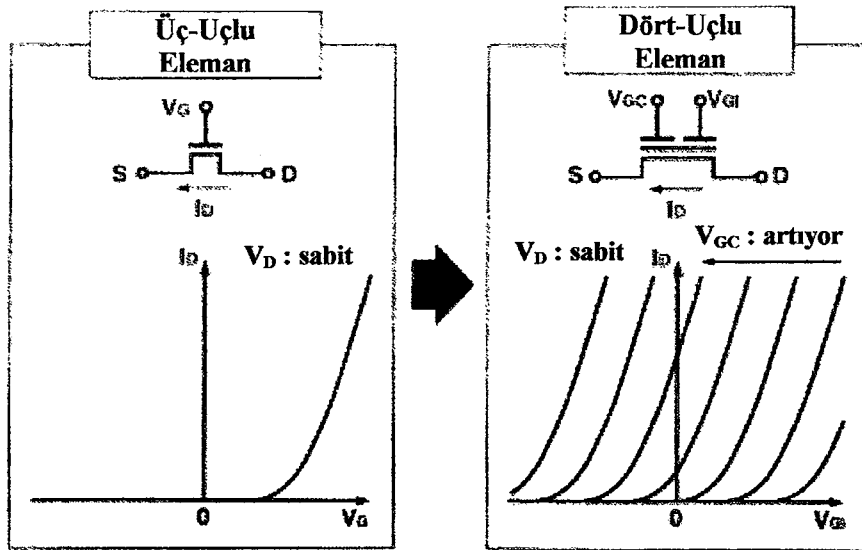
“Üç-Uçlu” Eleman



“Dört-Uçlu” Eleman



Şekil 3.8 Temel elemanlarda iki-uçlu elemandan dört-uçlu elemana doğru işlevselliğin geliştirilmesi (Ohmi, 1997)



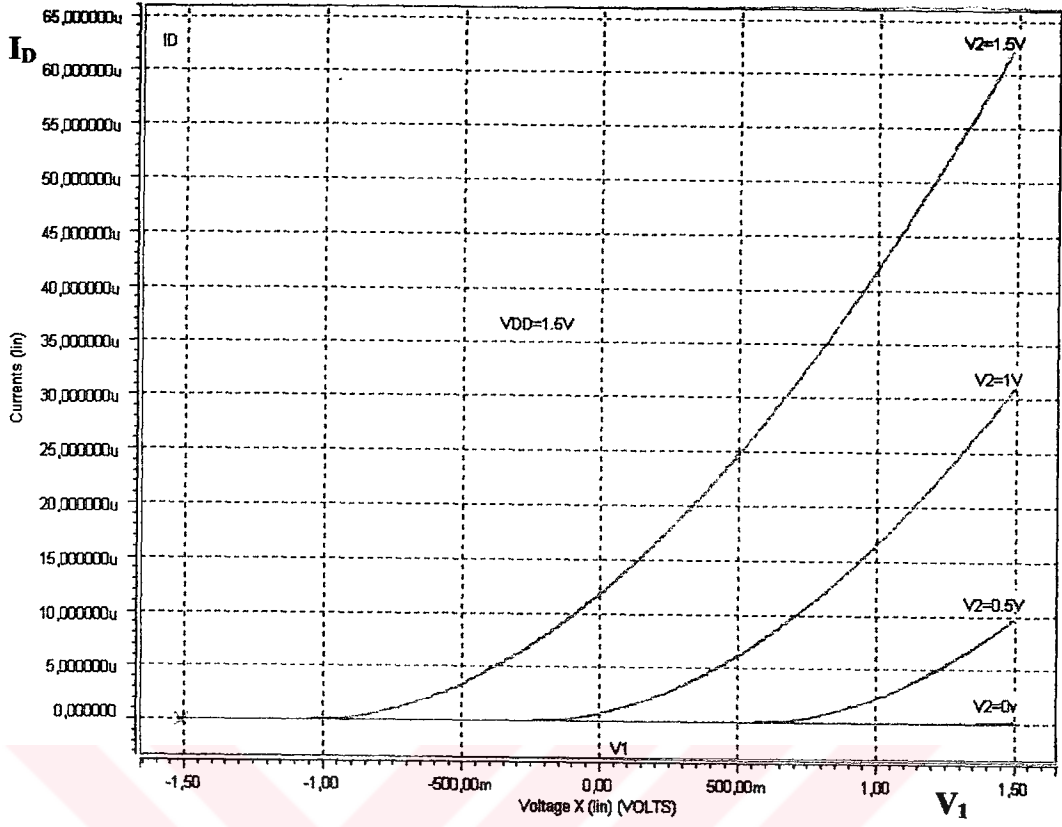
Şekil 3.9 Bir üç-uçlu ve bir dört-uçlu elemanın (FGMOS) karşılaştırılması (Ohmi, 1997)

3.7 FGMOS Transistorun Kullanım Alanları ve Avantajları

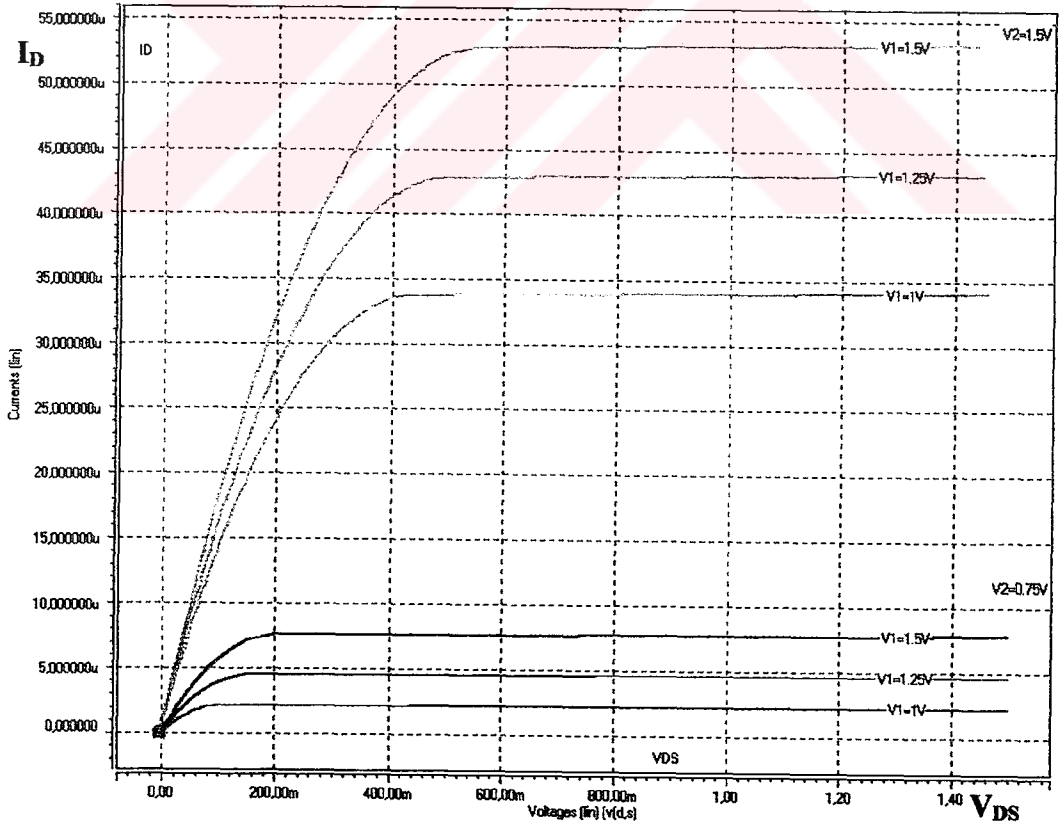
- Bu yapının gelecekte çok umut verici özellikleri gelişmiş bilgi işleme uygulamalarında kanıtlanmaktadır: çok geniş bir alana hitap etmektedir; binary dijital devreler, gerçek zamanlı konfigüre edilebilir lojik kapılar, kendi kendine öğrenen nöral ağlar, görüntü işleme, analog çarpıcılar, WTA entegrelerini gerçekleştirme gibi.
- CMOS'a göre daha hızlı ve daha az yer kaplıyor örneğin; bir tam toplayıcıyı neuMOS'larla gerçeklenirse CMOS'lara oranla %55 daha az alan, çarpıcılarda ise bu %65 daha az alan harcamasına kadar varıyor (Hirose, 1996).
- FGMOS'larla simetrik fonksiyonları gerçeklemek kolay oluyor.
- VLSI tasarımıda normal MOS'lara göre daha az güç harcaması var.
- Çoklu-değerli lojik yapılara uygunluk doğasında var
- Analog tasarım için özellikle düşük gerilim operasyonlarında yeni imkanlar sağlayan birkaç özelliği var
- Diğer MOS'lara göre yani tek girişli MOS'lara göre akımın kontrol edilebilmesi daha genişletilmiştir ve bu çok büyük bir esneklik sağlıyor (yani akımı kontrol altına almak daha özgürce).
- “Çok-düşük gerilimli” (Ultra-Low Voltage) devrelerin tasarımına olanak sağlıyor.

3.8 İki Girişli N-Kanallı FGMOS Transistorun Simülasyonla Elde Edilen Özeğrileri

Girişlerine V_1 ve V_2 işaretleri uygulanmış iki-girişli bir n-kanallı FGMOS transistorun simülasyonla elde edilen geçiş özeğrisi Şekil 3.10'da, çıkış özeğrisi Şekil 3.11'de verilmiştir. Tezde hep $\pm 1.5V$ çalışıldığından burada da besleme gerilimi $1.5V$ alınmıştır, eğer $5V$ alınırsa eğriler daha ayrıntılı bir şekilde görülebilir. Simülasyonda YİTAL $1.5\mu m$ CMOS parametreleri kullanılmış olup transistor geometrisi ile ilgili değerler; $W=6\mu m$, $L=1.5\mu m$, $C_1=25fF$, $C_2=50fF$ ve $C_T=85fF$ olarak alınmıştır.



Şekil 3.10 2-girişli n-kanallı FGMOS transistorun geçiş özeliğisi



Şekil 3.11 2-girişli n-kanallı FGMOS transistorun çıkış özeliğisi

4. FGMOS NÖRONUN TASARIMI

4.1 Farksal Gerilim Zayıflatıcı

Giriş bölümünde adı geçen, Vlassis ve Siskos'un 2001 yılında önerdikleri (Vlassis, 2001) FGMOS transistorlarla kurulmuş farksal gerilim zayıflatıcı devresi Şekil 4.1'de verilmiştir. Burada amaç, sadece dengeli giriş gerektiren herhangi bir devreyi zayıflatıcı kullanarak tek ucu sonlandırılmış giriş kabul eden benzerine dönüştürmek için bir yöntem geliştirmektir. Bu yöntem kullanılarak, gerilimin karesini alan, dört-bölgeli çarpıcı ve vektör toplayıcı gibi çok sayıda tek ucu sonlandırılmış hesaplama devreleri üretilmiştir (Vlassis, 2001).

Gerilim zayıflatıcılar; geri beslemeli kuvvetlendiriciler, geçiş iletkenlikleri gibi analog entegrelerin giriş katı olarak uygulama alanı bulan kullanışlı bir blok yapısıdır. Gerilim zayıflatıcılar aynı zamanda yüksek performanslı-sonlu kazançlı kuvvetlendiricilerde bir işlemsel kuvvetlendiricinin geri besleme yolundaki direncin yerine de kullanılabilir. Burada ise, iki tane birbirinin aynı olan iki-girişli FGMOS transistorun doyma bölgesinde çalıştırılmasıyla farksal gerilim zayıflatıcı yapısı oluşturulmuştur.

Dengeli girişli herhangi bir devreyi tek ucu sonlandırılmış benzerine çevirmek için giriş katında sadece iki tane zayıflatıcı kullanılır. Bu yöntem analog hesaplama devrelerine uygulanabilir. Analog hesaplama devreleri; analog işaret işleme, bulanık sistemler ve yapay sinir ağlarında bir çok uygulama alanı bulan çok kullanışlı doğrusal olmayan devrelerdir. Bu devrelerin çoğu düzgün çalışma için dengeli giriş gerektirirken ve küçük doğrusal giriş aralığına sahipken farksal zayıflatıcıların kullanılmasıyla giriş aralığı beslemeden beslemeye arttırılmış ve çok daha fazla doğrusal giriş aralığı sağlanmış olur.

Vlassis ve Siskos, doyma bölgesinde çalışan n-girişli bir FGMOS transistorun savak akımını ikincil etkilerin ihmal edilmesiyle şöyle tanımlamışlardır (Vlassis, 2001):

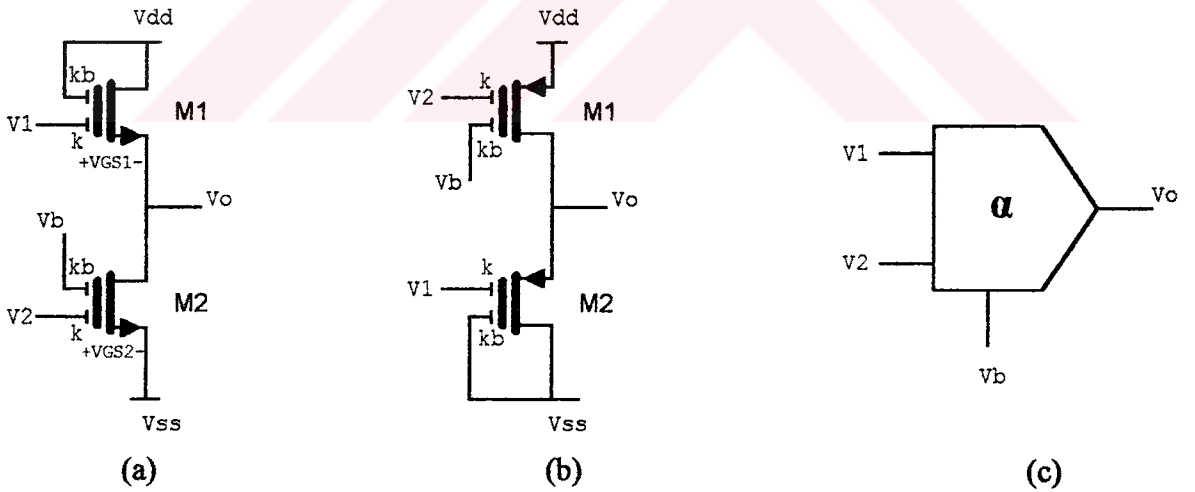
$$I_D = \beta [k_1(V_1 - V_s) + k_2(V_2 - V_s) + \dots + k_n(V_n - V_s) - V_T]^2 \quad (4.1)$$

burada $\beta = \frac{1}{2}K_n = \frac{1}{2} \frac{W}{L} \mu_n C_{ox}$ dur ve V_i ($i = 1, 2, \dots, n$) giriş gerilimleridir, k_i ($i = 1, 2, \dots, n$) giriş kapasitif etki oranıdır ve örtüşme kapasiteleri ihmal edilirse aşağıdaki gibi ifade edilebilir;

$$k_i = \frac{C_i}{\sum_{i=1}^n C_i + C_{GS}} \quad (4.2)$$

burada C_i i. girişle yüzen geçit arasındaki giriş kapasitesini temsil eder, C_{GS} ise yüzen geçit-kaynak kapasitesidir ve değeri; $\frac{2W}{3L}C_{ox}$ 'dur. Dikkat edilecek olursa 3. bölümde (3.14) denkleminde örtüşme kapasitelerinin ihmal edilmesiyle (3.17) denklemi elde edilmişti. Burada, $k_i = \frac{C_i}{C_T}$ olduğuna göre toplam kapasite değeri C_T 'nin (3.17) denklemindeki C_T 'ye eşit olduğu görülür.

Savak akımı incelenecek olursa; (3.16) denklemi örtüşme kapasitelerinin ihmal edilmesiyle (3.19) denklemindeki halini alır, Vlassis ve Siskos ise doymada çalışan bir FGMOS transistorun savak akımını örtüşme kapasitelerinin ihmal edilmesiyle (4.1) denklemindeki gibi tanımlamıştı. Sonuç olarak, örtüşme kapasitelerinin ihmal edilmesiyle (3.19) ve (4.1) denklemlerinin birbirine uyduğu görülür. Tezde ise, hesaplamaların çok karmaşık hal almaması ve anlaşılır olması için (4.1) denklemi temel akım denklemi olarak alınmıştır. Fakat simülasyon yapılırken gerçeğine çok yakın olan örtüşme kapasitelerinin ihmal edilmediği önceki bölümde tanıtılan (3.13) denkleminde dayalı makro model kullanılmıştır. Bilindiği üzere bu makro modelde savak akımını, örtüşme kapasitelerini de içeren (3.16) denklemi belirler.



Şekil 4.1 Doğrusal gerilim zayıflatıcı yapısı (a) n-kanallı FGMOS'lu yapı, (b) p-kanallı FGMOS'lu yapı, (c) zayıflatıcının sembolü

4.1.1 Farksal Gerilim Zayıflatıcının Yapısı

Şekil 4.1a'da gösterilen doğrusal gerilim zayıflatıcı, doyma bölgesinde çalışan iki tane birbirinin aynısı olan n-kanallı FGMOS transistorlar (M1 ve M2)' dan oluşur. Şekil 4.1b'de p-kanallı FGMOS transistorlardan oluşan zayıflatıcı yapısı gösterilmiştir, Şekil 4.1c'de ise

zayıflatıcının sembol gösterimi yer almaktadır. Her iki transistörün, k ve k_B kapasitif etki oranına sahip iki giriş geçidi vardır. Burada $k = C/(C_{GS} + C + C_B)$ ve $k_B = C_B/(C_{GS} + C + C_B)$ 'dir ve M1 ve M2 transistörleri aynı boyutlara sahip olduğundan C_{GS} kapasiteleri de eşittir.

V_1 ve V_2 giriş gerilimleri tek ucu sonlandırılmış işaretlerdir, V_B gerilimi çıkışta (V_o) oluşan offset gerilimini ayarlar. Eşit geçit-kaynak gerilimine ($V_{GS1} = V_{GS2}$) sahip olan M1 ve M2 transistörlerinden aynı akım akar. Böylece (4.1)'in kullanılmasıyla;

$$k_B(V_{DD} - V_o) + k(V_1 - V_o) = k_B(V_B - V_{SS}) + k(V_2 - V_{SS}) \quad (4.3)$$

elde edilir. Bu denklemin düzenlenmesiyle V_o çıkış gerilimi:

$$V_o = \alpha(V_1 - V_2) + \frac{kV_{SS} - k_B V_B}{k + k_B} \quad (\text{p-kanallı zayıflatıcı için } \Rightarrow V_o = \alpha(V_1 - V_2) + \frac{kV_{DD} - k_B V_B}{k + k_B}) \quad (4.4)$$

olur. Burada $\alpha = k/(k + k_B)$ zayıflatma çarpanıdır ve $V_{DD} = -V_{SS}$ 'dir. V_o , giriş gerilimlerinin farkıyla orantılı bir gerilimin ve offset geriliminin toplamıdır. Devre, yaklaşık zayıflatma çarpanını ve offset gerilimi ayarlayan V_B gerilimini seçmeye imkan tanır. Offset geriliminin sıfır olması için V_B :

$$V_B = \frac{k}{k_B} V_{SS}, \quad (\text{p-kanallı zayıflatıcı için } \Rightarrow V_B = \frac{k}{k_B} V_{DD}) \quad (4.5)$$

Yukarıda görüldüğü gibi aynı inceleme p-kanallı zayıflatıcı (Şekil 4.1b) için de geçerlidir. Böylece gövde etkisinden kurtulmak için, p-kuyu teknolojisinde n-kanallı FGMOS; n-kuyu teknolojisinde ise p-kanallı FGMOS seçebiliriz.

M1 ve M2' nin savak-kaynak gerilimleri $V_{DS1} = V_{DD} - V_o$ ve $V_{DS2} = V_o - V_{SS}$ 'dir. Zayıflatma çarpanı α 1'den çok küçük olduğunda, çıkış gerilimindeki değişim birkaç yüz mili voltlar aralığında sınırlı kalır. Böylece $V_{DS1,2} > V_{GS1,2} - V_T$ doyma şartı her zaman sağlanmış olur ve M1, M2 transistörleri her zaman için doyma bölgesinde çalışır.

M1, M2 transistörlerinin her ikisinin de giriş aralığının beslemeden beslemeye olabilmesi için her zaman iletimde olmaları gerekir, böylece geçit-kaynak gerilimleri eşik gerilimi V_T 'den büyük olmalıdır. (4.1)'in kullanılmasıyla aşağıdaki eşitsizlikler eş zamanlı olarak geçerli olmak zorundadır.

$$k_B(V_{DD} - V_o) + k(V_1 - V_o) \geq V_T \quad (4.6a)$$

$$k_B(V_B - V_{SS}) + k(V_2 - V_{SS}) \geq V_T \quad (4.6b)$$

Deminki eşitsizliklere göre, en kötü durum her iki girişin de negatif besleme gerilimine eşit olduğu durumdur. $V_1 = V_2 = V_{SS}$ ve $V_B = (k/k_B)V_{SS}$ (sıfır çıkış offseti için) alınıp (4.6a) ve (4.6b) eşitsizlikleri sadeleştirildiğinde şunu elde ederiz:

$$k_B \geq k - \frac{V_T}{V_{SS}} \quad (\text{n-kanallı zayıflatıcı için}) \quad (4.7a)$$

p-kanallı zayıflatıcı için (4.6) eşitsizliklerinde gerekli değişiklikler yapıp $V_1 = V_2 = V_{DD}$ ve $V_B = (k/k_B)V_{DD}$ için çözülürse şöyle olur:

$$k_B \geq k - \frac{V_T}{V_{DD}} \quad (\text{p-kanallı zayıflatıcı için}) \quad (4.7b)$$

(4.7)'den görüldüğü gibi, kapasitif etki oranları k ve k_B , eşik gerilimi ve negatif (p-kanallı için pozitif) besleme gerilimi tarafından belirlenir. Düşük eşik gerilimi teknolojisinin kullanılmasıyla k ve k_B 'nin değer aralığı ve böylece zayıflatma çarpanı α 'nın da değer aralığı arttırılabilir.

4.1.2 İkincil Etkiler ve Eleman veya Parametre Dengesizliklerinin İncelenmesi

Kanal kısılması, hareket kabiliyetinin değişimi ve eleman dengesizliklerinden dolayı oluşan ideal karesel akım yasasından sapmalar ihmal edilerek zayıflatıcının çalışması incelendi. Bu idealsizlikler zayıflatıcının hatalarının temel kaynağını temsil eder. M1 ve M2 FGMOS transistörlerin kaynak ve tabanları beraber aynı yere bağlandığından gövde etkisi incelenmeyecektir.

Kapasitif etki oranları k ve k_B sırasıyla 0.1 ve 0.8, her iki transistör için (W/L) oranı $20\mu\text{m}/5\mu\text{m}$, V_B gerilimi (4.5) bağıntısındaki gibi ve besleme gerilimleri $V_{DD} = -V_{SS} = 1.5 \text{ V}$ alınmıştır.

4.1.2.1 Kanal Kısılması (λ)

(4.4) ile verilen çıkış bağıntısı elde edilirken doymada çalışan FGMOS transistörlerin kanal kısılması göz ardı edilmiştir. Bunun hesaba katılması halinde, her iki transistörün savak akım ifadesi $(1 + \lambda V_{DS})$ çarpanını içerir, burada λ , kanal kısılması çarpanıdır. $V_1 = V_2$ alınırsa, λ 'yı içeren çıkış gerilimi;

$$V_{o,\lambda} = \frac{\{[2\alpha V_T + 2V_{DD}(k - k_B\alpha)]\lambda + 2k\}V_2 - k\alpha\lambda V_2^2}{(k_B + k)[\lambda(\alpha V_2 + 2V_{DD}) + 2]} \quad (4.8)$$

eşitliğiyle verilir. Tüm giriş koşulları altında $\lambda = 0.04$ alınır, $V_{o,\lambda}$, ideal V_o 'dan hemen hemen 4.3 mV farklı çıkmıştır. FGMOS transistörlerin kanal uzunlukları λ 'yı küçültmek üzere, büyük tutulmalıdır.

4.1.2.2 Hareket Yeteneğinin Düşey Elektrik Alan İle Değişimi (θ)

Zayıflatıcının doğruluğunu azaltan diğer bir faktör de hareket yeteneğinin değişmesidir. Bir transistörde kanal hareket yeteneğinin düşey elektrik alanla azalması

$$\mu = \frac{\mu_o}{1 + \theta |V_{GS} - V_T|} \quad (4.9)$$

ampirik bağıntısıyla verilir. Burada θ , hareket yeteneği değişim sabiti olan prosesle belirli bir sabit, μ_o ise gövdenin hareket yeteneğidir. (4.9) bağıntısı savak akımı ifadesinde yerine konulup $V_1 = V_2$ alınır, θ 'yı içeren çıkış gerilimi;

$$V_{o,\theta} = \frac{(k_b V_{DD} - V_T) \left(\sqrt{1 + \theta(k_b V_{DD} - V_T + k V_2)} - \sqrt{1 + \theta(k_b V_{DD} - V_T)} - k \sqrt{1 + \theta(k_b V_{DD} - V_T) V_2} \right)}{\sqrt{1 + \theta(k_b V_{DD} - V_T + k V_2)}} \quad (4.10)$$

Tüm giriş koşulları altında $\theta = 0.2/V$ alındığında, $V_{o,\theta}$, ideal V_o 'dan aşağı yukarı -3.6mV farklı çıkmıştır.

