ASD: Çok Amaçlı Ayarlanabilir Sınıflandırıcı Devreler

Proje No: 106E139

Prof. Dr. Cem GÖKNAR Prof. Dr. Shahram MINAEI Dr. Merih YILDIZ Dr. Engin DENİZ

> EYLÜL 2010 İSTANBUL

ÖNSÖZ

Bu projenin ilk aşamasında mevcut sınıflandırıcı yapıları, üstünlük ve sakıncaları, uygulama alanları incelenmiştir. Mevcut sakıncaları gideren, yeni imkanlar tanıyacak şekilde tümleştirmeye uygun sınıflandırıcı yapısı tasarlanmıştır. Ayrıca bu yapıya uygun sınıflandırıcı algoritmaları geliştirilmiş ve simulasyonları yapılmıştır.

Projenin ikinci kısmında ise tasarlanmış olan sınıflandırıcı devresinin geliştirilen serimi yazılımla sınanmış sonuçta elde edilen serimin tümdevre üretimi yaptırılmıştır. Önce üretilen tümdevrenin sınanması sonra da simulasyonlar ile yapılan uygulamaların tümdevre ile testleri gerçekleştirilerek yazılımsal sonuçların ölçüm sonuçlarıyla uyum içinde oldukları gösterilmiştir.

Proje TÜBİTAK Elektrik, Elektronik ve Enformatik Araştırma Grubu (EEEAG) tarafından 106E139 numaralı kontrat altında desteklenmiştir.

Her aşamasında önemli desteğini ve katkısını gördüğümüz EEEAG'na, TÜBİTAK personeline çok teşekkür ederiz.

İÇİNDEKİLER

| ÖNGÖZ | ••• |
|---|---------|
| | ,.111 |
| IÇINDEKILER | V |
| ŞEKIL LIŞTEŞI Özet | V |
| | vm · |
| | IX |
| I. GIRIŞ | I |
| 1.1 Literatur Taramasi | 2 |
| 1.1.1 Yazılımsal yöntemler | 2 |
| 1.1.2 Donanimsal yapılar. | 4 |
| 1.2 Projede Kullanılan Yöntem ve Amaç | 8 |
| 1.3 Çalışmada Yapılanlar | 9 |
| 2. SINIFLANDIRICI DEVRESI TASARIMI VE BENZETIMLERI | . 11 |
| 2.1 Sınıflandırıcı Yapısı Blok Diyagramı | . 11 |
| 2.2 Çekirdek Devreler | . 14 |
| 2.2.1 Çekirdek devre-1 | . 14 |
| 2.2.2 Çekirdek devre-2 | . 22 |
| 3. SINIFLANDIRICI DEVRE UYGULAMALARI (KUANTALAYICI VE KARAKTE | R |
| TANIMA) | . 33 |
| 3.1 Kuantalayıcı | . 33 |
| 3.2 Karakter Tanıma | . 34 |
| 4. EĞİK IZGARALI SINIFLANDIRICILAR | . 40 |
| 4.1 Bölgelerin Oluşturulması | . 40 |
| 4.2 