4.1.2.3 Girişlere Ait Kapasitif Etki Oranlarının Dengesizliği

FGMOS transistörler arasındaki kapasitif etki oranı dengesizliğinin zayıflatıcının performansına önemli bir etkisi vardır. M1 ve M2 transistörlerinin giriş-geçit dengesizlikleri sırasıyla,

$$k_1 = k_B + \Delta k_1, \quad k_2 = k + \Delta k_2 \quad (4.11)$$

$$k_3 = k + \Delta k_3, \quad k_4 = k_B + \Delta k_4 \quad (4.12)$$

olsun. Burada k ve k_B ortalama değerlerdir. Kapasitif etki oranı dengesizlikleri zayıflatma çarpanı α ve çıkış geriliminde değişiklikler meydana getirir. (4.11) ve (4.12)'nin (4.4)'de yerine konmasıyla kapasitif etki oranı dengesizliklerini içeren zayıflatma çarpanı $\alpha_{\Delta k}$:

$$\alpha_{\Delta k} = \alpha \left(1 + \frac{k_B}{k(k + k_B)} \Delta k_2 - \frac{1}{k + k_B} \Delta k_1 \right) \quad (4.13)$$

olur. Zayıflatma çarpanındaki değişim (4.13)'den görüldüğü gibi sadece M1 transistörüne ait

k_1 ve k_2 arasındaki kapasitif etki oranı uyumsuzluklarına bağlıdır. Giriş kapasitif etki oranındaki dengesizlik $\Delta k_1 = \Delta k_2 = \%1$ olduğunda α 'daki değişim $\%0.88$ olur.

Giriş kapasitif etki oranı dengesizliklerini içeren çıkış gerilimi $V_{o, \Delta k}$, şu hali alır:

$$V_{o, \Delta k} = V_o - \frac{k(\Delta k_1 + \Delta k_2 - \Delta k_3) - k_B \Delta k_3}{(k_B + k)^2} V_2 - \frac{k_B(\Delta k_1 - \Delta k_3 - \Delta k_4) + k \Delta k_4}{k_B(k_B + k)} V_{DD} \quad (4.14)$$

Tüm giriş koşulları altında $\Delta k_1 = \Delta k_2 = -\Delta k_3 = -\Delta k_4 = \%1$ alındığında, $V_{o, \Delta k}$, ideal V_o 'dan hemen hemen 3.13 mV farklı çıkmıştır.

4.1.2.4 Geçiş İletkenliği Dengesizliği

M1 ve M2 transistörleri arasındaki geçiş iletkenliği parametreleri uyumsuzluğu;

$$\beta_i = \beta + \Delta \beta_i, \quad (i = 1, 2) \quad (4.15)$$

olsun. Burada β ortalama değerdir. Geçiş iletkenliği parametrelerindeki dengesizlik sadece çıkış geriliminde bir değişim meydana getirir. (4.15)'in (4.4)'de yerine konmasıyla geçiş iletkenliği parametresindeki dengesizliği içeren çıkış gerilimi $V_{o, \Delta \beta}$:

$$V_{o, \Delta \beta} = V_o - \frac{k_B V_{DD} + k V_2 - V_T}{2\beta(k + k_B)} (\Delta \beta_1 - \Delta \beta_2) \quad (4.16)$$

olur. Tüm giriş koşulları altında β_1 ve β_2 arasındaki dengesizlik $\Delta \beta_1 = -\Delta \beta_2 = \%1$ alındığında $V_{o, \Delta \beta}$, ideal V_o 'dan yaklaşık 2.5 mV farklı çıkar.

4.1.2.5 Eşik Gerilimi Dengesizliği

M1 ve M2 transistörleri arasındaki eşik gerilimi dengesizliği;

$$V_{Ti} = V_T + \Delta V_{Ti}, \quad (i = 1, 2) \quad (4.17)$$

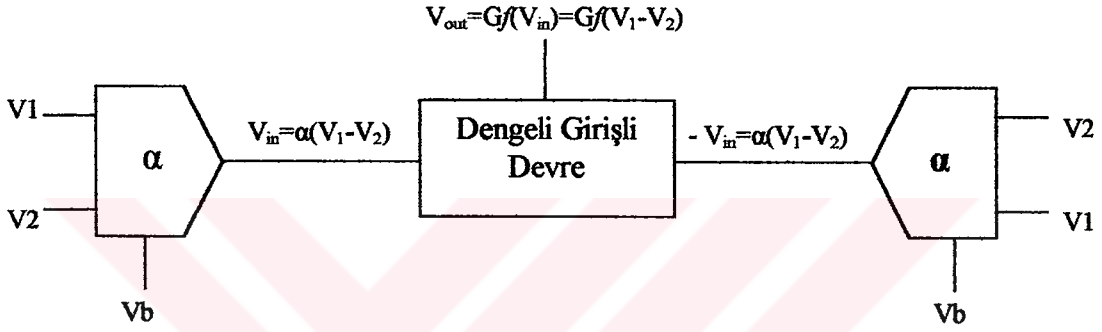
olsun. Burada V_T ortalama değerdir. Eşik gerilimindeki dengesizlik sadece çıkış geriliminde bir değişim meydana getirir. (4.17)'nin (4.4)'de yerine konmasıyla eşik gerilimindeki dengesizliği içeren çıkış gerilimi $V_{o, \Delta V_T}$:

$$V_{o, \Delta V_T} = V_o - \frac{\Delta V_{T1} - \Delta V_{T2}}{k + k_B} \quad (4.18)$$

olur. Tüm giriş koşulları altında V_{T1} ve V_{T2} arasındaki dengesizlik $\Delta V_{T1} = -\Delta V_{T2} = \%1$ alındığında $V_{o, \Delta V_T}$, ideal V_o 'dan yaklaşık 10 mV farklı çıkar.

4.2 Zayıflatıcıları Kullanarak Dengeli Girişli Bir Devrenin Tek Ucu Sonlandırılmış Haline Dönüştürülmesi

V-I çeviriciler, kare alıcılar ve çarpıcılar gibi hesaplama yapan devrelerin çoğunun bir kusuru, girişlerine eğer dengeli bir işaret uygulanırsa doğru çalışabilmeleridir. Dengeli giriş kullanılmadığı takdirde giriş gerilim aralığı kayba uğrayacaktır. Bu yöntem, giriş olarak sadece dengeli bir girişi kabul eden her hangi bir devreye uygulanarak, devrenin tek ucu sonlandırılmış girişle çalışabilmesini sağlar. Dolayısıyla, tek ucu sonlandırılmış işaretlerin gerekli olduğu yapay sinir ağları uygulamalarına çok uygundur.



Şekil 4.2 İki tane zayıflatıcı kullanarak dengeli girişin tek ucu sonlandırılmış girişe dönüştürülmesi

Şekil 4.2, bu yöntemin nasıl uygulandığını gösterir. Dengeli girişli devre, V_{in} 'in fonksiyonu olan bir çıkış gerilimi, $V_{out} = Gf(V_{in})$ 'i üretmek için V_{in} ve $-V_{in}$ girişlerini gerektirir. Burada G , kazanç; $f(V_{in})$ ise V_{in} 'in uygulanmasıyla devrenin gerçekleştirdiği fonksiyondur. Giriş katı olarak görev yapan iki doğrusal zayıflatıcı, tek ucu sonlandırılmış giriş işaretleri V_1 ve V_2 'yi farklarıyla orantılı bir gerilime çevirmek için kullanılmıştır. Böylece toplam devrenin çıkışı şu şekilde verilir:

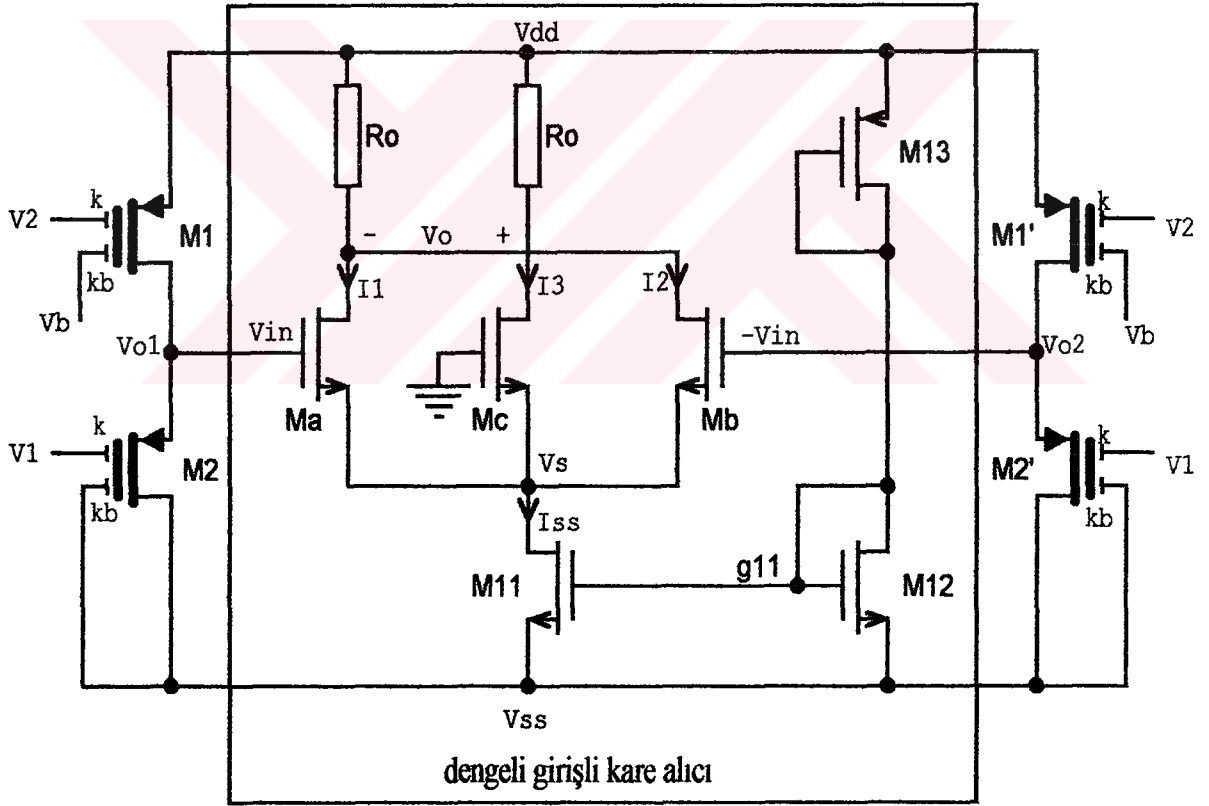
$$V_{out} = Gf(V_1 - V_2) \quad (4.19)$$

Bu yöntemin önemli bir avantajı giriş aralığının beslemeden beslemeye arttırılmış olmasıdır. Yöntemin dengeli girişe sahip gerilimin karesini alan bir devreye nasıl uygulandığı bir sonraki bölümde anlatılacaktır.

4.3 Kare Alıcı Devre

Şekil 4.3'te tek ucu sonlandırılmış kare alıcı devre şeması gösterilmiştir. Devre, giriş katlarında iki doğrusal zayıflatıcı bulunan dengeli girişli bir basit kare alıcıdan oluşur. Basit kare alıcı devre çoklu-kuyruk yöntemine dayanır (Giustolisi, 1997). Basit kare alıcı devre doyma bölgesinde çalışan M_a , M_b ve M_c transistorlarından oluşur. M_c transistorunun (W/L) oranı M_a ve M_c transistorlarının (W/L) oranının iki katıdır ve $\beta = \beta_a = \beta_b = \beta_c/2$ olur. M_c 'nin geçidi toprağa bağlanırken, M_a ve M_b 'nin geçidine dengeli giriş gerilimi V_{in} uygulanır. Ortak işaret giriş gerilimi sıfırdır. Dengeli girişler, V_{in} ve $-V_{in}$, iki zayıflatıcı tarafından üretilir. Doyma bölgesinde çalışan MOS transistorların iyi bilinen karesel-akım modelini ikincil etkileri ihmal ederek alırsak, kare alıcının çıkış gerilimi şu şekilde ifade edilir:

$$V_o = 2R_o \beta V_{in}^2 \quad (4.20)$$



Şekil 4.3 Zayıflatıcıların kullanıldığı tek ucu sonlandırılmış gerilimin karesini alan devre şeması

Kare alma fonksiyonu böylece gerçekleştirilmiş oldu. Giriş katlarında iki zayıflatıcının kullanılmasıyla farksal dengeli girişler elde edilir. M_1 ve M_2' transistorlarının giriş geçitlerinden birine V_2 gerilimi, M_2 ve M_1' transistorlarının giriş geçitlerinin birine de V_1

gerilimi uygulanmıştır. V_B gerilimi çıkış offset gerilimini sıfır yapmak için ayarlanır. Böylece, iki zayıflatıcının çıkış gerilimleri V_{o1} ve V_{o2} şu şekilde verilir:

$$V_{o1} = \alpha (V_1 - V_2) \quad (4.21a)$$

$$V_{o2} = -\alpha (V_1 - V_2) \quad (4.21b)$$

(4.20), (4.21a) ve (4.21b) denklemlerinin birleştirilmesiyle tüm devrenin çıkış gerilimi şu hali alır:

$$V_o = 2R_o \beta \alpha^2 (V_1 - V_2)^2 \quad (4.22)$$

Sonuç olarak, $V_1 - V_2$ gerilim farkının karesiyle doğru orantılı bir çıkış elde edilmiş olur. Burada şunu da söylemek gerekir ki; iki zayıflatıcı ve basit kare alıcı tarafından kaynaklanan bütün harmonikler ve çıkışa yansıyan hatalı dc bileşen, tüm devrenin farksal yapısından dolayı yok edilmiştir.

Basit kare alıcının bir transistörünün kesime gittiği varsayılırsa, $V_{GS} - V_T = 0$, maksimum giriş aralığı aşağıdaki eşitsizlikle belirlenir (Mb transistörünün kesime gittiği varsayılırsa aşağıdaki eşitsizlik kolayca çıkar):

$$(V_1 - V_2)^2 \leq \frac{I_{ss}}{6\alpha^2 \beta} \quad (4.23)$$

(4.23)'e göre I_{ss} akımının artırılmasıyla daha geniş bir giriş gerilim aralığına sahip olunur. Ancak, akımın artmasıyla R_o direncinde daha fazla gerilim düşeceğinden transistörlerin doğrusal bölgede çalışmasına neden olur. Bu özellikle, V_1 maksimum değerini $V_1 = V_{DD}$ alırken, V_2 'nin de minimum değerini $V_2 = V_{SS}$ aldığı ikinci bölgede ve V_1 minimum değerini $V_1 = V_{SS}$ alırken, V_2 'nin de maksimum değerini $V_2 = V_{DD}$ aldığı dördüncü bölgede uygulanır.

Akım aynasına ait M11 transistörünün M13 yük transistörü ile belirli referans akımını akıtabilmesi için doyma bölgesinde çalışması sağlanması, direnç rejimine girmesi engellenmelidir. M11 transistörünün doymada kalma şartını şu eşitsizlikle ifade edebiliriz:

$$V_S \geq V_{g11} - V_T \quad (4.24)$$

(4.24) eşitsizliğinde görülen, akım aynasına ait ortak geçit gerilimi V_{g11} 'in tasarım parametreleri cinsinden karşılığı, M12 transistörünün doyma bölgesindeki akımının referans akımı I_{ss} 'ye eşitlenip M11 ve M12'nin kaynak gerilimine V_{SS} yerleştirilerek V_{g11} 'in yalnız bırakılmasıyla bulunur:

$$V_{g11} = V_{ss} + \sqrt{\frac{2I_{ss}}{\mu_n C_{ox} \left(\frac{W}{L}\right)_{M12}}} \quad (4.25)$$

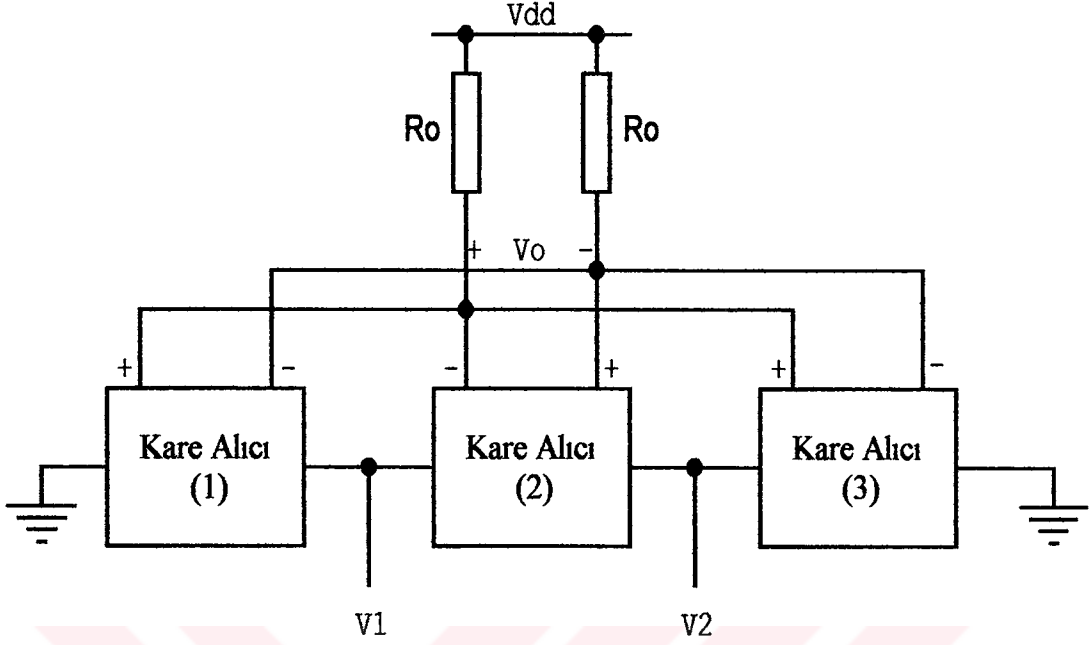
Besleme gerilimlerimizin ± 1.5 V olduğunu göz önünde bulundurarak (4.23) eşitsizliğindeki tüm değerleri yerine koyarsak, $I_{ss} \geq 170\mu A$ olması gerekir. Fakat çok büyük alınırsa, çıkış dirençlerinin üzerinde düşen gerilimin artmasıyla Ma, Mb, Mc ve M11 transistörlerinin doymadan çıkıp direnç bölgesine girmesine neden olur, ayrıca hem Ma, Mb ve Mc transistörlerinin hem de kullanılan akım aynasındaki transistörlerin boyutları büyüyeceğinden simülasyonu yapılan devrelerde I_{ss} değeri $170\mu A$ 'den belli bir güvenlik payı bırakılarak $200\mu A$ alınmıştır. Burada ayrıca şunu da söylemek gerekir ki; Vlassis ve Siskos, Şekil 4.3'deki devrede V_s ve V_{ss} uçları arasına I_{ss} akımını akıtacak bir akım kaynağı bağlamışlardır, tezde ise bu akım kaynağını akım aynası kullanarak gerçekleştirdiğimiz için Şekil 4.3'de akım aynası yapısının kullanıldığı devre şeması gösterilmiştir. Ayrıca bu devredeki M13 transistörü, sürekli doymada çalışmakta olup referans akımını sağlayan yük direnci olarak kullanılmıştır. Akım aynasının ve bu yük direncinin parametreleri belirlenmesi: M11 ve M12 transistörlerinin (W/L) oranının büyütülmesi M11 transistörünün doymadan çıkmasını zorlaştıracığından, M13'ün (W/L) oranı; M11 ve M12'nin (W/L) oranı birbirine eşit alınarak belirlendikten sonra, I_{ss} referans akımını akıtacak şekilde ayarlanır. Ayrıca M13 transistörü, daha kararlı bir yapı sağladığı için PMOS transistör seçilmiştir.

Düzgün çalışma için maksimum yük direncine bağımlılık şu şekilde ifade edilir (Mb transistörünün kesimde olması ve Ma transistörünün doğrusal bölgede çalışması varsayımıyla aşağıdaki ifadeye ulaşılır):

$$R_{o\max} = \frac{V_{DD}(1-2\alpha) + V_T}{\frac{I_{ss}}{2} + 4\beta\alpha^2 V_{DD}^2} \quad (4.26)$$

Giriş aralığı, $R_{o\max}$ 'dan daha küçük yük direnç değerleri için sınırlı kalır. (4.26) bağıntısında yukarıda aldığımız I_{ss} değeri de dahil olmak üzere değerler yerine konulursa, $R_{o\max} \approx 8 K\Omega$ gibi bir değer çıkar. Bu değer belirlenirken dikkat edilmelidir ki; çok büyük seçilirse dirençlerin üzerinde fazla gerilim düşümü meydana geleceğinden transistörlerin doyma bölgesinden çıkmasına sebep olur, ayrıca doğrudan çıkış gerilimini etkilediği için çok küçük seçilirse de çıkış gerilimi küçülecektir, bunun için ortalama bir değer olarak $5 K\Omega$ seçilmiştir.

4.4 FGMOS Transistorlarla Kurulmuş Dört Bölge Analog Çarpma Devresi



Şekil 4.4 Tek ucu sonlandırılmış girişli dört-bölgeli çarpıcı

Şekil 4.4'te gösterildiği gibi bir dört-bölgeli çarpıcı, önceki bölümde genişçe anlatılan kare alıcı devre kullanılarak kolayca gerçekleştirilebilir. Üç tane kare alıcı devre kullanılır: birinci kare alıcının giriş gerilimleri V_1 ve 0'dır, ikincinin giriş gerilimleri V_1 ve V_2 ve son olarak üçüncü kare alıcının giriş gerilimleri 0 ve V_2 'dir. bu üç tane kare alıcı kullanılan devrenin çıkışı kabaca şu hali alır: $(V_1 - 0)^2 - (V_1 - V_2)^2 + (0 - V_2)^2$, düzenlenirse; $V_1^2 + V_2^2 - (V_1 - V_2)^2$ ifadesini elde edilir. Bu da, bu çarpıcının gerçekleşmesinin, $x^2 + y^2 - (x - y)^2 = 2xy$ özdeşliğine dayandığını gösterir. Bu özdeşliğin (4.22) yerine konulmasıyla çarpıcının çıkışı şu şekilde verilebilir:

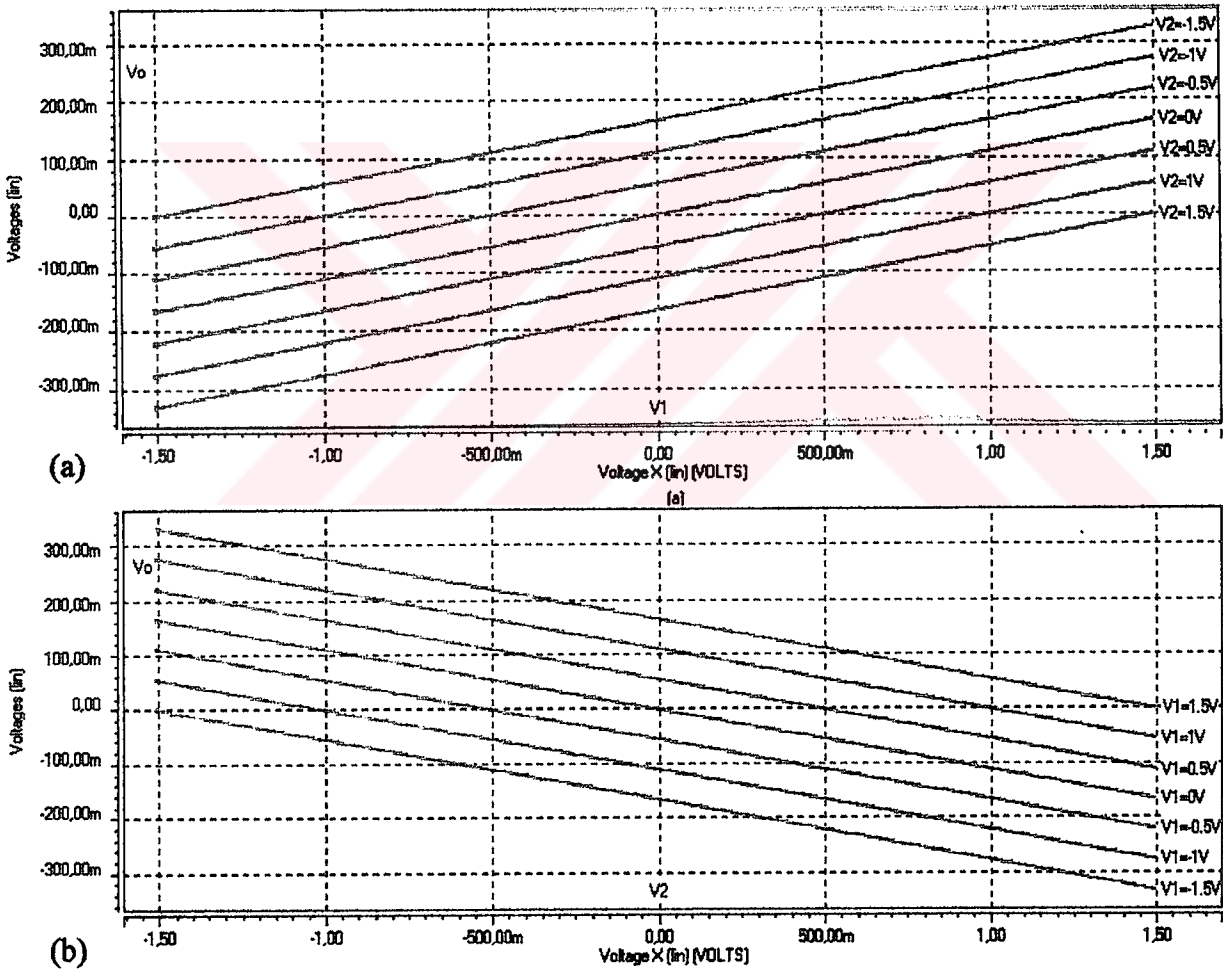
$$V_o = 4R_o\beta\alpha^2 V_1 V_2 \quad (4.27)$$

V_1 ve V_2 gerilimlerinin tek ucu sonlandırılmış işaretler olması ve beslemeden beslemeye dinamik aralığa sahip olabilmesi bu çarpıcının avantajlarındandır.

4.5 Zayıflatıcı, Kare Alıcı ve Çarpıcıya Ait Simülasyon Sonuçları

N-kuyu $1.5 \mu\text{m}$ YİTAL parametreleri kullanıldığından gövde etkisini önlemek için p-kanallı FGMOS transistorlu zayıflatıcılar seçilmiştir. Bütün devreler için besleme gerilimi $\pm 1.5 \text{ V}$ tur. Sonuçlar HSPICE simülasyon programı kullanılarak elde edilmiştir.

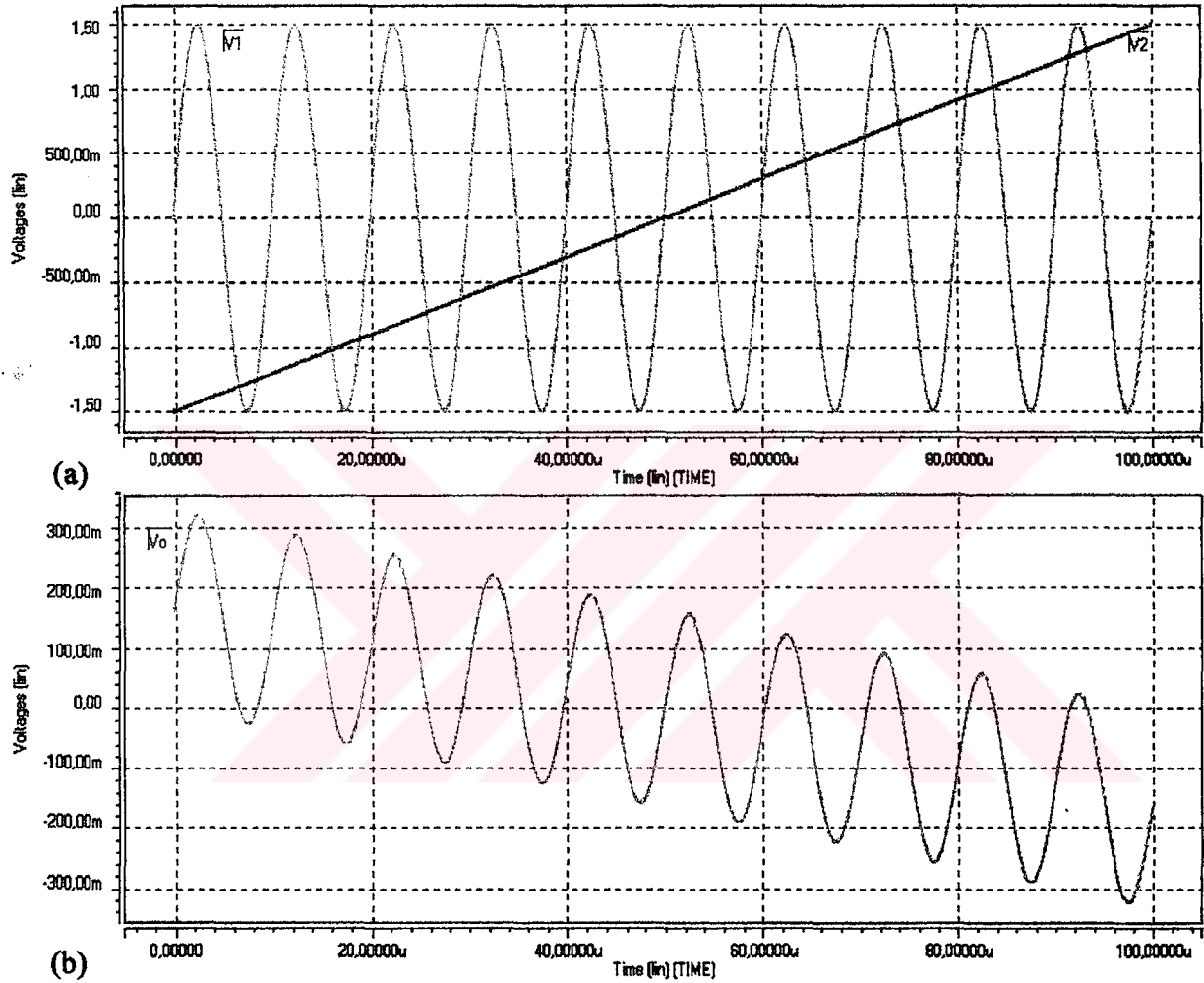
Doğrusal zayıflatıcının zayıflatma çarpanı $\alpha = 1/9$ ' dur, beslemeden beslemeye giriş gerilimine sahiptir ve çıkış offseti sıfırdır. $V_B = 0.1875 \text{ V}$ iken, giriş kapasitif etki oranları $k = 0.1$ ve $k_B = 0.8$ ve modelde belirtilen $k_S = C_{FS}/C_T = 0.098$, $k_{SUB} = C_{FSUB}/C_T = 0.001$ ve $k_D = C_{FD}/C_T = 0.001$ alınmıştır. $(W/L)_{M1,2} = 20/5$ ' dir. Böylece giriş kapasitelerinin değeri de, $C = 93 \text{ fF}$, $C_B = 744 \text{ fF}$ ve toplam kapasite $C_T = 930 \text{ fF}$ olur.



Şekil 4.5 Gerilim zayıflatıcının DC geçiş karakteristiği: (a) $V_o - V_1$ grafiği, V_2 parametre. (b) $V_o - V_2$ grafiği, V_1 parametre.

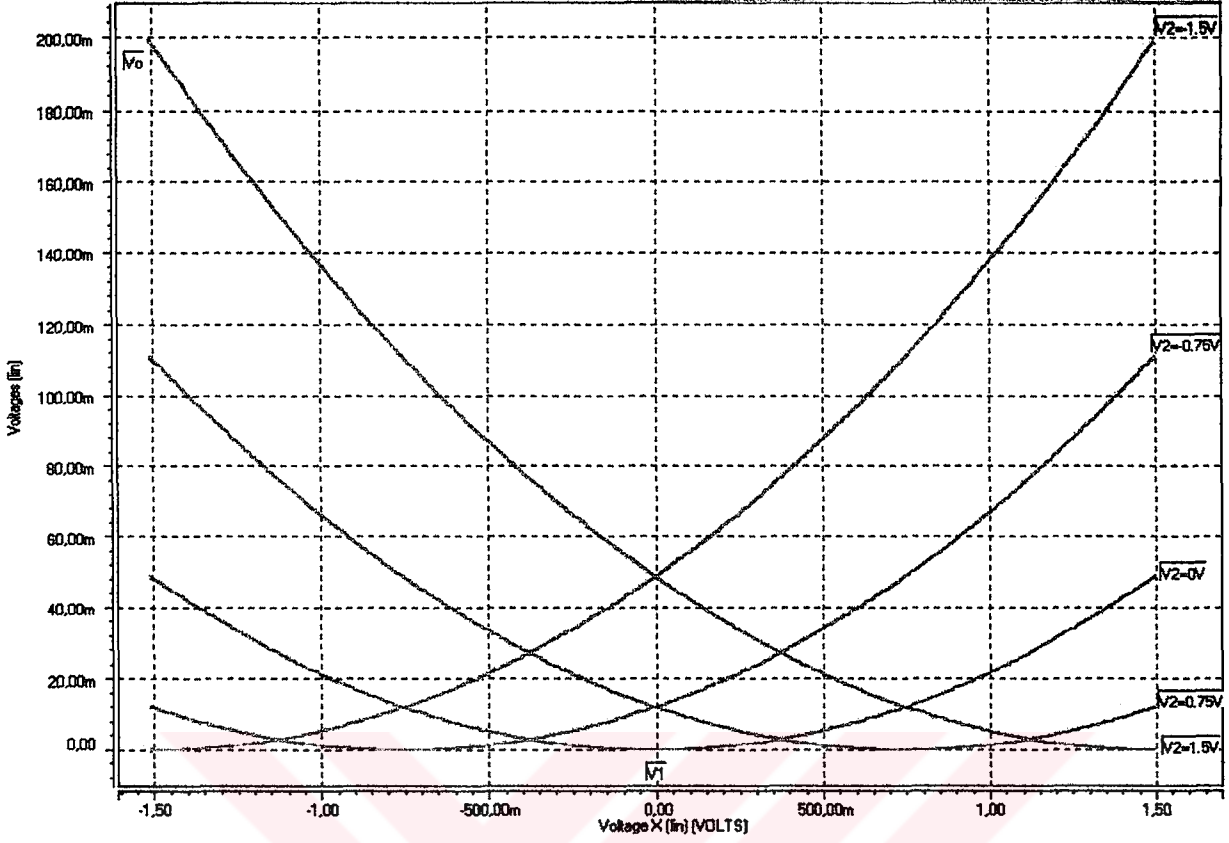
Şekil 4.5a'daki eğriler, çıkış gerilimi V_o 'nın V_1 ile değişimini göstermektedir, burada V_2 , -1.5 V 'tan 1.5 V 'a kadar 0.5 V adımlarla değişen değerler alan bir parametredir. Şekil 4.5b'de V_1 ile V_2 yer değiştirmiştir. Zayıflatıcının çıkış gerilimi -330 mV ile 330 mV arası

değişmektedir. Devre beslemeden beslemeye giriş aralığı için %0.057 gibi çok küçük lineersizliğe sahiptir. Zayıflatıcının transient cevabı Şekil 4.6'da gösterilmiştir. Şekil 4.6a'da iki giriş işaretleri gösterilmektedir, V_1 , genliği 1.5 V, frekansı 100kHz olan sinüzoidal bir işaret, V_2 ise -1.5 V 1.5 V arası değişen ve 10kHz'lik frekansa sahip bir rampa fonksiyonudur. Çıkış gerilimi V_o Şekil 4.6b'de gösterilmiştir.

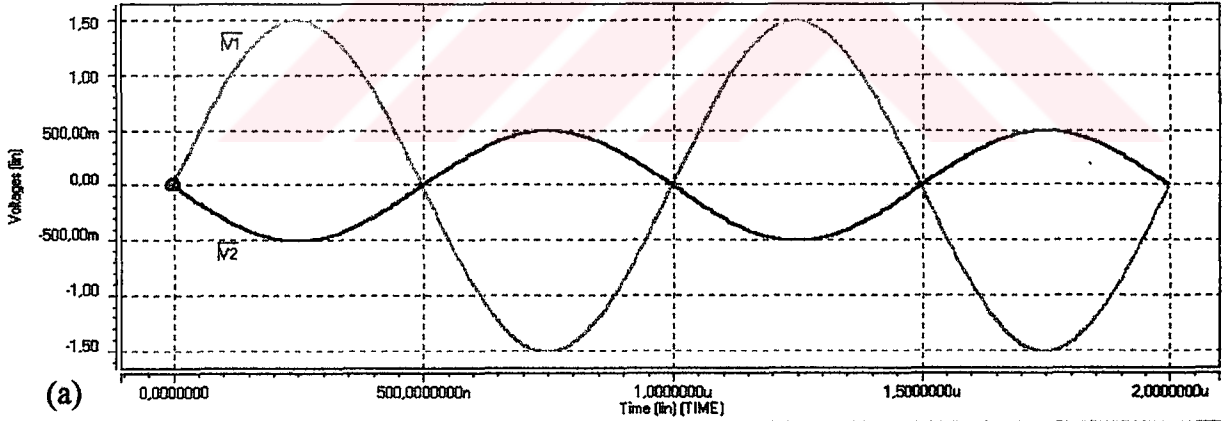


Şekil 4.6 Zayıflatıcının transient cevabı (a) giriş işaretleri (b) çıkış gerilimi

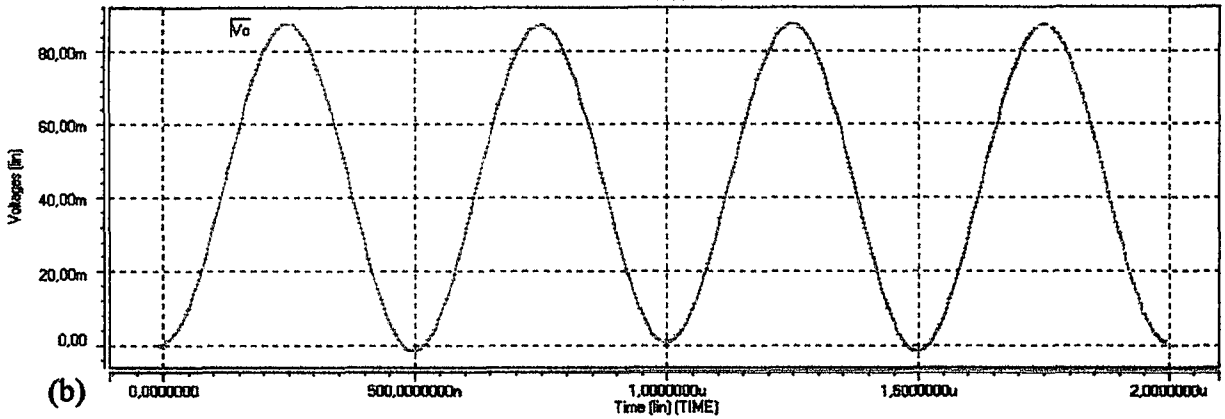
İki zayıflatıcıya sahip kare alıcının akım kaynağı ve yük direnci değerleri önceki bölümde ayrıntılı bir şekilde değinildiği gibi sırasıyla 200 μ A ve 5 K Ω olarak belirlenmiştir. M_a , M_b ve M_c transistorlarına ait $(W/L)_{M_a, b} = 33/5$ ve $(W/L)_{M_c} = 66/5$ değerleri alınmıştır. Farksal kare alıcının DC geçiş karakteristiği Şekil 4.7'de çizdirilmiştir, burada giriş gerilimi V_1 beslemeden beslemeye geçirirken parametre olan V_2 gerilimi -1.5 V'dan 1.5V'a kadar 0.75V'luk adımlarla değişmektedir. Şekil 4.8 ise kare alıcının transient cevabını gösterir. İki giriş işaretleri Şekil 4.8a'da gösterildiği gibi 1 MHz frekansına sahip sinüzoidal işaretlerdir ve V_1 'in genliği 1.5 V, V_2 'ninki ise 0.5 V' tur. Çıkış gerilimi V_o ise Şekil 4.8b'de gösterilmiştir.



Şekil 4.7 Kare alıcının DC geçiş karakteristiği



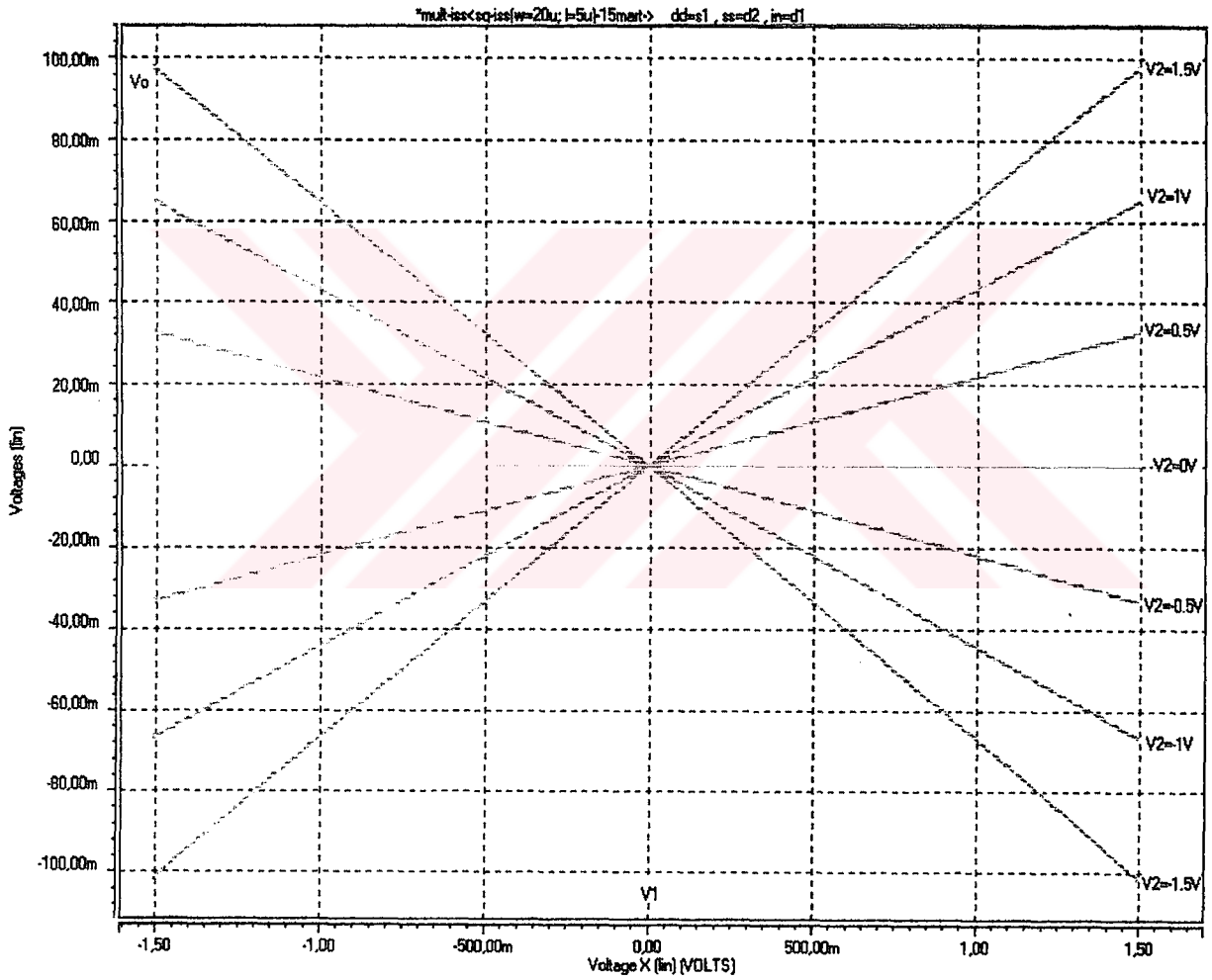
(a)



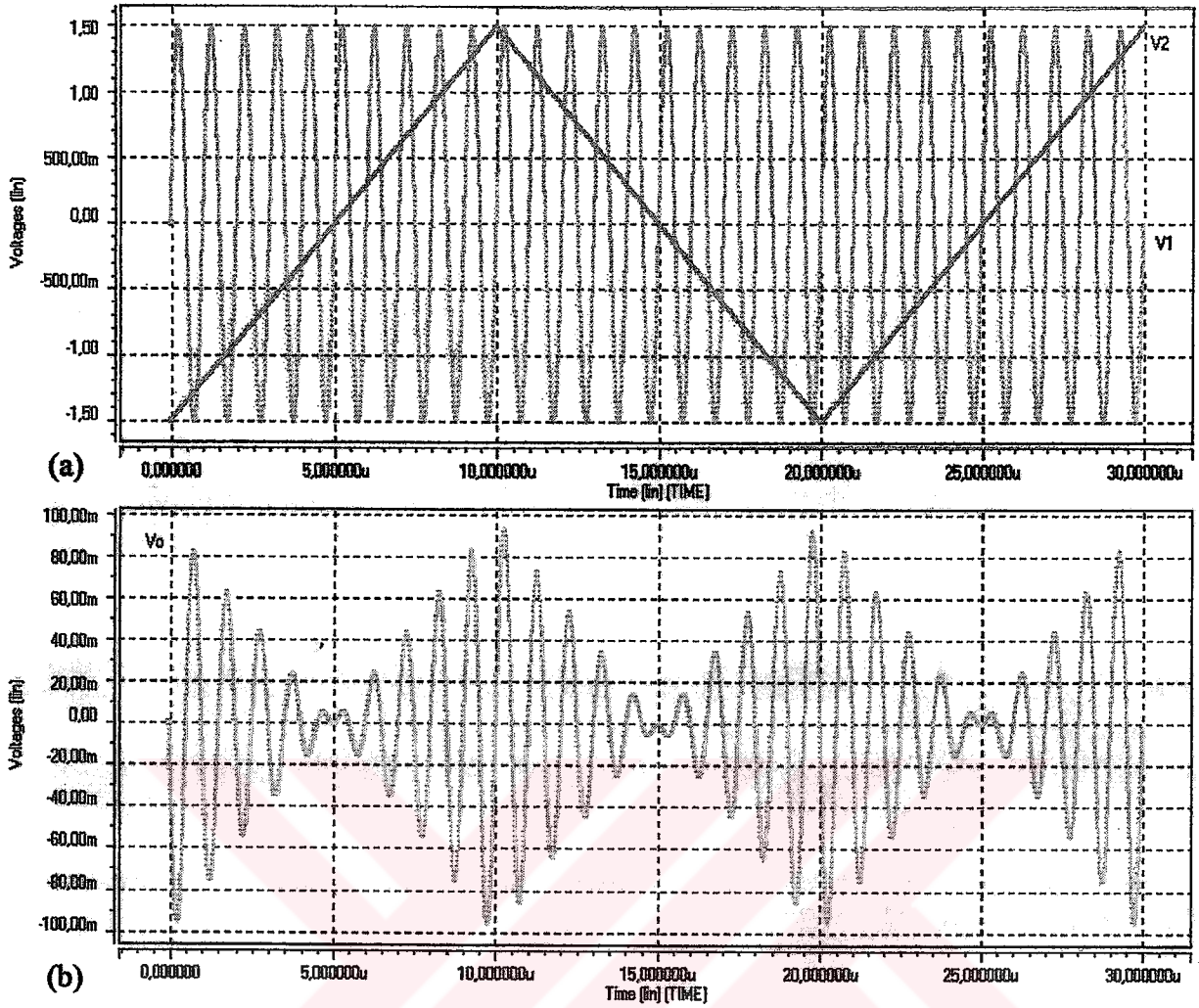
(b)

Şekil 4.8 Kare alıcının transient cevabı (a) giriş işaretleri (b) çıkış gerilimi

Şekil 4.9'da tek ucu sonlandırılmış girişli çarpıcının DC geçiş karakteristiği verilmiştir. Beslemeden beslemeye giriş gerilim aralığı için ± 100 mV çıkış salınım aralığı elde edilmiştir. Parametre olan V_2 gerilimi ise, -1.5 V'dan 1.5 V'a kadar 0.5 V'luk adımlarla değişmektedir. Şekil 4.10'da ise çarpıcının transient cevabı verilmiştir. V1 girişi olarak tam salınlımlı 1 MHz'lik sinüzoidal işaret, V2 girişi olarak tam salınlımlı 1 kHz frekanslı üçgen dalga (Şekil 4.10a'da girişler gösterilmiştir) uygulanması ile aynı zamanda genlik modülatörü olarak kullanılması gösterilmiştir (Şekil 4.10b).



Şekil 4.9 Dört-bölgeli çarpıcının DC geçiş karakteristiği



Şekil 4.10 Dört-bölgeli çarpıcının transient cevabı (genlik modülatörü olarak kullanılması)
(a) giriş işaretleri (b) çıkış gerilimi

4.6 En Temel Nöron Yapısı

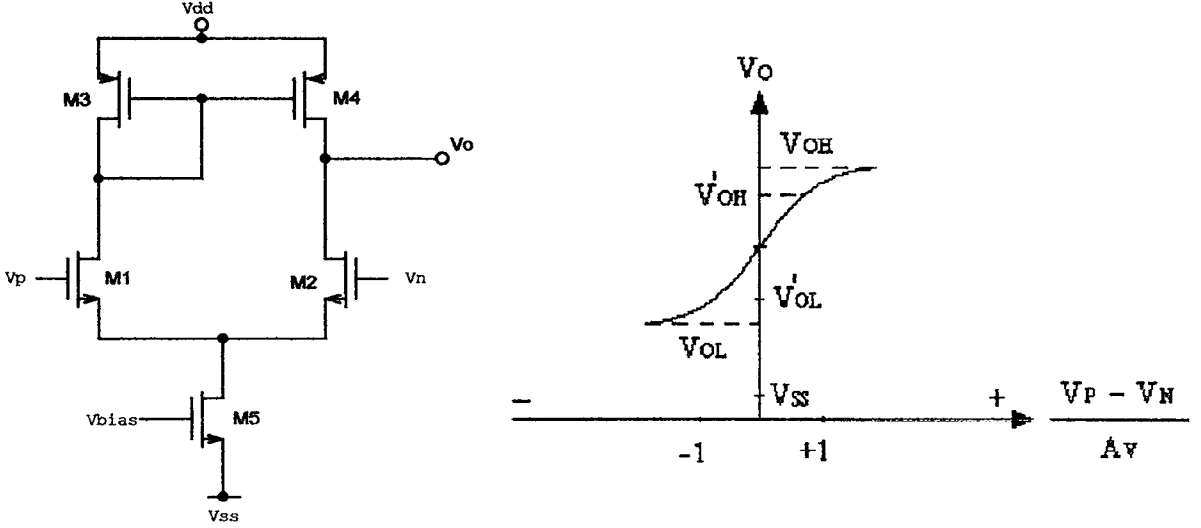
Nöronun yapısı 2. Bölümde ayrıntılı bir şekilde ele alınmıştır. Bundan dolayı bu bölümde nöron yapısının elektronik devre olarak nasıl gerçekleştirildiğine değinilecektir. Bilindiği gibi temel nöron mantığı, girişe gelen işaretlerin o girişle ilgili belli ağırlık değerleriyle çarpılması -yani ağırlıklı çarpım- daha sonra bu çarpım sonuçlarının toplanıp -yani çarpımların toplamı- bir aktivasyon fonksiyonundan geçirilip, bu fonksiyonun karakteristiğine göre sonuca varılması şeklinde kısaca özetlenebilir.

İlk olarak en basit ama temel yapı olan bir girişli bir nöron gerçekleştirilmiştir. Bu bir girişli yapıda dikkat edilecek olursa çarpımların toplamını almayı gerektirecek bir devreye ihtiyaç yoktur, bundan dolayı nöronun girişine gelen işaretin o girişle ilgili ağırlık-değeriyle (burada

zaten tek bir giriş olduğu için bir tane ağırlık değeri vardır) çarpılması ve aktivasyon fonksiyonundan geçirilmesi sonuca ulaşmak için yeterlidir. İşte bunu yaparken çarpım işlemini şu ana kadar anlatılan çarpıcı devreyle (çarpıcının girişlerine nörona gelen giriş işareti ve ağırlık değeri gerilim olarak uygulanır) gerçekleştirilirken, aktivasyon fonksiyonundan geçirilmesi işlemi farksal karşılaştırmacı kullanılarak gerçekleştirilmiştir. Bizim çarpıcımızın çıkışı gerilim farkı şeklinde olduğundan farksal karşılaştırmacı kullanmak uygun olmuştur.

4.6.1 Farksal Karşılaştırmacı Yapısı

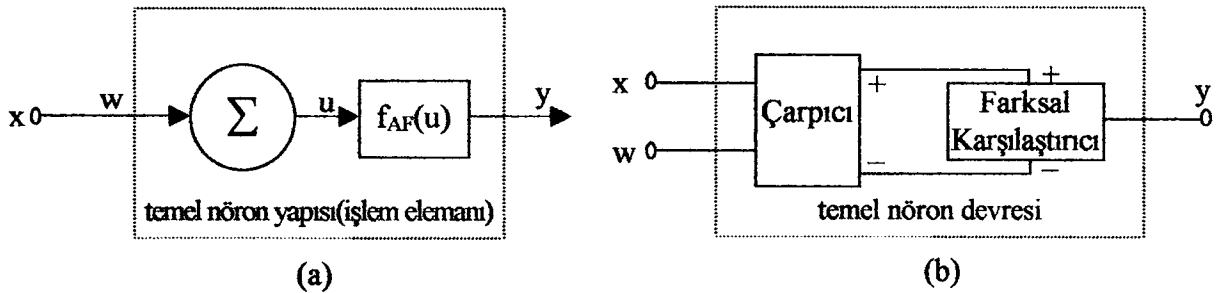
Şekil 4.11'de çok bilinen bir farksal karşılaştırmacının yapısı ve DC geçiş eğrisi verilmiştir. Farksal karşılaştırmacının çalışma şekline göre devrenin pozitif ile negatif giriş uçlarına verilen gerilim değerleri karşılaştırılır ve idealde pozitif uç negatiften büyükse pozitif bir gerilim, eşitse 0 çıkışı (bias ucuna verilen girişe göre değiştirilebilir), küçük ise negatif bir çıkış gerilimi üretmektedir. Yani şekilden de görüldüğü gibi devre girişine gelen iki işaretin farkını alarak, bu farkla orantılı olarak çıkışta bir gerilim üretmektedir. Çıkış devrenin kazancına göre genelde beslemeden beslemeye aralığında değişir. Şekil 4.11'de görülen çıkış eğrisinin eğimini, özellikle M1 ve M2 transistorlarının (W/L) oranlarını değiştirerek istediğimiz gibi ayarlayabiliriz. Devrenin bu giriş-çıkış ilişkisi bizi aktivasyon fonksiyonundaki giriş-çıkış ilişkisine götürür ve bu mantıkla transistorların (W/L) oranlarının değiştirilmesiyle fonksiyon, aktivasyon fonksiyonları olarak bilinen (2. bölümde anlatılmıştır) ve bizim de çalışmalarımızda kullandığımız simetrik basamak fonksiyonunun, doymalı rampa fonksiyonunun ve tanjant-hiperbolik fonksiyonunun kaba bir yaklaşımla gerçekleşmesini sağlar. Yapılan simülasyon sonuçlarında görülmüştür ki; transistorların kanal uzunlukları olan L ile çıkış eğrisinin eğimi arasında doğru orantı vardır. Bu yapıda M1 ve M2 transistorları boyutları kendi aralarında eşit, M3 ve M4 transistorlarının boyutları da kendi aralarında eşit alınmıştır. Böylece basit bir mantıkla, nöronları elektronik devre olarak gerçeklemede çok zor olan aktivasyon fonksiyonunun gerçekleşmesi işlemini basite indirgemiş oluruz.



Şekil 4.11 Farksal karşılaştırıcı devresi ve DC geçiş eğrisi

4.6.2 Temel Nöronun (Bir Girişli Nöron) Gerçeklenmesi

Şekil 4.12a'da temel nöron yapısı verilmiştir, buna göre amacımız bu temel yapıyı elektronik olarak gerçeklemektir, bu gerçekleştirilmiş yapı da Şekil 4.12b'de görülmektedir. Artık nöronun elektronik devre gerçeklemede bize gereken iki yapı taşını biliyoruz, bunlar; yukarıda anlatıldığı gibi bir çarpıcı bir de farksal karşılaştırıcıdır. Şekil 4.12'de temel nöron yapısı ve devre gerçeklemesi yan yana verilmiştir, aşıkardır ki sağ taraf sol tarafın aynı yapısına sahip bir devre gerçeklemesidir. Daha anlaşılır olması için şekil üzerinde kısaca açıklanacak olursa; sol tarafta x girişi ağırlığı olan w ile çarpılır ki, bu kısım sağ tarafta girişleri x ve w olan bir çarpıcı olarak gösterilmiştir, yine sol tarafta bu çarpımın çıkışı bir aktivasyon fonksiyonundan geçirilerek çıkış elde edilmiş olur, bu işlem ise sağ tarafta çarpımın bir farksal karşılaştırıcı girişlerine uygulanarak sonuç elde edilmesine karşılık gelir.

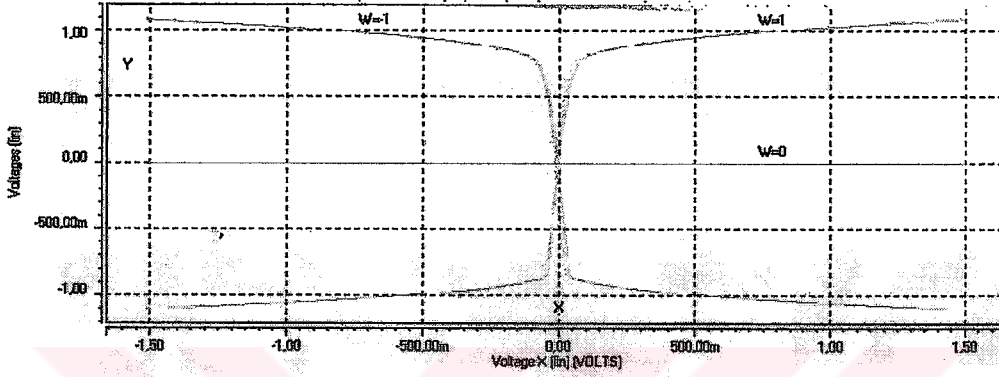


Şekil 4.12 (a) Temel nöron yapısı (b) Temel nöron devresinin blok şeması

Burada şunu belirtmek gerekir, bu yapı zaten bir nöron yapısı değildir, biyolojik nöron yapısı hatırlanacak olursa hücreye gelen birden çok uyarılara karşın bu uyarıların toplanmasıyla bu

toplamanın belli bir eşiğin altında mı üstünde mi olduğuna karar verilip işlem yapılıyordu, yani nöronun yapısında olan girişlerin ağırlıklı çarpımlarının toplamı yoktur, sonuç olarak burada paralel işlem mantığı yoktur. Fakat dikkat edilecek olunursa burada bu yapıdan “temel nöron yapısı” olarak bahsedilmiştir, yani gerçek nöron yapısını oluşturmada bir ön adım denilebilir

Şekil 4.13'te, Şekil 4.12b'de verilen “temel nöron devresi” nin simülasyon sonucu görülmektedir. Devreye x girişi olarak -1.5 V ile $+1.5\text{ V}$ arası değişen gerilim verilir, w ağırlık değerleri -1 , 0 ve 1 V alınarak DC analiz yapılmıştır.



Şekil 4.13 Temel nöron devresinin değişen ağırlık değerlerine göre DC analizi

4.7 İki Girişli Nöron Yapısı

Bu bölümde, geçen bölümdeki bir ön adım olan temel nöron yapısı yerine artık “gerçek nöron yapısı” elektronik devre olarak gerçekleştirilecektir. Literatürde rastlanan nöron tasarımlarında genelde (Bayraktaroğlu, 1998) akım çıkışlı bir çarpıcı, sonra bu çarpımların toplama işlemi için bir işlemsel kuvvetlendirici (bu yapı aynı zamanda I-V çevirimi yapar), daha sonra aktivasyon fonksiyonunu gerçekleştiren örneğin sigmoid fonksiyon üreticiler kullanılıyor. Fakat bu çalışmada, bütün bunların yerine toplama işlemini ve aktivasyon fonksiyonundan geçirme işlemini (çarpıcının çıkışı gerilim olduğundan I-V çevirimine gerek kalmıyor), ikisini bir arada yapabilen bir yapı olan FGMOS transistörlerle gerçekleştirilmiş bir farksal karşılaştırıcı kullanılarak bu sorun da daha az eleman ve yapı kullanılarak halledilmiştir.

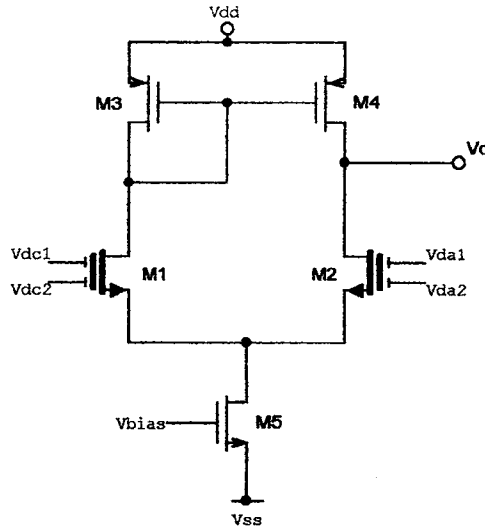
4.7.1 FGMOS Farksal Karşılaştırıcı

Temel nöron yapısını gerçeklerken bir toplama işlemi olmadığından sadece farksal karşılaştırıcı kullanmak yeterli olmuştu, fakat burada ağırlıklı çarpımların toplanması işlemi gerektiği için bir gerilim toplayıcıya ihtiyaç duyulur bu da ek bir devre gerektirir, işte burada

'yüzen-geçit' li MOS transistorun yani FGMOS'un bu gerilim toplama işlemini çok basit bir şekilde yapabileceği düşünülmüştür. Bilindiği gibi FGMOS'un yapısında zaten girişlere gelen gerilimleri belli ağırlık değerleriyle çarparak toplama vardır. Basit bir farksal karşılaştırıcının giriş transistorlarının yerine FGMOS transistorlar takılarak, FGMOS farksal karşılaştırıcı Şekil 4.14'te gösterildiği gibi gerçekleştirilmiş olur. Bu şekilde iki girişli FGMOS transistorlar gösterilmiştir (iki-girişli nöron gerçekleştirileceği için) fakat bilindiği gibi ağırlıkların (giriş kapasitif etki oranları) değiştirilmesi şartıyla FGMOS transistorların giriş sayısı 2'den n'ye kadar olabilmektedir. Bunu yaparken önemli olan önceki bölümlerde temel yapı olarak alınan zayıflatıcıda kullanılan FGMOS yapısını incelemek gerekir, bu yapıda giriş kapasitif etki oranları (k ve k_B) ve MOS transistorun kendi yapısından kaynaklanan kapasitif etki oranları (k_S , k_{SUB} ve k_D) toplamının 1'e eşit olduğu önceki bölümlerden bilinmektedir. Zayıflatıcı devresinde bu giriş kapasitif etki oranları toplamı, $k + k_B = 0.9$ olarak alınmıştır buna göre, diğer kapasitif etki oranları toplamı ; $1 - 0.9 = 0.1$ çıkar. Burada önemli olan 0.1'lik etki oranı sabit kalacak şekilde –bu durumda zaten girişlere toplam 0.9'luk bir etki kalması söz konusu- giriş oranlarını kendi aralarında ayarlamaktır. Burada ise FGMOS farksal karşılaştırıcının girişleri aynı etkiye sahip olması bekleneneğinden tüm girişlerin kapasitif etki oranları eşit alınır, bu durum şöyle ifade edilebilir:

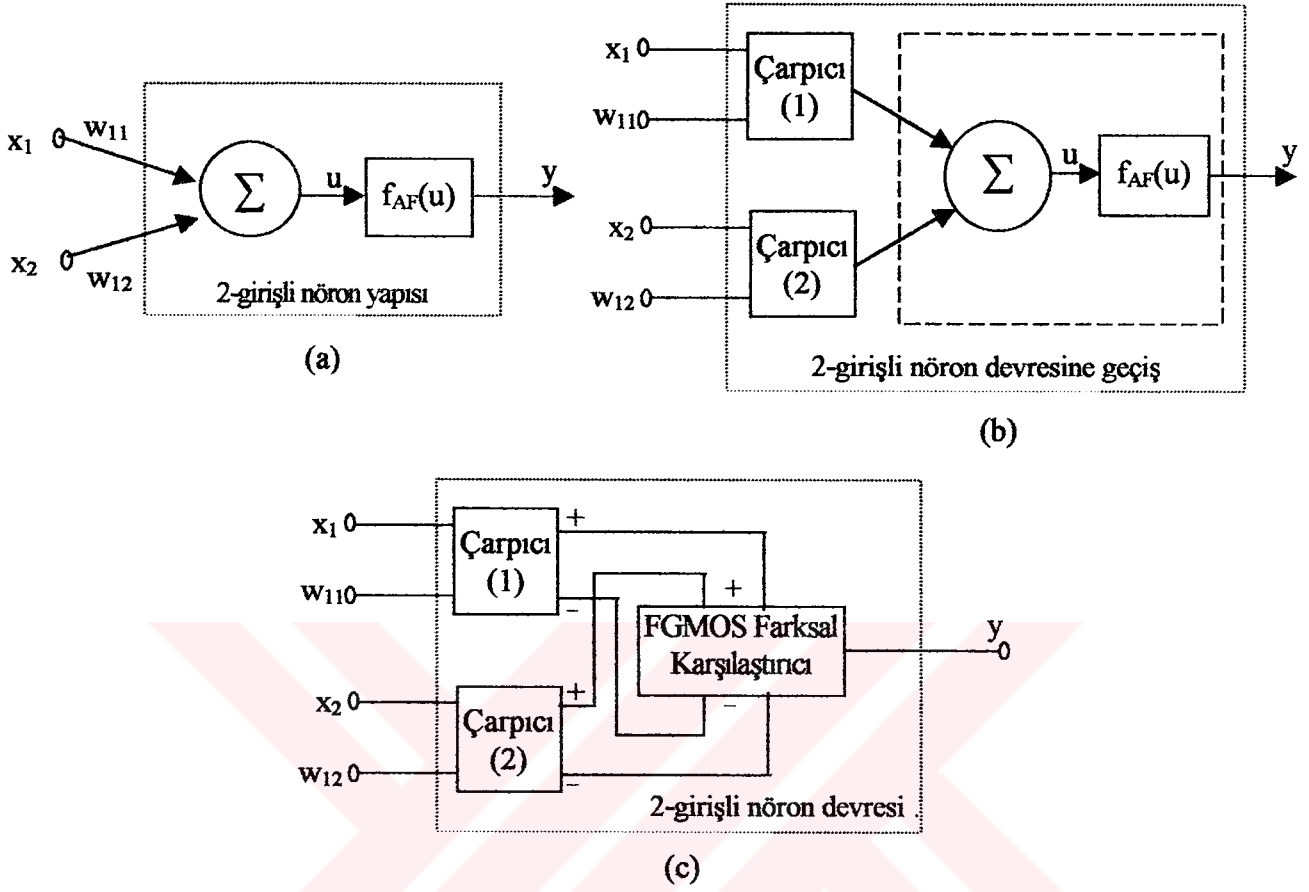
$$\text{herbir girişin kapasitif etki oranı} = \frac{\text{girişlerin toplam kapasitif etki oranı}}{\text{giriş sayısı}} \quad (4.28)$$

Örneğin iki girişli bir FGMOS farksal karşılaştırıcının giriş kapasitif etki oranları (4.28)'e göre 0.45'dir.



Şekil 4.14 FGMOS farksal karşılaştırıcı devresi

4.7.2 İki Çarpıcı ve Bir FGMOS Farksal Karşılaştırıcı Kullanılarak İki-Girişli Bir Nöronun Gerçeklenmesi



Şekil 4.15 (a) 2-girişli nöron yapısı, (b) 2-girişli nöron yapısıyla devresi arasındaki geçiş, (c) 2-girişli nöron devresi

Şekil 4.15a'da 2-girişli nöron yapısı görülmektedir, burada amaç bu görülen yapının elektronik olarak gerçekleştirilmesidir. Buradaki işlem, nöronun girişine gelen işaretle o girişe ilişkin ağırlığın çarpılması (ki burada çarpıcıya ihtiyaç duyulur) ve bu çarpımların toplanıp aktivasyon fonksiyonundan geçirilmesiyle sonuca ulaşılmasıdır. Bu işlemi yaparken ilk önce çarpma devrelerinin kullanılacağı açıktır (Şekil 4.15b ve c). Fakat bundan sonra toplama işlemi ve fonksiyondan geçirme işlemi yapılması gerekir ki bu da Şekil 4.15b'de ifade edilmiştir. Bunu elektronik olarak gerçeklemek istersek bir toplayıcı ve fonksiyon üretici gerekecektir. Fakat bu toplama ve aktivasyon fonksiyonundan geçirme işlemi Şekil 4.15c'de gösterildiği gibi tek bir tane FGMOS farksal karşılaştırıcı kullanarak gerçekleştirilip sonuç elde edilmiştir. Böylece iki-girişli nöron gerçekleştirilmiştir.

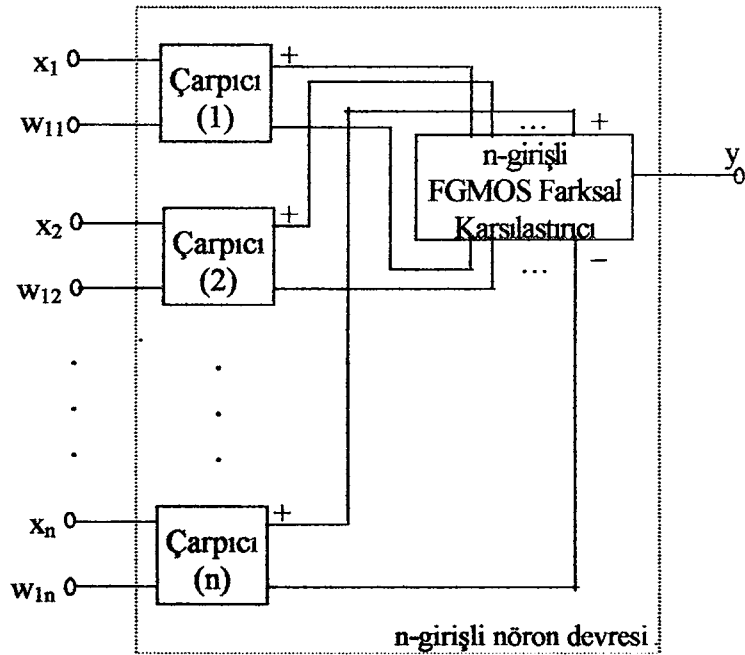
FGMOS karşılaştırıcının yaptığı işlemi açıklamak gerekirse; çarpıcıların çıkışı farksaldir, bu farksal çıkışlardan pozitif olanlar FGMOS karşılaştırıcının pozitif girişine, negatif çıkışlar da

negatif girişlere verilirse, FGMOS transistorların yapısı gereği pozitif ve negatif işaretler kendi aralarında toplanmış olur, bundan sonra da karşılaştırıcının yapısı gereği karşılaştırma işlemi yapılacağından, FGMOS karşılaştırıcının önce toplama yapıp daha sonra da karşılaştırma işlemi yaptığı görülür.

Gerçeklenen iki-girişli nöron devresinin simülasyonunda kullanılan HSPICE giriş dosyası EK-1’de verilmiştir. Bu devre tüm devre teknolojisine uygun olarak tasarlandığından, çarpıcı yapısında bulunan dirençler MOS transistorlarla gerçekleştirilmiş, böylece devre sadece transistor içerecek şekilde kurulmuştur.

4.8 N-Girişli Nöron Tasarımı

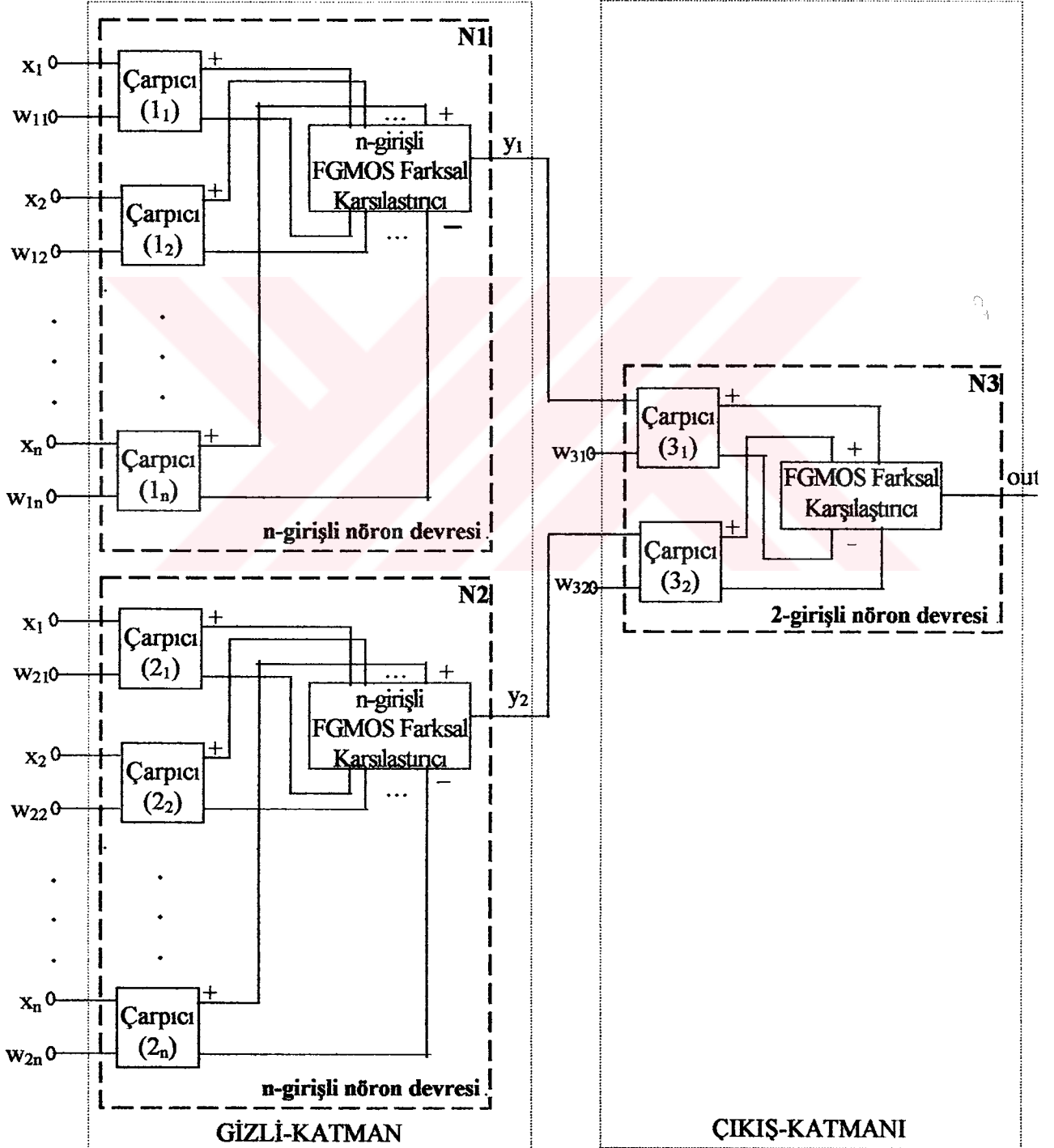
Şekil 4.16’da n-girişli nöron devresinde görüldüğü gibi, iki-girişli nöronun genelleştirilmesiyle n-girişli nöron elde edilebilir. Önceki bölümde anlatıldığı gibi iki-girişli nöronun yapısında iki tane çarpıcı vardır. Bunu, ‘giriş sayısı kadar çarpıcı olmalıdır’ şeklinde genelleştirmek mümkündür. Burada esas önemli olan nokta; bilindiği gibi çarpıcı çıkışlarının FGMOS farksal karşılaştırıcının girişlerini oluşturmasıdır, yani giriş sayısı değiştikçe çarpıcı sayısı değişecek ve bu da FGMOS karşılaştırıcının giriş sayısını değiştireceğinden, karşılaştırıcının girişlerinde bulunan FGMOS transistorların fiziksel yapıları değişecektir. Karşılaştırıcının girişindeki FGMOS transistorların giriş kapasitif etki oranları (4.28)’e göre hesaplanır.



Şekil 4.16 N-girişli nöron devresi

4.9 Yapay Sinir Ağı Devresi Tasarımı

Şekil 4.17’de n-girişli bir çıkışlı basit bir yapay sinir ağı devresi örnek olarak verilmiştir. Bu YSA, ileri beslemeli bir ağ olup, giriş katmanı, bir tane gizli katman ve çıkış katmanından oluşan üç-katmanlı bir ağ yapısına sahiptir. Ağ, n-girişli iki nöron ve bunların çıkışlarına bağlı iki-girişli bir nörondan oluşmaktadır. Görüldüğü gibi nöron yapıları çok modüler bir hal aldığından istenilen yapıda bir yapay sinir ağı tasarlanabilir.



Şekil 4.17 N-girişli bir-çıkışlı basit bir yapay sinir ağı devresi

4.10 Uygulamalar

YSA'da çok bilinen XOR probleminin gerçekleştirilmesi ve bir pattern sınıflayıcı uygulaması olarak zor bir data olan ama literatürde çok sık olarak önemle üzerinde durulan Iris çiçeğinin sınıflandırılmasının gerçekleştirilmesi ele alınacaktır.

Burada hedef, eşdeğer işlevi görecektir, tek gizli katmanlı, ileri beslemeli tipteki yapay sinir ağı devresini kurmak ve geriye yayılma algoritması (Back Propagation Algorithm: BPA) ve doğrusal olmayan tipteki aktivasyon fonksiyonlarını kullanarak yapay sinir ağı devresini istenen sınıflamayı yapacak şekilde eğitmektir.

Bilgisayar simülasyonu tümleşik devre tasarımında en önemli adımlardan biridir. Tümleşik devrelerin analiz ve tasarımının doğruluğu, devre elemanlarının bilgisayar simülasyonunun doğruluğu kadar olduğuna göre, yeterli doğrulukta gerçekleştirme tümleşik devre tasarımında gerekli bir faktördür.

Bu uygulamalarda sürekli değerli girişe sahip, ileri beslemeli bir yapay sinir ağı olan çok katmanlı algılayıcı (MLP) kullanılmıştır. Eğitim algoritması olarak da geriye yayılma algoritması kullanılmıştır. Geriye yayılım eğitim algoritması çok katmanlı algılayıcının aktif çıkışıyla istenen çıkış (hedef çıkış) arasındaki karesel ortalama hatanın minimize edilmesine dayanan bir gradyant (türev temelli arama) algoritmasıdır. Dolayısıyla bu yöntem sürekli türevi alınabilir doğrusal olmayan aktivasyon fonksiyonuna ihtiyaç gösterir. Öğrenme işlemi BPA ile ağırlıklar sürekli ayarlanarak gerçekleştirilir.

Lipmann 1987'deki çalışmasında (Lipmann, 1987) doğru ile ayrılabilen bölgelerin tanıtılabilmesi için bir, konveks bölgelerin tanıtılabilmesi için iki ve karmaşık karar bölgelerinin tanıtılabilmesi için üç katmanlı algılayıcının yeterli olduğunu göstermektedir.

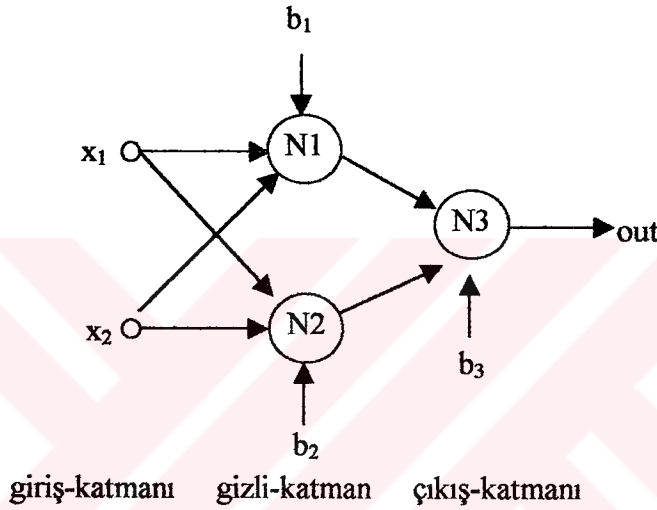
4.10.1 YSA'da Çok Bilinen XOR Problemi

Temel (tek-katmanlı) perseptron'un saklı katmanı yoktur. Bu nedenle, doğrusal olarak ayrılamayan giriş paternlerini sınıflayamaz. Ancak, nonlineer olarak ayrılabilen paternler genel bir olaydır ve sıkça rastlanır. İşte bu durum XOR probleminde genel problemin özel bir durumu olarak ortaya çıkar. Bilindiği gibi XOR, girişleri aynı olunca '0' çıkışı, farklı olunca '1' çıkışı verir. Düzlemde bir birim kare alınırsa, bu '0' çıkışlarına '0' sınıfı, '1' çıkışlarına da '1' sınıfı dersek, görülür ki karenin karşılıklı köşelerinde '0' sınıfları, diğer karşılıklı köşelerinde de '1' sınıfı bulunur. Bu durum bize, bu sınıflamanın doğrusal bir şekilde yapılamayacağını gösterir. Yani, tek bir doğruyla sınırlanırsa sınıflama işlemi gerçekleştirilemez,

en azından iki doğruya ihtiyaç vardır. Bu da, saklı katmanlı yapıya götürür, yani ‘Çok Katmanlı Algılayıcı’ (Multi Layer Perceptron:MLP) yapısı kullanılabilir.

4.10.1.1 Ağı Eğitilmesi ve Ağ Yapısının Belirlenmesi

Bu uygulama için geliştirilen YSA için Şekil 4.18’de gösterildiği gibi; giriş katmanı, bir gizli katman ve çıkış katmanı olmak üzere çok katmanlı algılayıcı yapısı kullanılmıştır. İki tane giriş olduğu için gizli katmanda iki tane nöron, bir çıkış olduğu için de çıkış katmanında bir nöron kullanmak uygun bulunmuştur. Aynı zamanda her bir nöronun bir bias girişi vardır. Ağırlıkların başlangıç değerleri seçimi, sonucu etkilemediği görüldüğünden rasgele alınmıştır.



Şekil 4.18 XOR problemini gerçekleştiren ağ yapısı

Ağ yapısı MATLAB programına girildikten sonra, girişlerin devreye uygun olması (beslemeden beslemeye giriş) ve hedef değerlerin de devrenin ortalama maksimum çıkışı olan 1.2V’a eşit olması açısından Çizelge 4.1’de verilen değerler kullanılmıştır. Eğitime yapılırken MATLAB’ın hazır öğrenme metodlarından ‘trainlm’ (Levenberg-Marquardt BPA), aktivasyon fonksiyonları olarak da ‘satlins’ (doymalı rampa fonksiyonu) kullanılmıştır. Elde edilen ağırlık değerlerine göre ağ yeniden test edildiğinde doğru sonuç verdiği görülmüş ve devre yapısı oluşturulmuştur.

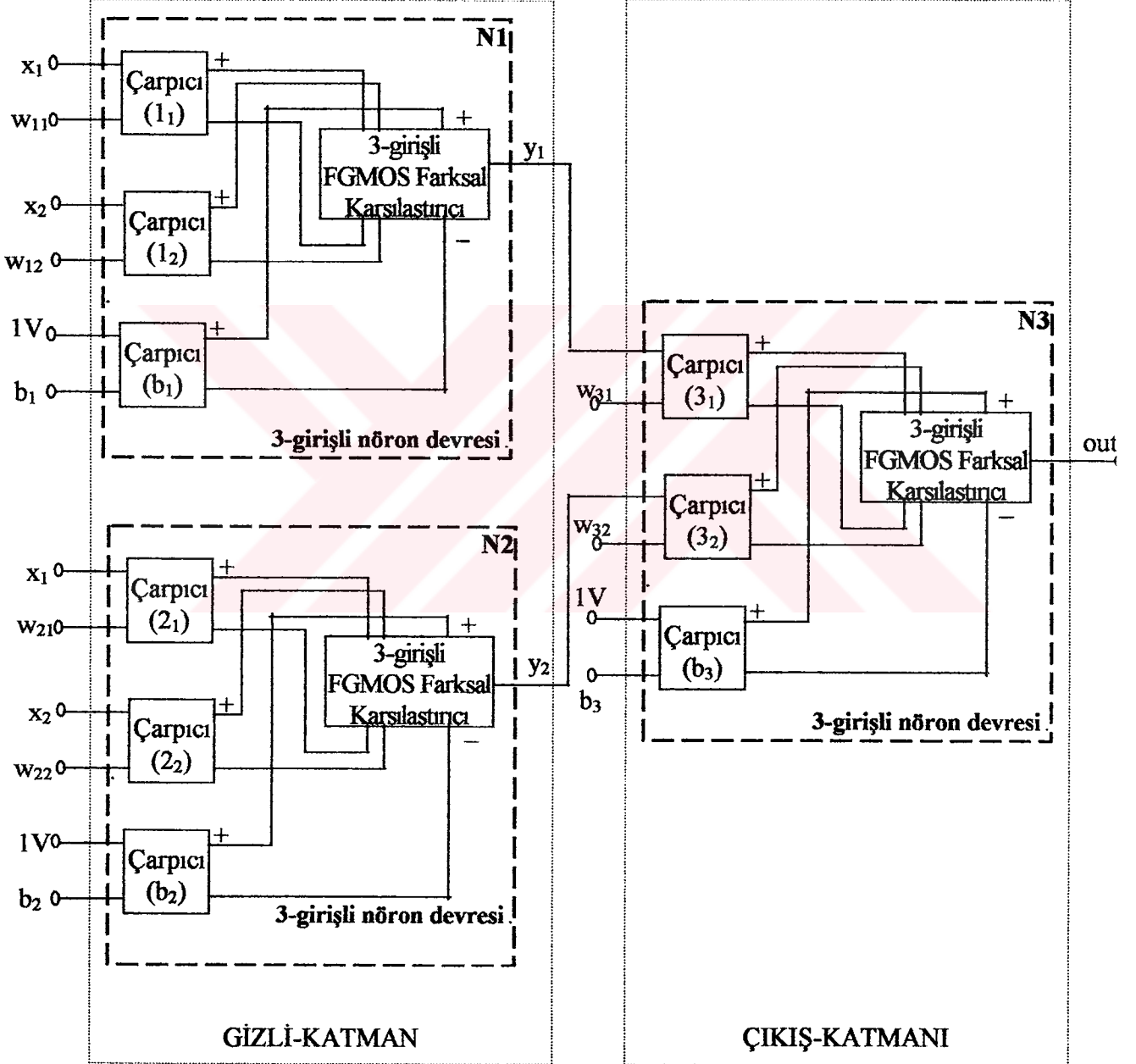
Çizelge 4.1 Giriş ve hedef çıkışların devrenin giriş-çıkış aralığına göre ayarlanması

$x1 \oplus x2$	y	$x1 \oplus x2$	out
$0 \oplus 0$	0	$-1.5 \oplus -1.5$	-1.2
$0 \oplus 1$	1	$-1.5 \oplus 1.5$	1.2
$1 \oplus 0$	1	$1.5 \oplus -1.5$	1.2
$1 \oplus 1$	0	$1.5 \oplus 1.5$	-1.2

4.10.1.2 XOR Problemini Gerçekleyen Devre Yapısı ve Simülasyon Sonuçları

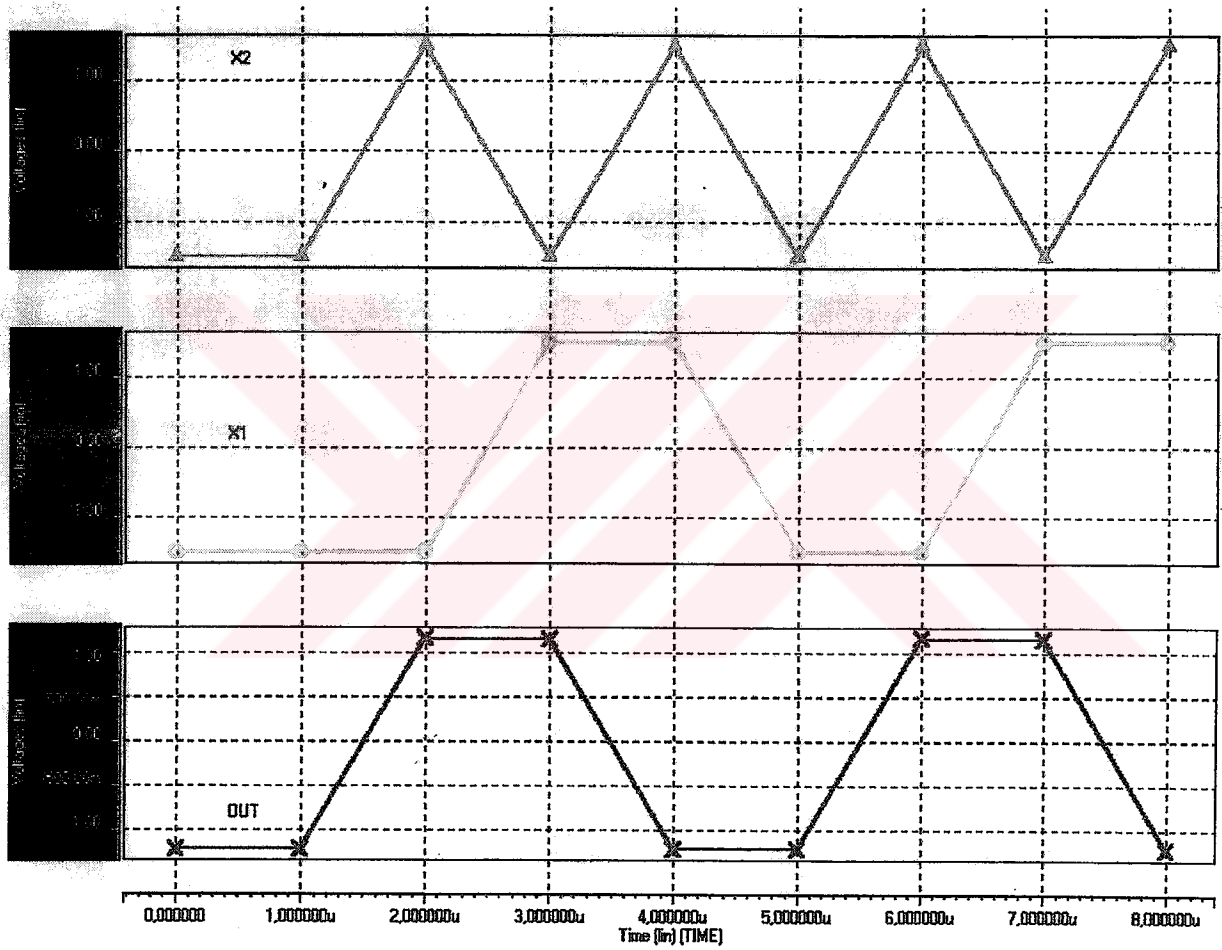
Normalde ağıın iki girişı var, fakat her nörona ait bir bias girişı olduğundan bu bias girişlerini 1V almak şartıyla giriş gibi alınabilir. Böylece üç tane üç-girişli nöron kullanarak XOR problemini çözebilen YSA'nı gerçekleştirilmiş olur.

XOR problemini gerçekleyen devre Şekil 4.19'da, simülasyonunda kullanılan HSPICE giriş dosyası ise EK-2'de verilmiştir.



Şekil 4.19 XOR problemini gerçekleyen devre

Şekil 4.20’de gerçekleştirilen XOR devresine ilişkin giriş-çıkış işaretleri görülmektedir. Yalnız burada dikkat edilmesi gereken bir husus; Çizelge 4.1’e dikkat edilirse buradaki girişler -1.5 ve $+1.5$ V değerlerini almaktadır, şekildeki girişler de çizelgeye uyumlu bir şekilde verilmiştir. Burada ilk giriş $1\mu s$ ’de başlar ve sırayla 2,3 ve 4. μs ’de son bulur (çizelgeye uygun olarak), sonra 5,6,7 ve 8. μs ’lerde aynen tekrarlanmıştır. Girişler zaman ekseninde süreksiz verildiği için ara değerlere bakılmaması gereklidir. Bir örnekle daha anlaşılır hale getirebiliriz ; 2. μs ’ye bakalım, x1 girişi: $-1.5V$, x2 girişi: $1.5V$ ve çıkış: $1,24V$ ’dur. Çizelgeye baktığımızda da bunu aynen görürüz ($-1.5 \oplus 1.5 = 1.2$).



Şekil 4.20 XOR devresinin girişleri ve çıkışı

4.10.2 Iris Çiçeğinin Sınıflandırılması

Iris, pattern tanıma literatüründe belki de en tanınmış veri tabanıdır. Fisher'ın makalesi (Fisher, 1938) bu alanda bir klasiktir ve hala sık sık referans verilir. İki yüze yakın çeşidi bulunan Iris'in dünyanın bir çok ülkesinde yüzlerce organizasyonu vardır. Data kümesi her sınıfın 50 örnek içerdiği 3 sınıftan oluşur ki; bu sınıflar da Iris'in tipini belirten sınıflardır. Iris, çok zor bir data olarak bilinir, şöyle ki; bir sınıf diğer ikisinden doğrusal olarak ayrılabilir fakat kalan iki sınıf birbirlerinden doğrusal bir şekilde ayrılamaz.

Her sınıf 50 örnekten oluştuğundan, toplamda 150 örnek içeren data kümesinde [3], her bir örnek, yaprak uzunluğu, yaprak genişliği, taç yaprak uzunluğu ve taç yaprak genişliği olmak üzere 4 özellikten oluşur. Sınıflar ise Iris'in çeşitleri olan, *Iris-Setosa*, *Iris-Versicolour* ve *Iris-Virginica*'dan oluşur.

Iris'e ilişkin orijinal giriş data kümesi EK-3'te verilmiştir.

4.10.2.1 MATLAB'da Ağın Eğitilmesi ve Ağ Yapısının Belirlenmesi (MLP - BPA)

Uygulamalarla ilgili bölümün (4.10) başlangıcından bu yana ve XOR probleminde anlatılan YSA yapıları ve eğitme algoritmaları ilgili her şey burada da geçerli olduğundan burada bir daha ayrıntısıyla değinilmeyecektir. Iris'in dört özelliği olduğu için YSA'nın da dört girişi olmalıdır, burada yine MLP yapısı kullanılacağından giriş sayısı kadar saklı nöron sayısı vermek denemeler sonucu uygun bulunmuştur. Çıkış katmanında ise bizi sınırlayan üç sınıfın olmasıdır. MATLAB'de değişik öğrenme algoritmalarıyla yapılan bir çok deneme sonucu tek gizli katmanla bu ağın kurulabilir olması ve demin belirttiğimiz nedenlerden dolayı saklı katmanda dört nöron, çıkış katmanında üç nöron bulunması gerektiği, Şekil 4.21'de gösterilen ileri beslemeli YSA yapısının kullanılmasını gerektirmiştir.

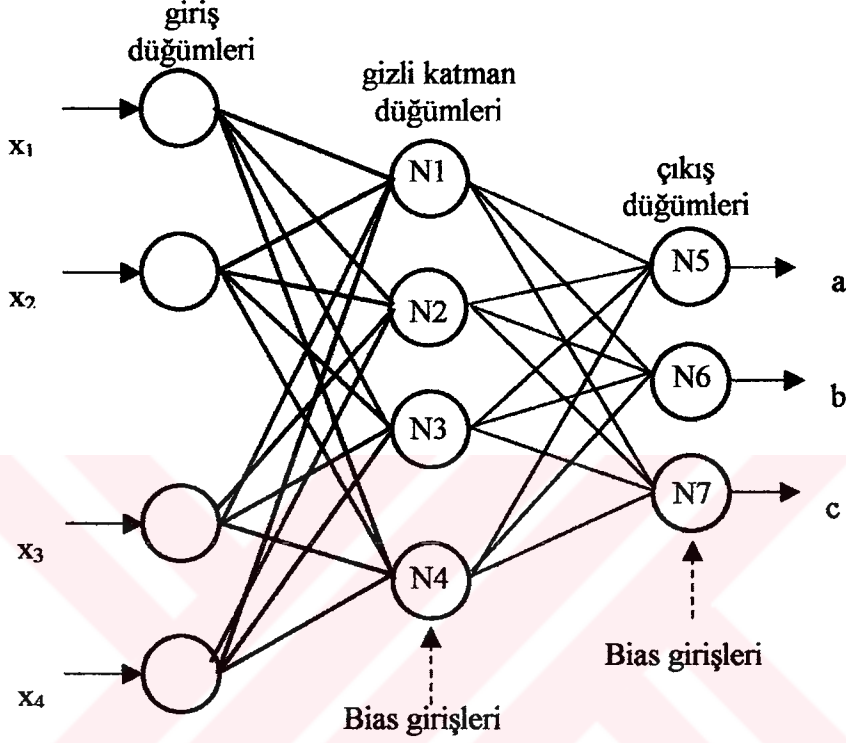
Giriş değerleri gerçekleştirilecek olan devrenin giriş değer aralığına uymadığı için ilk önce girişler şu formüle göre normalize edilmiştir:

Gerçek girişlerin değer aralığı: $\min_r \dots x_r \dots \max_r$ olsun. Burada $\min_r$, gerçek girişin minimumunu; $\max_r$, maksimumunu; x_r ise, normalize değerini bulacağımız giriş değerini belirtsin.

Normalize girişlerin değer aralığı ise: $\min_n \dots x_n \dots \max_n$ olsun. Burada da yine benzer şekilde yukarıda yazılanlar gerçek değer yerine normalize değerleri belirtsin. Buna göre normalize x_n değeri:

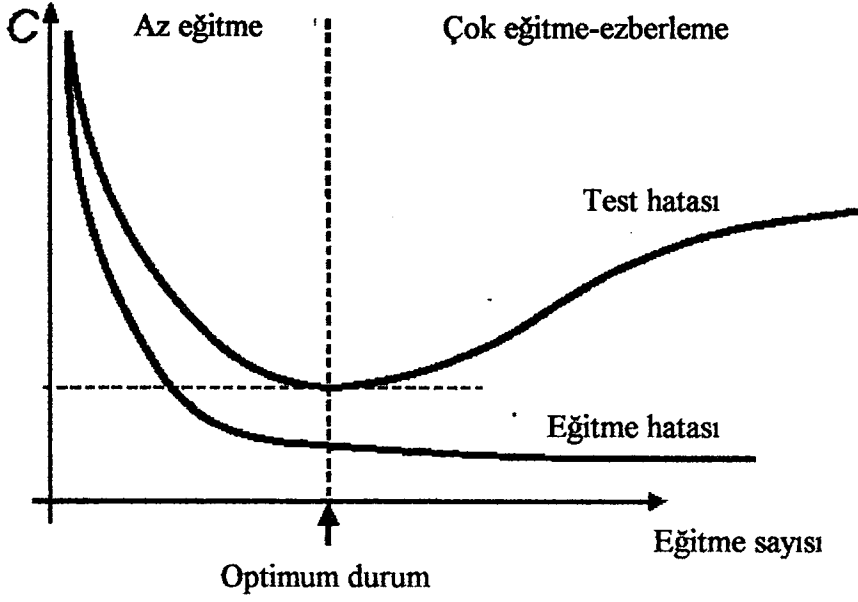
$$x_n = \min_n + (x_r - \min_r) \times \frac{(\max_n - \min_n)}{\max_r - \min_r} \quad (4.29)$$

Elde edilen normalize giriş değerleri, EK-4’de verilmiştir.



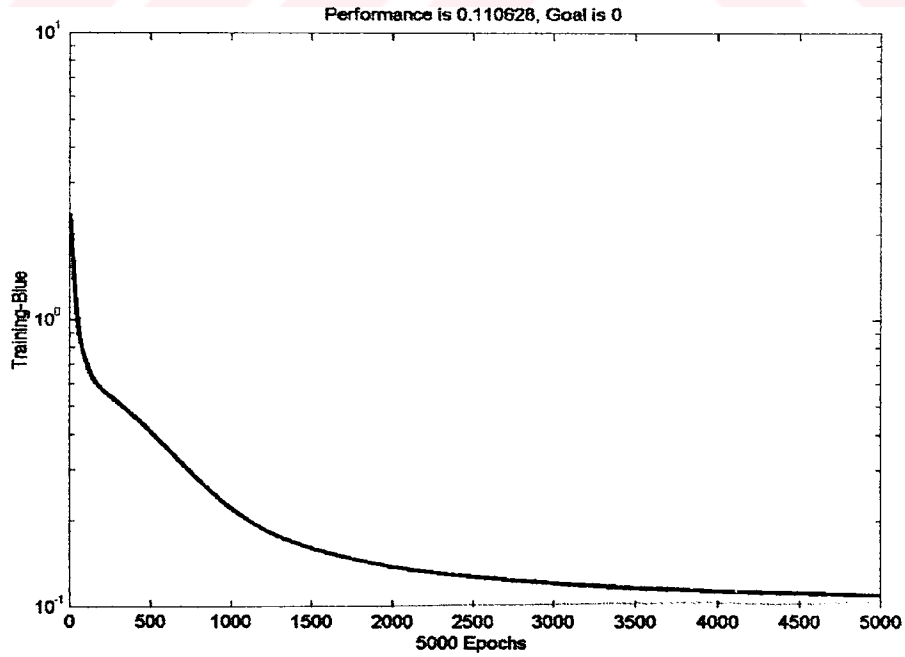
Şekil 4.21 Iris sınıflama işlemini gerçekleyen ileri beslemeli YSA

MATLAB programında eğitime yapılırken programa ait bir çok eğitime algoritması denemiştir; örneğin, XOR’daki gibi ‘trainlm’ yi kullandığımızda ağırlıklar çok fazla büyük çıktığından ağırlıkların da normalize işlemi yapılması gerekeceğinden bu ek bir külfet getirecektir ayrıca, ağırlıkların normalize edilmesi girişlerin normalize edilmesi kadar kolay olmamaktadır. Bunu için yine BPA’sı olan ‘traingd’ denenmiş, bunun da momentum katsayısını içermemesinden dolayı momentumu içeren algoritma olan ‘traingdm’ (gradient descent with momentum BPA) kullanılmıştır. Bu algoritmada öğrenme hızı $\eta = 0.01$ alınmış, öğrenme sırasında yerel minimumlara yakalanmamak ve en kısa zamanda eğitimi tamamlamak için momentum katsayısı $\alpha = 0.9$ alınmıştır. Şekil 4.22’de gösterildiği gibi çok fazla eğitimeyle, yanlış sonuçlar elde edilebilir. Bu, ağırlık eğitme datasını adeta ezberleyip test datasına gelince yanlış sonuçlar vermesiyle noktalanır (overfitting).



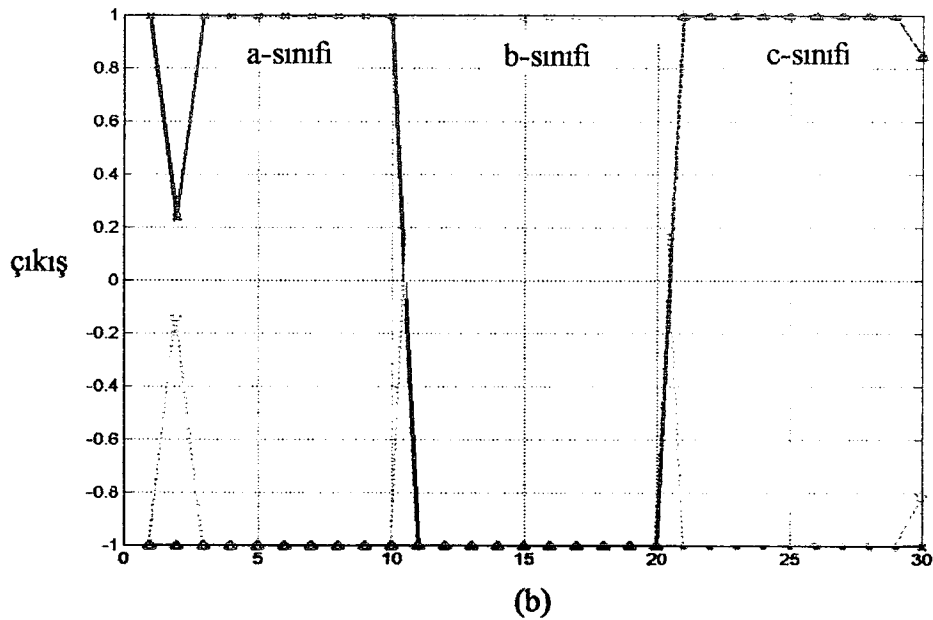
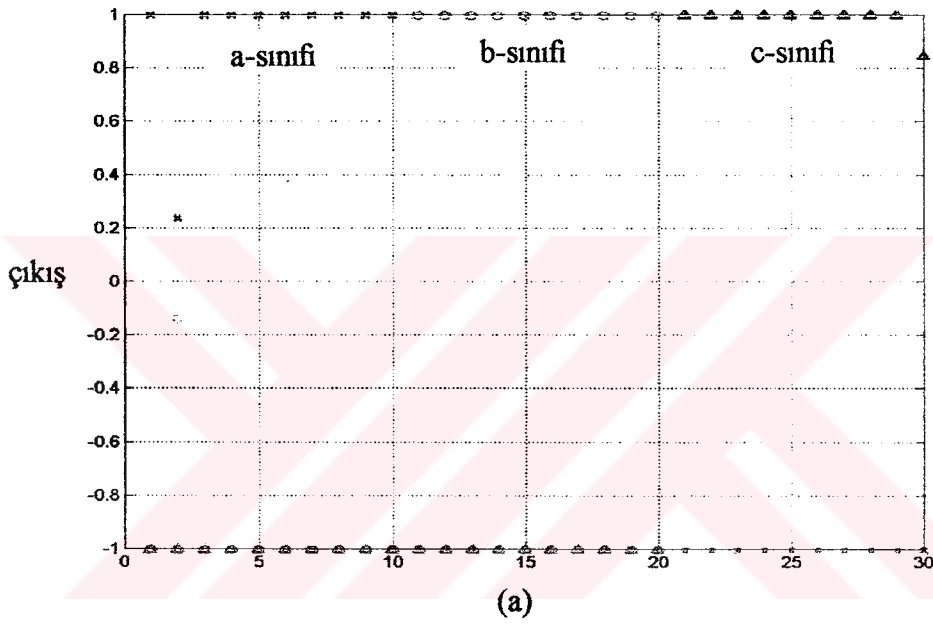
Şekli 4.22 Ağın eğitilmesinde optimum durum ve ezberleme

Eğitme yapılırken aktivasyon fonksiyonu olarak 'satlins' seçilmiştir. Çıkış hedefi olarak tasarımı yapılacak devrenin çıkışına uyması için 1.2 seçilmiştir. Üç sınıf olduğundan içlerinden bir tanesi 1.2 çıkarken diğerleri -1.2 çıkar, buradan da giriş datasının hangi sınıfa ait olduğu anlaşılır. Eğitim işlemi Şekil 4.23'te gösterildiği gibi 5000 adımda sonlandırılmış ve eğitim hatası 0.11 olarak çıkmıştır.

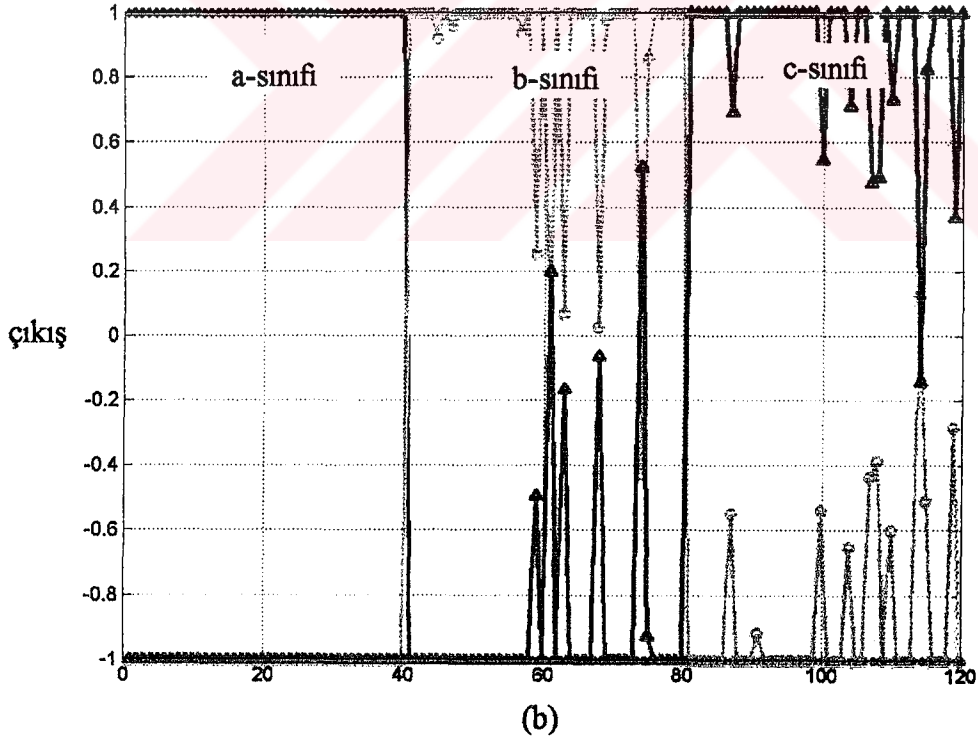
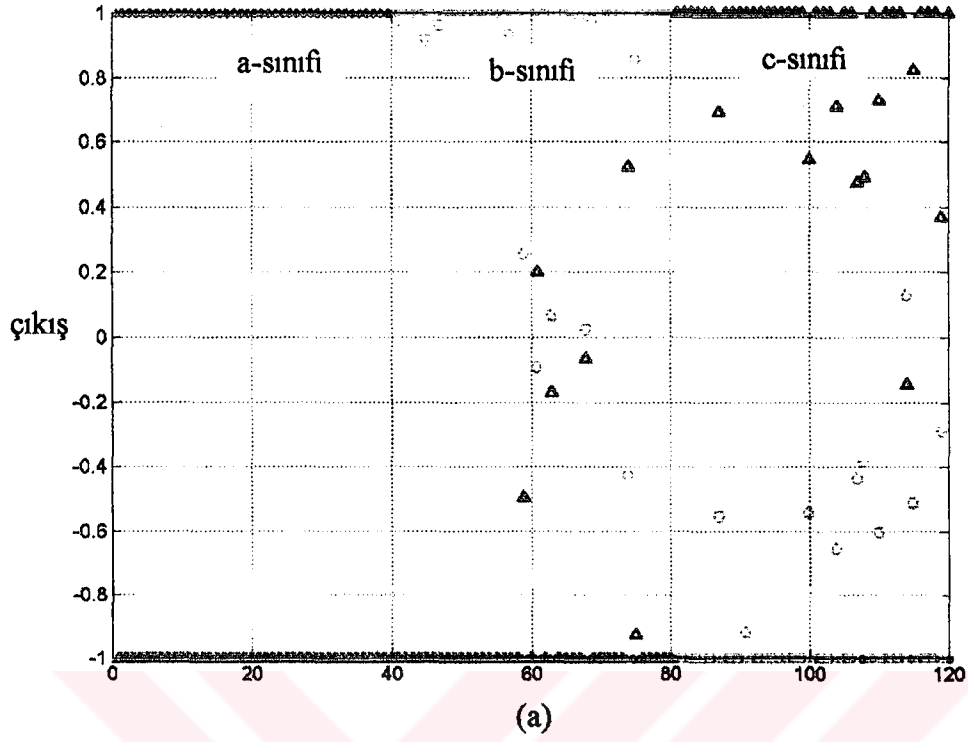


Şekil 4.23 Matlab programında eğitim işlemi

Ağ eğitilirken, normalize girişlerin her sınıfa ait ilk 40 tanesi toplamda 120 giriş eğitme datası olarak, geriye kalanlar ise test datası olarak seçilmiştir. Bu yapay sinir ağı, eğitme datalarıyla eğitildikten sonra test dataları kullanılarak test işlemine tabi tutulmuştur. Sonra elde edilen ağırlık değerleri kullanılarak test datası denenip Şekil 4.24a'da görüldüğü gibi dağılım grafiği çizdirilmiştir. Şekil 4.24b'de ise yine aynı sonuç, bir sonraki bölümde devre gerçeklemesi sonuçlarıyla kolay karşılaştırılabilsin diye bu sefer sürekli şekilde çizdirilmiştir. Aynı ağırlık değerleriyle bu sefer eğitme data kümesi denenmiş ve yine aynı şekilde grafikleri Şekil 4.25a ve b'de gösterildiği gibi çizdirilmiştir.



Şekil 4.24 (a) Test datasının dağılım grafiği, (b) test datasının sürekli grafiği



Şekil 4.25 (a) Eğitim verisinin dağılım grafiği, (b) eğitim verisinin sürekli grafiği

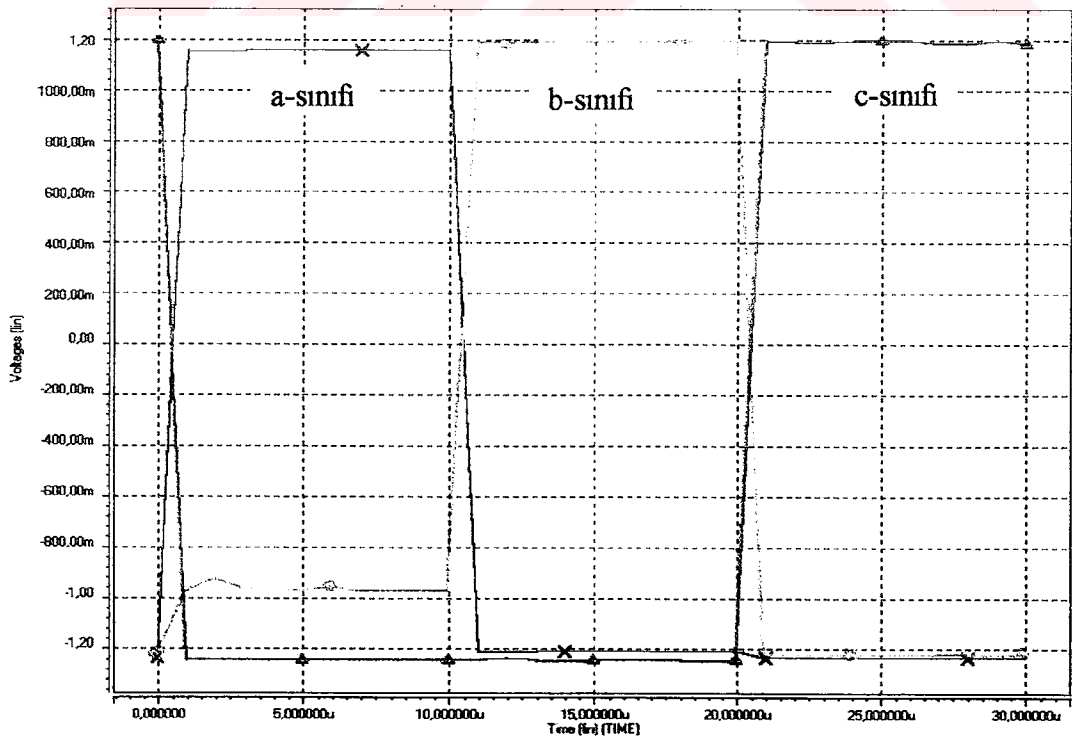
Bu sonuçlardan anlaşıyor ki; test verisinde hiç hata olmamasına karşın eğitim verisinde 3 tane hatalı çıkış görülmektedir, yani tasarladığımız bu ağ, MATLAB programında eğitilerek toplam veride 3/150'den %2'lik bir hata ile sonuçlanmış olur.

4.10.2.2 Iris Sınıflandırıcı ve Simülasyon Sonuçları

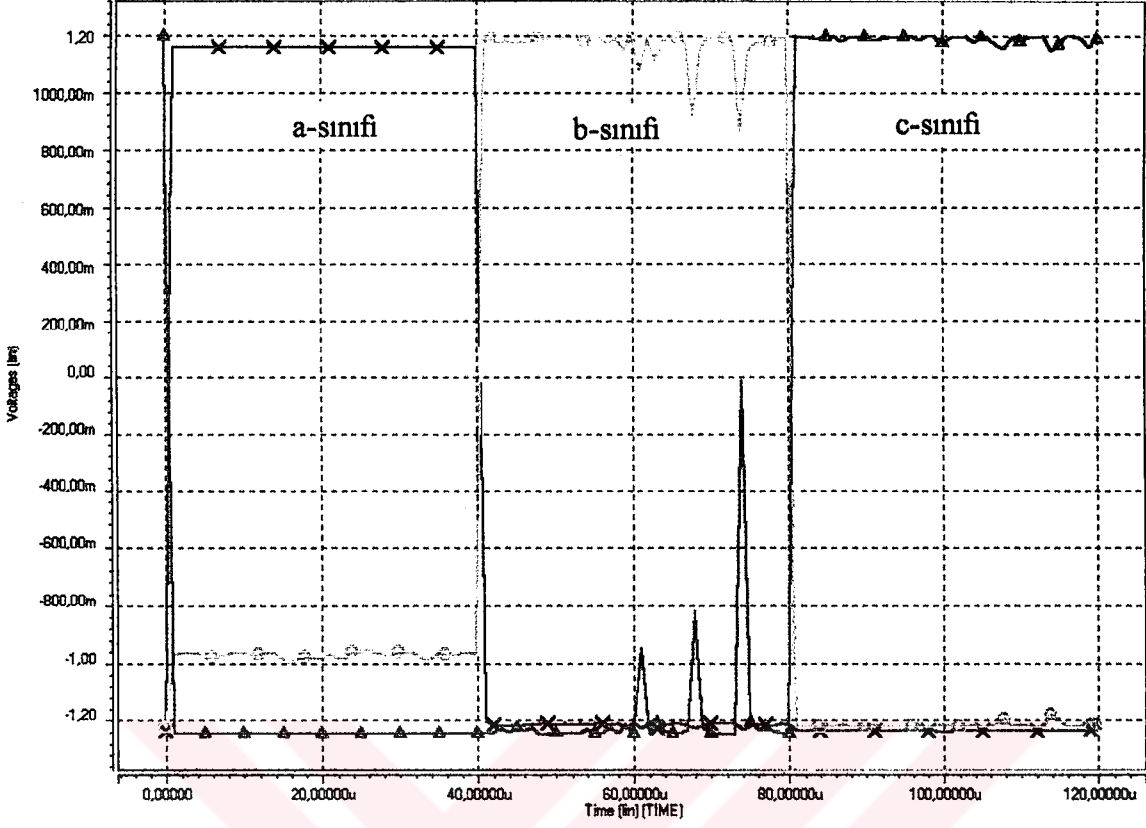
YSA normalde dört girişe sahip, fakat ayrıca her nörona ait bir bias girişi olduğu için -bu bias girişlerine 1V vermek şartıyla- bunları giriş gibi düşünebiliriz. Bias girişlerini kabul eden çarpıcı devrelerin diğer girişleri bias ağırlıklarını temsil ettiklerinden böylece devreye bias'ın etkisi girmiş olur, aynı zamanda bias girişlerine 1V verildiğinden çarpımda etkisiz olduğu için istenmeyen bir durum oluşturmaz. Böylece 4'ü gizli katmanda, 3'ü çıkış katmanında olmak üzere toplam 7 tane beş-girişli nöron kullanarak Iris sınıflayıcı YSA'nı gerçekleştirmiş oluruz. Devre yapısı, nöron modelleri modüler hal aldığından aynı XOR devresindeki gibidir, tek farkı orada üç-girişli nöronlar kullanılırken burada beş-girişli nöron kullanılmaktadır.

Tasarımı yapılan devre, MATLAB programından elde edilen ağırlık değerleriyle, eğitime datası için, test datası için ve tüm giriş datası için ayrı ayrı HSPICE programı kullanılarak simule edilmiştir. Simülasyon sonuçları Şekil 4.26-27 ve 28'de verilmiştir. Burada şekillerden de görüldüğü gibi en ilgi çekici nokta, data çok zor bir data olmasına karşın devrenin bütün dataları doğru sınıflayabilmesidir, yani hata oranı %0'dır, halbuki MATLAB'de yapılan denemede toplam üç datada hata çıkmıştı. Grafiklerde girişler XOR devresinin sonuçlarında uzunca bahsettiğimiz gibi 1µs'den başlamaktadır.

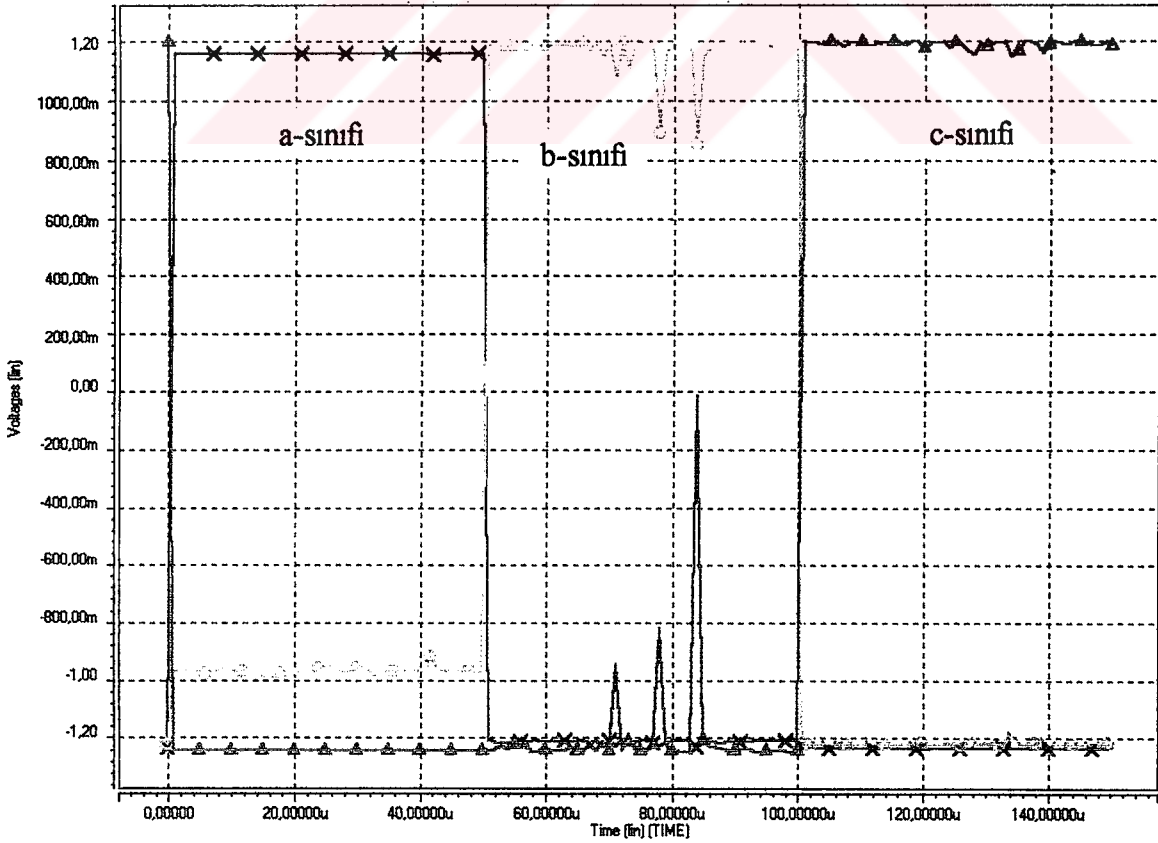
Son olarak, diğerlerine de örnek teşkil etmesi amacıyla tüm giriş datası için yapılan simülasyonda kullanılan HSPICE giriş dosyası EK-5'te verilmiştir.



Şekil 4.26 Test datasının simülasyon sonucu



Şekil 4.27 Eğitim verisinin simülasyon sonucu



Şekil 4.28 Tüm giriş verisinin simülasyon sonucu

5. SONUÇLAR

Sınıflama işlemi yapan bir yapay sinir ağı tasarlanmıştır. Bu tasarım, Vlassis ve Siskos'un önerdiği FGMOS transistor kullanılan zayıflatıcı ve kare alıcıdan oluşan bir çarpıcıdan başlayarak FGMOS nöronun gerçekleştirilmesine kadar aşama aşama oluşturulmuştur. FGMOS nöronun içinde doğrusallığı çok iyi olan ve zayıflatıcıların kullanılmasıyla beslemeden beslemeye giriş verilebilen analog bir dört-bölgeli çarpıcı vardır. Buna ek olarak, FGMOS transistorların kullanılması ve çarpıcıların çıkışının farksal olması nöron içinde kullanılacak olan yapıların FGMOS farksal karşılaştırıcı olması sonucunu doğurmuştur. FGMOS karşılaştırıcı kullanılması, önceki nöron tasarımlarında çarpıcı çıkışlarını toplamak ve aktivasyon fonksiyonunu gerçekleştirmek için kullanılan işlemsel kuvvetlendirici ve fonksiyon üretici gibi devrelerin kullanılması ihtiyacını ortadan kaldırarak hem yapıyı daha sade bir hale getirme, hem de az transistor kullanılması gibi avantajlar sağlamıştır.

YSA tasarımında dikkat edilmesi gereken noktalardan biri de kullanılan nöron yapılarının ağırlıklar, girişler ve önceki katmanlardan gelen işaretler gibi giriş etkileri dışında devrenin iç yapısına bağlı olarak ortaya çıkan, örneğin istenmeyen akım çekme durumları gibi (yani yükleme durumları gibi), sorunların nöronların arasında olmaması gerekir. Çünkü sinir ağlarında nöronların tıpkı bir adacık gibi çevresel sistemlerden giriş etkileri dışında izole edilmiş gibi kabul edilir. Burada tasarlanan nöron yapılarının, çarpıcı ve FGMOS karşılaştırıcı olarak ana iki devreden oluştuğu düşünülürse, ilk olarak çarpıcının iç yapısına bakıldığında kullanılan zayıflatıcı ve basit kare alıcılar arasında her hangi bir akım çekme gibi bir etkileşim olmadığı görülür. Çünkü zayıflatıcının çıkışları kare alıcının MOS transistorlarının geçitlerine bağlıdır, MOS transistorların ihmal edilebilecek düzeyde akım çektiği düşünülürse etkileşim olmadığı kabul edilebilir. Aynı ilişki çarpıcı ve FGMOS karşılaştırıcı arasında da vardır. Bunu genişletirsek yine aynı ilişkinin nöronlar arasında da olduğu görülür ve böylece modüler bir nöron yapısı elde edilerek nöronların birbirine bağlanmasıyla YSA'lar kolayca oluşturulabilir.

Simülasyonlarda devre parametreleri veya elemanları arasında herhangi bir dengesizlik olmadığı varsayılmıştır. Çarpıcının çok küçük olan doğrusallık hatası, kanaldaki delik hareket yeteneğinin, geçidin oluşturduğu dikine elektrik alanla değişmesinden kaynaklanmaktadır.

Iris sınıflayıcı devrenin, giriş datası zor ayrıştırılabilir bir data olduğu halde sınıflama işlemini sıfır hatayla yapması simülasyon sonuçlarından gözükmemektedir. MATLAB programında yazılım olarak yapılan sınıflama işleminde ise aynı ağ yapısının 150 datada üç datayı yanlış sınıflaması da gösterilerek karşılaştırma yapma olanağı sağlanmıştır. Burada yazılım olarak

yapılan simülasyonda hata varken donanım gerçeklemede hata olmaması şöyle açıklanabilir: sinir ağlarında hesaplama işlemi sonuçta belli toplamlar şeklinde olduğundan belli oranda hataya göz yumulabilir, fakat MATLAB ayrıntılı matematiksel hesaplar yaptığından kullanılan algoritma da hatanın geriye yayılması olduğundan hatayı datanın karışan noktalarında tam olarak minimize edemeyebilir, donanımda ise bu nokta daha kabaca ele alındığından karşılaştırmalı çıkışıyla, çıkış hemen ya en pozitif gerilime ya da en negatif gerilime çekildiğinden daha doğru sonuçlar elde edilmiştir. Aynı zamanda devrede kullanılan aktivasyon fonksiyonu yapısı bire bir MATLAB'da tanımlanan aktivasyon fonksiyonlarına uymaz. İleride gereksinime göre devrenin gerçekleştirdiği aktivasyon fonksiyonunun benzetimi yapılarak MATLAB'da devreye ait olan yeni bir aktivasyon fonksiyonu oluşturulur ve ağ yapısı bire bir donanımla uyacak şekilde yazılımla da simüle edilmiş olur.

Özel olarak Iris sınıflayıcı devreye bakılacak olunursa, burada yarı-özel (semi-custom) yapı kullanılmıştır. Ağırlıklar dışarıdan girildiği için değiştirilebilmektedir, böylece aynı veya daha az karmaşık bir ağ yapısına sahip farklı bir sınıflama veya herhangi bir işlem gerçekleştirmek mümkündür. Bu devre tüm devre olarak üretildiğinde içindeki yapılar modüler olduğu için içeriden herhangi bir uç alınarak ağın içindeki akış kontrol işlemi daha kolay bir şekilde yapılabilir. Örneğin bu tüm devre aynı zamanda çarpıcı olarak da kullanılabilir. Ayrıca bu tüm devreler birbirine bağlanmak suretiyle değişik ağ yapılarına sahip devreler de oluşturulabilir. Eğer Iris sınıflayıcı daha özel bir yapı şeklinde üretilmek istenirse (full-custom), mesela ağırlıkları sıfır olanlar hiç devreye katılmaz böylece her sıfır ağırlık için çarpıcı sayısı bir azalır. Ağırlıklar dışarıdan değil de MOS gerilim bölücüler kullanarak tüm devre içinde yapıldığında çok daha özel bir yapı elde edilir, fakat bu hiç kullanışlı bir yapı olmaz.

Çip içi öğrenme (on-chip-learning) ve çip dışı öğrenme (out-chip-learning) karşılaştırılması yapılırsa, çip-içi öğrenmenin daha hızlı ve entegre bir sistem olması avantajına karşın sadece öğrenme işlemi için ek bir donanıma ihtiyaç duyulması da dezavantajıdır, çip-dışı öğrenme de ise ağırlıklar bilgisayarda öğrenme işlemiyle belirlendiğinden diğer avantajlarından feragat edilirse böyle bir ek donanıma ihtiyaç yoktur (Annema, 1995). Burada yapılan çalışmada bilindiği gibi çip dışı öğrenme kullanılmıştır.

İlerideki çalışmalar olarak ağırlıkların kendi içinde saklanıp, eğitilebileceği ve aynı zamanda bu işlemin daha basit bir yapı olarak tasarlanmasında eşik altı FGMOS transistorlar kullanılacağı tasarımlar yapılabilir. Genel amaçlı bir hedef varsa sayısal tüm devre tasarımlarının üstün yanları olabileceğinden, aynı zamanda analog ve sayısal tasarımların bir arada yapıldığı hibrit tasarımlar üzerinde durulacaktır.

KAYNAKLAR

- Alspector, B., Allen, R. B., Jayakumar, A., Zeppenfeld, T. Ve Meir, R., (1991) "Relaxation Networks for Large Supervised Learning Problems", In Advances in Neural Information Processing Systems, San Mateo, CA: Morgan Kaufmann, 1015-1021.
- Andreou, A., G., (1992) "Minimal Circuit Models of Neurons, Synapses and Multivariable Functions for Analog VLSI Neuromorphic Computation", Report JHU/ECE-92-13, Johns Hopkins University, Baltimore, MD.
- Annema, A. J., (1995) "Feed Forward Neural Networks, Vector Decomposition Analysis, Modelling and Analog Implementation", Boston, MA: Kluwer Academic Publishers.
- Bailey, J. Ve Hammerstrom D., (1988) "Why VLSI Implementation of Associative VLCNs Require Connection Multiplexing", International Conference on Neural Networks, Vol2, San Diego CA, 1280.
- Bayraktaroğlu, İ., Öğrenci A. S., Dünder G., Balkır S. ve Alpaydın E., (1998) "ANNSyS: an Analog Neural Network System", Neural Networks 12, 325-338.
- Bohen, K. A., Pouliquen, P. O., Andreou A. G., ve Jenkins, R. E., (1989) "A Heteroassociative Memory Using Current-Mode Analog VLSI Circuits", IEEE Trans. On Circuits and Systems, vol. CAS30, 747-755.
- Boser, B. A., Sackinger, E., Bromley J., Lecun Y. Ve Jackel, L. D., (1992) "Hardware Requirements For Neural Network Pattern Classifiers", IEEE Micro 12, 32-40.
- Choi, J., Bangs, S. H. ve Sheu, B. J., (1993) "A Programmable Analog VLSI Neural Network Processor for Communication Receivers", IEEE on Neural Networks 4, 178-191.
- Churchland, P. S., (1986) "Neurophilosophy: Toward A Unified Science of The Mind/Brain", Cambridge, MA: MIT Press.
- Churchland, P. S. Ve Sejnowski, T. J., (1992) "The Computational Brain", Cambridge, MA, MIT Press.
- Cocke, J. ve Marstein, V., (1990) "The Evolution of RISC Technology at IBM", IBM Journal of Research and Development 34, 4-11.
- Fisher, R.A., (1936) "The Use of Multiple Measurements in Taxonomic Problems", Annual Eugenics, 7, Part II, 179-188.
- Giles, J., (2001) "Think Like a Bee", Nature 410, 510-512.
- Giustolisi G., Palmisano G., ve Palumbo G., (1997) "A Novel CMOS Voltage Squarer", In Proc. IEEE Int. Symp. Circuits. and Systems ISCAS'97, 253-256.
- Hammerstrom, D., (1992) "Electronic Neural Network Implementation", Tutorial No:5, International Joint Conference on Neural Networks, Baltimore, MD.
- Haykin, S. S., (1999) "Neural Networks: A Comprehensive Foundation", Prentice Hall, 2. ed.
- Hieda, K., Wada, M., Shibata, T., ve Iizuka, H., (1983) "A New EEPROM Cell with Dual Control Gate Structure," in Digest of Technical Papers of the 1983 Symposium on VLSI Technology, pp. 114-115.

- Hirose K., Yasuura H., (1996), "A Comparison of Parallel Multipliers with Neuron MOS and CMOS technologies", In Proc. of IEEE Asia Pasific Conference on Circuits and Systems, 488-491.
- Holler, M. A., Tam, S., Castro, H. ve Benson, R., (1989) "An Electrically Trainable Artificial Neural Network (Etann) with 10240 'Floating Gates' Synapses", International Joint Conference on Neural Networks, Vol. 2, San Diego, CA, 191-196.
- Kahng, D. ve Sze, S. M., (1967), "A Floating-Gate and Its Application to Memory Devices," The Bell System Technical Journal, vol. 46, no. 4, 1288-1295.
- Lee, J. C., Sheu, B. J., Fang, W. C. ve Chellappa, R., (1993) "VLSI Neuroprocessor For Video Motion Dedector", IEEE Trans. On Neural Networks 4, 178-191.
- Lipmann, R., (1987) "An Introduction to Computing with Neural Networks", IEEE ASSP Magazine, (4) 14.
- Macq, D., Verleysen, M., Jespers, P., ve Legat, J.-D., (1993) "Analog Implementation of Cohonen Map with On-Chip Learning", IEEE Trans. On Neural Networks 4, 456-461.
- Marr, D., (1982) Vision, New york, Freeman.
- Mead, C., (1989) "Analog VLSI and Neural Systems", Reading, MA: Addison-Wesley.
- Mehrvarz, H. R. ve Kwok, C. Y., (1996) "A Novel Multi-Input Floating-Gate MOS Four-Quadrant Analog Multiplier", IEEE Journal of Solid State Circuits, 31(8), 1123-1131.
- Murray, A. F., Del Corso, D. ve Tarresenko, L., (1991) "Pulse-Stream VLSI Neural Networks Mixing Analog and Digital Techniques", IEEE Trans. on Neural Networks, London, UK, 27-32
- Ochiai, T. ve Hatano, H., (1999) "A Proposition On Floating-Gate Neuron MOS Macromodelling For Device Fabrications", IEICE Trans. Fundamentals, 2485-2491.
- Ohmi, T., (1997) "Integrating Intelligence On Silicon Electronic Systems. An Inter-University Cooperative Research Project For Innovative Process, Device, Circuit and System Technologies", Computers Elect. Engng. vol 23, no 6, 371-379.
- Ramacher, U., Beichter, J. ve Bruls, N., (1991) "Architecture of General Purpose Neural Signal Processor", International Joint Conference on Neural Networks, vol.1, Seattle, 443-446
- Ramirez, A. J., (1996) " $\pm 0.75V$ BiCMOS Four Quadrant Analog Multiplier with Rail- Rail Input Signal-Swing," in Proceedings of the 1996 IEEE International Symposium on Circuits and Systems, Atlanta, GA, vol. 1, 242-245.
- Shibata, T. ve Ohmi, T., (1992) "A Functional MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations," IEEE Transactions on Electron Devices, vol. 39, no. 6, 1444-1455.
- Sin, C., Kramer, A., Hu, V., Robert, R. C. ve Ko, P. K., (1992) "Eeprom as an Analog Device with Particular Applications in Neural Networks", IEEE Transactions on Electron Devices 39(6), 1410-1417.
- Vlassis, S. ve Siskos, S., (2001) "Differential-Voltage Attenuator Based on Floating-Gate MOS Transistors and Its Applications", IEEE Transactions on Circuits and Systems-I, 48(11), 1372-1378.

Watanabe, T. K., Kimura, K., Aoki, M., Sakata, T. ve ITO, K., (1993) "A Single 1.5 V Digital Chip for a 10^6 Synapsa Neural Network", IEEE Trans. On Neural Networks 4, 387-393.

Werner, W., Prange S. J., Thewes, R., Wohrlab, E. Ve Luck, A., (1996) "On The Application of the Neuron MOS Transistor Principle for Modern VLSI Design", IEEE Transactions on Electron Devices 43(10), 17000-1708.

Yang, K. ve Andreou, A. G., (1994.) "A Multiple Input Differential Amplifier Based on Charge Sharing on a Floating-Gate MOSFET," Journal of Analog Integrated Circuits and Signal Processing, vol. 6, no. 3, 197-208.

Yuh, J. D. ve Newcomb, R. W., (1993) "A Multilevel Neural Network for A/D Conversion", IEEE Trans. On Neural Networks 4, 470-483.

Zeck, G.ve Fromherz,P. (2001) "Noninvasive Neuroelectronic Interfacing with Synaptically Connected Snail Neurons on a Semiconductor Chip", PNAS,98,10457-10462.

INTERNET KAYNAKLARI

[1] <http://www.biochem.mpg.de/mnphys/index.html>

[2] <http://www.physik.uni-regensburg.de/aktuell/KollWS0001.div/Fromherz.html>

[3] <http://www-old.ics.uci.edu/pub/ml-repos/machine-learning-databases/>

EKLER

- Ek 1 İki-girişli nöron devresinin simülasyonunda kullanılan HSPICE giriş dosyası
- Ek 2 XOR'un simülasyonunda kullanılan HSPICE giriş dosyası
- Ek 3 Iris' e ait orijinal data dosyası
- Ek 4 Iris'in simülasyonunda kullanılan normalize edilmiş girişlere ait data dosyası
- Ek 5 Iris'in simülasyonunda kullanılan HSPICE giriş dosyası



Ek 1 İki-girişli nöron devresinin simülasyonunda kullanılan HSPICE giriş dosyası

```

*neuron-2in
.option post
xml x1 y1 dc1 da1 s1 d2 multiplier
xm2 x2 y2 dc2 da2 s1 d2 multiplier
.subckt multiplier x2 x1 dc da s1 d2
xs1 0 x1 dc da s1 d2 squarer
xs2 x1 x2 da dc s1 d2 squarer
xs3 x2 0 dc da s1 d2 squarer
.subckt squarer x1 x2 dc da s1 d2
xp1 fg1 x1 y1 d1 s1 vpmos
xp2 fg2 x2 d2 d2 d1 vpmos
xp3 fg3 x2 y1 d3 s1 vpmos
xp4 fg4 x1 d2 d2 d3 vpmos
.subckt vpmos fg x y d s
mpmos d fg s s modp w=20u l=5u
g1 0 1 x 0 1
r1 0 1 0.1
g2 0 2 y 0 1
r2 0 2 0.8
gs 0 3 s 0 1
rs 0 3 0.099
gd 0 4 d 0 1
rd 0 4 1e-3
gsub 0 5 sub 0 1
rsub 0 5 0.001
vsub sub 0 dc -1.5
e fg 0 poly(5) 1 0 2 0 3 0 4 0 5 0 (-0.063) 1 1 1 1 1
rx x 1 1e10
ry y 2 1e10
rsu sub 5 1e10
.model modp pmos level=3
+js=0.0062 cj=0.000378 cjsw=2.2e-10 tox=250e-10 ld=0.06e-6 wd=0.6e-6 uo=230
+vto=-0.63 theta=0.17 rs=120 rd=120 delta=0.4 xj=0.3e-6 vmax=0
+eta=0.01 kappa=0.07 nfs=4e11 gamma=0.50 phi=0.35
+jsw=2.2e-9 tpg=-1
.ends vpmos
vb y1 0 dc 0.1875
ma da d1 sa sa modn w=33u l=5u
mc dc 0 sa sa modn w=66u l=5u
mb da d3 sa sa modn w=33u l=5u
m1 sa g1 d2 d2 modn w=150u l=1.5u
m2 g1 g1 d2 d2 modn w=150u l=1.5u
m3 g1 g1 s1 s1 modp w=17.5u l=2u
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends squarer

```

```

mo1 da da s1 s1 modp w=22u l=1.5u
mo2 dc dc s1 s1 modp w=22u l=1.5u
.ends multiplier
v1 x1 0 dc 1.5
v2 y1 0 dc 1.5
v3 x2 0 dc 1.5
v4 y2 0 dc 1.5
vdd s1 0 dc 1.5
vss d2 0 dc -1.5
.dc v1 -1.5 1.5 .01
*****FG farksal comperator yapisi*****
xn1 fg1 dc1 dc2 g3 d5 vnmos
xn2 fg2 da1 da2 out d5 vnmos
m3 g3 g3 s1 s1 modp w=8u l=5u
m4 out g3 s1 s1 modp w=8u l=5u
m5 d5 bias d2 d2 modn w=10u l=5u
vbias bias 0 dc 1.5
.subckt vnmos fg x y d s
mnmos d fg s s modn w=5u l=5u
g1 0 1 x 0 1
r1 0 1 0.45
g2 0 2 y 0 1
r2 0 2 0.45
gs 0 3 s 0 1
rs 0 3 0.099
gd 0 4 d 0 1
rd 0 4 1e-3
gsub 0 5 sub 0 1
rsub 0 5 0.001
vsub sub 0 dc 1.5
e fg 0 poly(5) 1 0 2 0 3 0 4 0 5 0 (0.68) 1 1 1 1 1
rx x 1 1e10
ry y 2 1e10
rsu sub 5 1e10
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends vnmos
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.end

```

Ek 2 XOR'un simülasyonunda kullanılan HSPICE giriş dosyası

```

*xor
.option post
x3in1 x1 w11 x2 w12 x3 w13 out1 s1 d2 neuron3in
x3in2 x1 w21 x2 w22 x3 w23 out2 s1 d2 neuron3in
x3in3 out1 w31 out2 w32 b2 w3b out s1 d2 neuron3in
.subckt neuron3in x1 y1 x2 y2 x3 y3 out s1 d2
xc1 dc1 dc2 dc3 da1 da2 da3 out s1 d2 fgcomp
xm1 x1 y1 dc1 da1 s1 d2 multiplier
xm2 x2 y2 dc2 da2 s1 d2 multiplier
xm3 x3 y3 dc3 da3 s1 d2 multiplier
.subckt multiplier x2 x1 dc da s1 d2
xs1 0 x1 dc da s1 d2 squarer
xs2 x1 x2 da dc s1 d2 squarer
xs3 x2 0 dc da s1 d2 squarer
.subckt squarer x1 x2 dc da s1 d2
xp1 fg1 x1 y1 d1 s1 vpmos
xp2 fg2 x2 d2 d2 d1 vpmos
xp3 fg3 x2 y1 d3 s1 vpmos
xp4 fg4 x1 d2 d2 d3 vpmos
.subckt vpmos fg x y d s
mpmos d fg s s modp w=20u l=5u
g1 0 1 x 0 1
r1 0 1 0.1
g2 0 2 y 0 1
r2 0 2 0.8
gs 0 3 s 0 1
rs 0 3 0.099
gd 0 4 d 0 1
rd 0 4 1e-3
gsub 0 5 sub 0 1
rsub 0 5 0.001
vsub sub 0 dc -1.5
e fg 0 poly(5) 1 0 2 0 3 0 4 0 5 0 (-0.063) 1 1 1 1 1
rx x 1 1e10
ry y 2 1e10
rsu sub 5 1e10
.model modp pmos level=3
+js=0.0062 cj=0.000378 cjsw=2.2e-10 tox=250e-10 ld=0.06e-6 wd=0.6e-6 uo=230
+vto=-0.63 theta=0.17 rs=120 rd=120 delta=0.4 xj=0.3e-6 vmax=0
+eta=0.01 kappa=0.07 nfs=4e11 gamma=0.50 phi=0.35
+jsw=2.2e-9 tpg=-1
.ends vpmos
vb y1 0 dc 0.1875
ma da d1 sa sa modn w=33u l=5u
mc dc 0 sa sa modn w=66u l=5u
mb da d3 sa sa modn w=33u l=5u
m1 sa g1 d2 d2 modn w=150u l=1.5u
m2 g1 g1 d2 d2 modn w=150u l=1.5u
m3 g1 g1 s1 s1 modp w=17.5u l=2u

```

```

.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends squarer
mo1 da da s1 s1 modp w=22u l=1.5u
mo2 dc dc s1 s1 modp w=22u l=1.5u
.ends multiplier
*****farksal FG-comperator yapisi*****
.subckt fgcomp dc1 dc2 dc3 da1 da2 da3 out s1 d2
xn1 fg1 dc1 dc2 dc3 g3 d5 vnmos
xn2 fg2 da1 da2 da3 out d5 vnmos
m3 g3 g3 s1 s1 modp w=8u l=5u
m4 out g3 s1 s1 modp w=8u l=5u
m5 d5 bias d2 d2 modn w=10u l=5u
vbias bias 0 dc 1.5
.subckt vnmos fg x y z d s
mnmos d fg s s modn w=5u l=5u
g1 0 1 x 0 1
r1 0 1 0.3
g2 0 2 y 0 1
r2 0 2 0.3
g3 0 3 z 0 1
r3 0 3 0.3
gs 0 4 s 0 1
rs 0 4 0.098
gd 0 5 d 0 1
rd 0 5 1e-3
gsub 0 6 sub 0 1
rsub 0 6 0.001
vsub sub 0 dc 1.5
e fg 0 poly(6) 1 0 2 0 3 0 4 0 5 0 6 0 (0.068) 1 1 1 1 1 1
rx x 1 1e10
ry y 2 1e10
rz z 3 1e10
rsu sub 6 1e10
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends vnmos
.ends fgcomp
*****finish farksal FG-comp*****
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends neuron3in

```

```

v1 x1 0 pwl (time, px1)
v2 x2 0 pwl (time, px2)
v3 x3 0 dc 1
.data inputs
time px1 px2
0u -1.5 -1.5
1u -1.5 -1.5
2u -1.5 1.5
3u 1.5 -1.5
4u 1.5 1.5
5u -1.5 -1.5
6u -1.5 1.5
7u 1.5 -1.5
8u 1.5 1.5
.enddata
v4 w11 0 dc w11
v5 w12 0 dc w12
v6 w13 0 dc w13
v7 w21 0 dc w21
v8 w22 0 dc w22
v9 w23 0 dc w23
v10 w31 0 dc w31
v11 w32 0 dc w32
v12 w3b 0 dc w3b
v13 b2 0 dc 1
.data weights
w11 w12 w21 w22 w13 w23      w31 w32 w3b      $[ iw b1 lw b2]
    1.2583  1.4999
+ -0.6270 -0.6771
+ -1.0940
+ -0.6923
+ -1.5102 -1.5121
+ -1.4564
.enddata
vdd s1 0 dc 1.5
vss d2 0 dc -1.5
.tran data=inputs
.end

```

Ek 3 Iris' e ait orijinal data dosyası

5.1,3.5,1.4,0.2,Iris-setosa
 4.9,3.0,1.4,0.2,Iris-setosa
 4.7,3.2,1.3,0.2,Iris-setosa
 4.6,3.1,1.5,0.2,Iris-setosa
 5.0,3.6,1.4,0.2,Iris-setosa
 5.4,3.9,1.7,0.4,Iris-setosa
 4.6,3.4,1.4,0.3,Iris-setosa
 5.0,3.4,1.5,0.2,Iris-setosa
 4.4,2.9,1.4,0.2,Iris-setosa
 4.9,3.1,1.5,0.1,Iris-setosa
 5.4,3.7,1.5,0.2,Iris-setosa
 4.8,3.4,1.6,0.2,Iris-setosa
 4.8,3.0,1.4,0.1,Iris-setosa
 4.3,3.0,1.1,0.1,Iris-setosa
 5.8,4.0,1.2,0.2,Iris-setosa
 5.7,4.4,1.5,0.4,Iris-setosa
 5.4,3.9,1.3,0.4,Iris-setosa
 5.1,3.5,1.4,0.3,Iris-setosa
 5.7,3.8,1.7,0.3,Iris-setosa
 5.1,3.8,1.5,0.3,Iris-setosa
 5.4,3.4,1.7,0.2,Iris-setosa
 5.1,3.7,1.5,0.4,Iris-setosa
 4.6,3.6,1.0,0.2,Iris-setosa
 5.1,3.3,1.7,0.5,Iris-setosa
 4.8,3.4,1.9,0.2,Iris-setosa
 5.0,3.0,1.6,0.2,Iris-setosa
 5.0,3.4,1.6,0.4,Iris-setosa
 5.2,3.5,1.5,0.2,Iris-setosa
 5.2,3.4,1.4,0.2,Iris-setosa
 4.7,3.2,1.6,0.2,Iris-setosa
 4.8,3.1,1.6,0.2,Iris-setosa
 5.4,3.4,1.5,0.4,Iris-setosa
 5.2,4.1,1.5,0.1,Iris-setosa
 5.5,4.2,1.4,0.2,Iris-setosa
 4.9,3.1,1.5,0.2,Iris-setosa
 5.0,3.2,1.2,0.2,Iris-setosa
 5.5,3.5,1.3,0.2,Iris-setosa
 4.9,3.6,1.4,0.1,Iris-setosa
 4.4,3.0,1.3,0.2,Iris-setosa
 5.1,3.4,1.5,0.2,Iris-setosa
 5.0,3.5,1.3,0.3,Iris-setosa
 4.5,2.3,1.3,0.3,Iris-setosa
 4.4,3.2,1.3,0.2,Iris-setosa
 5.0,3.5,1.6,0.6,Iris-setosa
 5.1,3.8,1.9,0.4,Iris-setosa
 4.8,3.0,1.4,0.3,Iris-setosa
 5.1,3.8,1.6,0.2,Iris-setosa
 4.6,3.2,1.4,0.2,Iris-setosa
 5.3,3.7,1.5,0.2,Iris-setosa

5.0,3.3,1.4,0.2,Iris-setosa
 7.0,3.2,4.7,1.4,Iris-versicolor
 6.4,3.2,4.5,1.5,Iris-versicolor
 6.9,3.1,4.9,1.5,Iris-versicolor
 5.5,2.3,4.0,1.3,Iris-versicolor
 6.5,2.8,4.6,1.5,Iris-versicolor
 5.7,2.8,4.5,1.3,Iris-versicolor
 6.3,3.3,4.7,1.6,Iris-versicolor
 4.9,2.4,3.3,1.0,Iris-versicolor
 6.6,2.9,4.6,1.3,Iris-versicolor
 5.2,2.7,3.9,1.4,Iris-versicolor
 5.0,2.0,3.5,1.0,Iris-versicolor
 5.9,3.0,4.2,1.5,Iris-versicolor
 6.0,2.2,4.0,1.0,Iris-versicolor
 6.1,2.9,4.7,1.4,Iris-versicolor
 5.6,2.9,3.6,1.3,Iris-versicolor
 6.7,3.1,4.4,1.4,Iris-versicolor
 5.6,3.0,4.5,1.5,Iris-versicolor
 5.8,2.7,4.1,1.0,Iris-versicolor
 6.2,2.2,4.5,1.5,Iris-versicolor
 5.6,2.5,3.9,1.1,Iris-versicolor
 5.9,3.2,4.8,1.8,Iris-versicolor
 6.1,2.8,4.0,1.3,Iris-versicolor
 6.3,2.5,4.9,1.5,Iris-versicolor
 6.1,2.8,4.7,1.2,Iris-versicolor
 6.4,2.9,4.3,1.3,Iris-versicolor
 6.6,3.0,4.4,1.4,Iris-versicolor
 6.8,2.8,4.8,1.4,Iris-versicolor
 6.7,3.0,5.0,1.7,Iris-versicolor
 6.0,2.9,4.5,1.5,Iris-versicolor
 5.7,2.6,3.5,1.0,Iris-versicolor
 5.5,2.4,3.8,1.1,Iris-versicolor
 5.5,2.4,3.7,1.0,Iris-versicolor
 5.8,2.7,3.9,1.2,Iris-versicolor
 6.0,2.7,5.1,1.6,Iris-versicolor
 5.4,3.0,4.5,1.5,Iris-versicolor
 6.0,3.4,4.5,1.6,Iris-versicolor
 6.7,3.1,4.7,1.5,Iris-versicolor
 6.3,2.3,4.4,1.3,Iris-versicolor
 5.6,3.0,4.1,1.3,Iris-versicolor
 5.5,2.5,4.0,1.3,Iris-versicolor
 5.5,2.6,4.4,1.2,Iris-versicolor
 6.1,3.0,4.6,1.4,Iris-versicolor
 5.8,2.6,4.0,1.2,Iris-versicolor
 5.0,2.3,3.3,1.0,Iris-versicolor
 5.6,2.7,4.2,1.3,Iris-versicolor
 5.7,3.0,4.2,1.2,Iris-versicolor
 5.7,2.9,4.2,1.3,Iris-versicolor
 6.2,2.9,4.3,1.3,Iris-versicolor
 5.1,2.5,3.0,1.1,Iris-versicolor
 5.7,2.8,4.1,1.3,Iris-versicolor

6.3,3.3,6.0,2.5,Iris-virginica
 5.8,2.7,5.1,1.9,Iris-virginica
 7.1,3.0,5.9,2.1,Iris-virginica
 6.3,2.9,5.6,1.8,Iris-virginica
 6.5,3.0,5.8,2.2,Iris-virginica
 7.6,3.0,6.6,2.1,Iris-virginica
 4.9,2.5,4.5,1.7,Iris-virginica
 7.3,2.9,6.3,1.8,Iris-virginica
 6.7,2.5,5.8,1.8,Iris-virginica
 7.2,3.6,6.1,2.5,Iris-virginica
 6.5,3.2,5.1,2.0,Iris-virginica
 6.4,2.7,5.3,1.9,Iris-virginica
 6.8,3.0,5.5,2.1,Iris-virginica
 5.7,2.5,5.0,2.0,Iris-virginica
 5.8,2.8,5.1,2.4,Iris-virginica
 6.4,3.2,5.3,2.3,Iris-virginica
 6.5,3.0,5.5,1.8,Iris-virginica
 7.7,3.8,6.7,2.2,Iris-virginica
 7.7,2.6,6.9,2.3,Iris-virginica
 6.0,2.2,5.0,1.5,Iris-virginica
 6.9,3.2,5.7,2.3,Iris-virginica
 5.6,2.8,4.9,2.0,Iris-virginica
 7.7,2.8,6.7,2.0,Iris-virginica
 6.3,2.7,4.9,1.8,Iris-virginica
 6.7,3.3,5.7,2.1,Iris-virginica
 7.2,3.2,6.0,1.8,Iris-virginica
 6.2,2.8,4.8,1.8,Iris-virginica
 6.1,3.0,4.9,1.8,Iris-virginica
 6.4,2.8,5.6,2.1,Iris-virginica
 7.2,3.0,5.8,1.6,Iris-virginica
 7.4,2.8,6.1,1.9,Iris-virginica
 7.9,3.8,6.4,2.0,Iris-virginica
 6.4,2.8,5.6,2.2,Iris-virginica
 6.3,2.8,5.1,1.5,Iris-virginica
 6.1,2.6,5.6,1.4,Iris-virginica
 7.7,3.0,6.1,2.3,Iris-virginica
 6.3,3.4,5.6,2.4,Iris-virginica
 6.4,3.1,5.5,1.8,Iris-virginica
 6.0,3.0,4.8,1.8,Iris-virginica
 6.9,3.1,5.4,2.1,Iris-virginica
 6.7,3.1,5.6,2.4,Iris-virginica
 6.9,3.1,5.1,2.3,Iris-virginica
 5.8,2.7,5.1,1.9,Iris-virginica
 6.8,3.2,5.9,2.3,Iris-virginica
 6.7,3.3,5.7,2.5,Iris-virginica
 6.7,3.0,5.2,2.3,Iris-virginica
 6.3,2.5,5.0,1.9,Iris-virginica
 6.5,3.0,5.2,2.0,Iris-virginica
 6.2,3.4,5.4,2.3,Iris-virginica
 5.9,3.0,5.1,1.8,Iris-virginica

Ek 4 Iris'in simülasyonunda kullanılan normalize edilmiş girişlere ait data dosyası

time	px1	px2	px3	px4
0	0	0	0	0
1u	-0.8333	0.375	-1.2966	-1.375
2u	-1	-0.25	-1.2966	-1.375
3u	-1.1667	0	-1.3475	-1.375
4u	-1.25	-0.125	-1.2458	-1.375
5u	-0.9167	0.5	-1.2966	-1.375
6u	-0.5833	0.875	-1.1441	-1.125
7u	-1.25	0.25	-1.2966	-1.25
8u	-0.9167	0.25	-1.2458	-1.375
9u	-1.4167	-0.375	-1.2966	-1.375
10u	-1	-0.125	-1.2458	-1.5
11u	-0.5833	0.625	-1.2458	-1.375
12u	-1.0833	0.25	-1.1949	-1.375
13u	-1.0833	-0.25	-1.2966	-1.5
14u	-1.5	-0.25	-1.4492	-1.5
15u	-0.25	1	-1.3983	-1.375
16u	-0.3333	1.5	-1.2458	-1.125
17u	-0.5833	0.875	-1.3475	-1.125
18u	-0.8333	0.375	-1.2966	-1.25
19u	-0.3333	0.75	-1.1441	-1.25
20u	-0.8333	0.75	-1.2458	-1.25
21u	-0.5833	0.25	-1.1441	-1.375
22u	-0.8333	0.625	-1.2458	-1.125
23u	-1.25	0.5	-1.5	-1.375
24u	-0.8333	0.125	-1.1441	-1
25u	-1.0833	0.25	-1.0424	-1.375
26u	-0.9167	-0.25	-1.1949	-1.375
27u	-0.9167	0.25	-1.1949	-1.125
28u	-0.75	0.375	-1.2458	-1.375
29u	-0.75	0.25	-1.2966	-1.375
30u	-1.1667	0	-1.1949	-1.375
31u	-1.0833	-0.125	-1.1949	-1.375
32u	-0.5833	0.25	-1.2458	-1.125
33u	-0.75	1.125	-1.2458	-1.5
34u	-0.5	1.25	-1.2966	-1.375
35u	-1	-0.125	-1.2458	-1.5
36u	-0.9167	0	-1.3983	-1.375
37u	-0.5	0.375	-1.3475	-1.375
38u	-1	-0.125	-1.2458	-1.5
39u	-1.4167	-0.25	-1.3475	-1.375
40u	-0.8333	0.25	-1.2458	-1.375
41u	-0.9167	0.375	-1.3475	-1.25
42u	-1.3333	-1.125	-1.3475	-1.25
43u	-1.4167	0	-1.3475	-1.375
44u	-0.9167	0.375	-1.1949	-0.875
45u	-0.8333	0.75	-1.0424	-1.125
46u	-1.0833	-0.25	-1.2966	-1.25
47u	-0.8333	0.75	-1.1949	-1.375
48u	-1.25	0	-1.2966	-1.375

49u	-0.6667	, 0.625	, -1.2458,	-1.375
50u	-0.9167	, 0.125	, -1.2966,	-1.375
51u	0.75	, 0	, 0.3814,	0.125
52u	0.25	, 0	, 0.2797,	0.25
53u	0.6667	, -0.125	, 0.4831,	0.25
54u	-0.5	, -1.125	, 0.0254,	0
55u	0.3333	, -0.5	, 0.3305,	0.25
56u	-0.3333	, -0.5	, 0.2797,	0
57u	0.1667	, 0.125	, 0.3814,	0.375
58u	-1	, -1	, -0.3305,	-0.375
59u	0.4167	, -0.375	, 0.3305,	0
60u	-0.75	, -0.625	, -0.0254,	0.125
61u	-0.9167	, -1.5	, -0.2288,	-0.375
62u	-0.1667	, -0.25	, 0.1271,	0.25
63u	-0.0833	, -1.25	, 0.0254,	-0.375
64u	0	, -0.375	, 0.3814,	0.125
65u	-0.4167	, -0.375	, -0.178,	0
66u	0.5	, -0.125	, 0.2288,	0.125
67u	-0.4167	, -0.25	, 0.2797,	0.25
68u	-0.25	, -0.625	, 0.0763,	-0.375
69u	0.0833	, -1.25	, 0.2797,	0.25
70u	-0.4167	, -0.875	, -0.0254,	-0.25
71u	-0.1667	, 0	, 0.4322,	0.625
72u	0	, -0.5	, 0.0254,	0
73u	0.1667	, -0.875	, 0.4831,	0.25
74u	0	, -0.5	, 0.3814,	-0.125
75u	0.25	, -0.375	, 0.178,	0
76u	0.4167	, -0.25	, 0.2288,	0.125
77u	0.5833	, -0.5	, 0.4322,	0.125
78u	0.5	, -0.25	, 0.5339,	0.5
79u	-0.0833	, -0.375	, 0.2797,	0.25
80u	-0.3333	, -0.75	, -0.2288,	-0.375
81u	-0.5	, -1	, -0.0763,	-0.25
82u	-0.5	, -1	, -0.1271,	-0.375
83u	-0.25	, -0.625	, -0.0254,	-0.125
84u	-0.0833	, -0.625	, 0.5847,	0.375
85u	-0.5833	, -0.25	, 0.2797,	0.25
86u	-0.0833	, 0.25	, 0.2797,	0.375
87u	0.5	, -0.125	, 0.3814,	0.25
88u	0.1667	, -1.125	, 0.2288,	0
89u	-0.4167	, -0.25	, 0.0763,	0
90u	-0.5	, -0.875	, 0.0254,	0
91u	-0.5	, -0.75	, 0.2288,	-0.125
92u	0	, -0.25	, 0.3305,	0.125
93u	-0.25	, -0.75	, 0.0254,	-0.125
94u	-0.9167	, -1.125	, -0.3305,	-0.375
95u	-0.4167	, -0.625	, 0.1271,	0
96u	-0.3333	, -0.25	, 0.1271,	-0.125
97u	-0.3333	, -0.375	, 0.1271,	0
98u	0.0833	, -0.375	, 0.178,	0
99u	-0.8333	, -0.875	, -0.4831,	-0.25

100u	-0.3333 ,	-0.5 ,	0.0763 ,	0
101u	0.1667 ,	0.125 ,	1.0424 ,	1.5
102u	-0.25 ,	-0.625 ,	0.5847 ,	0.75
103u	0.8333 ,	-0.25 ,	0.9915 ,	1
104u	0.1667 ,	-0.375 ,	0.839 ,	0.625
105u	0.3333 ,	-0.25 ,	0.9407 ,	1.125
106u	1.25 ,	-0.25 ,	1.3475 ,	1
107u	-1 ,	-0.875 ,	0.2797 ,	0.5
108u	1 ,	-0.375 ,	1.1949 ,	0.625
109u	0.5 ,	-0.875 ,	0.9407 ,	0.625
110u	0.9167 ,	0.5 ,	1.0932 ,	1.5
111u	0.3333 ,	0 ,	0.5847 ,	0.875
112u	0.25 ,	-0.625 ,	0.6864 ,	0.75
113u	0.5833 ,	-0.25 ,	0.7881 ,	1
114u	-0.3333 ,	-0.875 ,	0.5339 ,	0.875
115u	-0.25 ,	-0.5 ,	0.5847 ,	1.375
116u	0.25 ,	0 ,	0.6864 ,	1.25
117u	0.3333 ,	-0.25 ,	0.7881 ,	0.625
118u	1.3333 ,	0.75 ,	1.3983 ,	1.125
119u	1.3333 ,	-0.75 ,	1.5 ,	1.25
120u	-0.0833 ,	-1.25 ,	0.5339 ,	0.25
121u	0.6667 ,	0 ,	0.8898 ,	1.25
122u	-0.4167 ,	-0.5 ,	0.4831 ,	0.875
123u	1.3333 ,	-0.5 ,	1.3983 ,	0.875
124u	0.1667 ,	-0.625 ,	0.4831 ,	0.625
125u	0.5 ,	0.125 ,	0.8898 ,	1
126u	0.9167 ,	0 ,	1.0424 ,	0.625
127u	0.0833 ,	-0.5 ,	0.4322 ,	0.625
128u	0 ,	-0.25 ,	0.4831 ,	0.625
129u	0.25 ,	-0.5 ,	0.839 ,	1
130u	0.9167 ,	-0.25 ,	0.9407 ,	0.375
131u	1.0833 ,	-0.5 ,	1.0932 ,	0.75
132u	1.5 ,	0.75 ,	1.2458 ,	0.875
133u	0.25 ,	-0.5 ,	0.839 ,	1.125
134u	0.1667 ,	-0.5 ,	0.5847 ,	0.25
135u	0 ,	-0.75 ,	0.839 ,	0.125
136u	1.3333 ,	-0.25 ,	1.0932 ,	1.25
137u	0.1667 ,	0.25 ,	0.839 ,	1.375
138u	0.25 ,	-0.125 ,	0.7881 ,	0.625
139u	-0.0833 ,	-0.25 ,	0.4322 ,	0.625
140u	0.6667 ,	-0.125 ,	0.7373 ,	1
141u	0.5 ,	-0.125 ,	0.839 ,	1.375
142u	0.6667 ,	-0.125 ,	0.5847 ,	1.25
143u	-0.25 ,	-0.625 ,	0.5847 ,	0.75
144u	0.5833 ,	0 ,	0.9915 ,	1.25
145u	0.5 ,	0.125 ,	0.8898 ,	1.5
146u	0.5 ,	-0.25 ,	0.6356 ,	1.25
147u	0.1667 ,	-0.875 ,	0.5339 ,	0.75
148u	0.3333 ,	-0.25 ,	0.6356 ,	0.875
149u	0.0833 ,	0.25 ,	0.7373 ,	1.25
150u	-0.1667 ,	-0.25 ,	0.5847 ,	0.625

Ek 5 Iris'in simülasyonunda kullanılan HSPICE giriş dosyası

```

*iris
.option post
x5in1 x1 w11 x2 w12 x3 w13 x4 w14 x5 w15 out1 s1 d2 neuron5in
x5in2 x1 w21 x2 w22 x3 w23 x4 w24 x5 w25 out2 s1 d2 neuron5in
x5in3 x1 w31 x2 w32 x3 w33 x4 w34 x5 w35 out3 s1 d2 neuron5in
x5in4 x1 w41 x2 w42 x3 w43 x4 w44 x5 w45 out4 s1 d2 neuron5in
x5in5 out1 w51 out2 w52 out3 w53 out4 w54 b2 w5b outa s1 d2 neuron5in
x5in6 out1 w61 out2 w62 out3 w63 out4 w64 b2 w6b outb s1 d2 neuron5in
x5in7 out1 w71 out2 w72 out3 w73 out4 w74 b2 w7b outc s1 d2 neuron5in
.subckt neuron5in x1 y1 x2 y2 x3 y3 x4 y4 x5 y5 out s1 d2
xc1 dc1 dc2 dc3 dc4 dc5 da1 da2 da3 da4 da5 out s1 d2 fgcomp
xm1 x1 y1 dc1 da1 s1 d2 multiplier
xm2 x2 y2 dc2 da2 s1 d2 multiplier
xm3 x3 y3 dc3 da3 s1 d2 multiplier
xm4 x4 y4 dc4 da4 s1 d2 multiplier
xm5 x5 y5 dc5 da5 s1 d2 multiplier
.subckt multiplier x2 x1 dc da s1 d2
xs1 0 x1 dc da s1 d2 squarer
xs2 x1 x2 da dc s1 d2 squarer
xs3 x2 0 dc da s1 d2 squarer
.subckt squarer x1 x2 dc da s1 d2
xp1 fg1 x1 y1 d1 s1 vpmos
xp2 fg2 x2 d2 d2 d1 vpmos
xp3 fg3 x2 y1 d3 s1 vpmos
xp4 fg4 x1 d2 d2 d3 vpmos
.subckt vpmos fg x y d s
mpmos d fg s s modp w=20u l=5u
g1 0 1 x 0 1
r1 0 1 0.1
g2 0 2 y 0 1
r2 0 2 0.8
gs 0 3 s 0 1
rs 0 3 0.099
gd 0 4 d 0 1
rd 0 4 1e-3
gsub 0 5 sub 0 1
rsub 0 5 0.001
vsub sub 0 dc -1.5
e fg 0 poly(5) 1 0 2 0 3 0 4 0 5 0 (-0.063) 1 1 1 1 1
rx x 1 1e10
ry y 2 1e10
rsu sub 5 1e10
.model modp pmos level=3
+js=0.0062 cj=0.000378 cjsw=2.2e-10 tox=250e-10 ld=0.06e-6 wd=0.6e-6 uo=230
+vto=-0.63 theta=0.17 rs=120 rd=120 delta=0.4 xj=0.3e-6 vmax=0
+eta=0.01 kappa=0.07 nfs=4e11 gamma=0.50 phi=0.35
+tjsw=2.2e-9 tpg=-1
.ends vpmos
vb y1 0 dc 0.1875

```

```

ma da d1 sa sa modn w=33u l=5u
mc dc 0 sa sa modn w=66u l=5u
mb da d3 sa sa modn w=33u l=5u
m1 sa g1 d2 d2 modn w=150u l=1.5u
m2 g1 g1 d2 d2 modn w=150u l=1.5u
m3 g1 g1 s1 s1 modp w=17.5u l=2u
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends squarer
mo1 da da s1 s1 modp w=22u l=1.5u
mo2 dc dc s1 s1 modp w=22u l=1.5u
.ends multiplier
*****farksal FG-comperator yapisi*****
.subckt fgcomp dc1 dc2 dc3 dc4 dc5 da1 da2 da3 da4 da5 out s1 d2
xn1 fg1 dc1 dc2 dc3 dc4 dc5 g3 d5 vnmos
xn2 fg2 da1 da2 da3 da4 da5 out d5 vnmos
m3 g3 g3 s1 s1 modp w=13u l=8u
m4 out g3 s1 s1 modp w=13u l=8u
m5 d5 bias d2 d2 modn w=16u l=8u
vbias bias 0 dc 1.5
.subckt vnmos fg x y z t u d s
mnmos d fg s s modn w=8u l=8u
g1 0 1 x 0 1
r1 0 1 0.18
g2 0 2 y 0 1
r2 0 2 0.18
g3 0 3 z 0 1
r3 0 3 0.18
g4 0 4 t 0 1
r4 0 4 0.18
g5 0 5 u 0 1
r5 0 5 0.18
gs 0 6 s 0 1
rs 0 6 0.099
gd 0 7 d 0 1
rd 0 7 1e-3
gsub 0 8 sub 0 1
rsub 0 8 0.001
vsub sub 0 dc 1.5
e fg 0 poly(8) 1 0 2 0 3 0 4 0 5 0 6 0 7 0 8 0 (0.68) 1 1 1 1 1 1 1 1
rx x 1 1e10
ry y 2 1e10
rz z 3 1e10
rt t 4 1e10
ru u 5 1e10
rsu sub 8 1e10
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570

```

```

+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends vnm0s
.ends fgcomp
*****finish farksal FG-comp*****
.model modn nmos level=3
+js=0.0016 cj=0.00021 cjsw=2.6e-10 tox=250e-10 ld=12.5e-8 wd=0.6e-6 uo=570
+vto=0.68 theta=0.05 rs=75 rd=75 delta=0.4 xj=1.5e-7 vmax=230000
+eta=0.005 kappa=0.05 nfs=1.05641e10 gamma=0.50 phi=0.35
+jsw=4e-10 tpg=1
.ends neuron5in
****girisler+bias1*****
v1 x1 0 pwl (time, px1)
v2 x2 0 pwl (time, px2)
v3 x3 0 pwl (time, px3)
v4 x4 0 pwl (time, px4)
v5 x5 0 dc 1 $ bias1
.data inputs mer $lam
file='iris-inputs_norm.data' time=1 px1=2 px2=3 px3=4 px4=5
.enddata
****layer1 agirliliklari*****
v6 w11 0 w11
v7 w12 0 w12
v8 w13 0 w13
v9 w14 0 w14
v10 w15 0 w15
v11 w21 0 w21
v12 w22 0 w22
v13 w23 0 w23
v14 w24 0 w24
v15 w25 0 w25
v16 w31 0 w31
v17 w32 0 w32
v18 w33 0 w33
v19 w34 0 w34
v20 w35 0 w35
v21 w41 0 w41
v22 w42 0 w42
v23 w43 0 w43
v24 w44 0 w44
v25 w45 0 w45
*****bias2+layer2 agirliliklari*****
v26 w51 0 dc w51
v27 w52 0 dc w52
v28 w53 0 dc w53
v29 w54 0 dc w54
v30 w5b 0 dc w5b
v31 w61 0 dc w61
v32 w62 0 dc w62
v33 w63 0 dc w63

```

```

v34 w64 0 dc w64
v35 w6b 0 dc w6b
v36 w71 0 dc w71
v37 w72 0 dc w72
v38 w73 0 dc w73
v39 w74 0 dc w74
v40 w7b 0 dc w7b
v41 b2 0 dc 1

```

```

*****

```

```

.data weights

```

```

w11 w12 w13 w14 w21 w22 w23 w24 w31 w32 w33 w34 w41 w42 w43 w44 w15 w25 w35 w45
+ w51 w52 w53 w54 w61 w62 w63 w64 w71 w72 w73 w74 w5b w6b w7b
    0.5552  0.1691  0.3530  0.2165
+  0.4368 -0.9130  0.3979  0.8360
+ -0.0901  0.0303 -0.2831  0.4837
+  0.3865  0.5756 -1.5000 -1.3357
+ -1.1137
+  0.8029
+  0.6660
+  1.5000
+  0.0377 -1.2939 -0.0729 -0.2214
+ -0.1700  1.5000 -0.9759  1.5000
+  0.8414  0.3870  0.3114 -1.5000
+  0.1047
+ -0.9375
+  0.0773

```

```

.enddata

```

```

vdd s1 0 dc 1.5

```

```

vss d2 0 dc -1.5

```

```

.tran data=inputs

```

```

.end

```

ÖZGEÇMİŞ

Doğum tarihi 15.09.1978

Doğum yeri İstanbul

Lise 1993-1996 Ataköy Lisesi

Lisans 1996-2000 Yıldız Teknik Üniversitesi Elektrik-Elektronik Fak.
Elektronik ve Haberleşme Mühendisliği Bölümü

Yüksek Lisans 2000-2003 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektronik ve Haberleşme. Müh. Anabilim Dalı
Elektronik Programı

Çalıştığı kurum

2001-Devam ediyor YTÜ Fen Bilimleri Enstitüsü Araştırma Görevlisi

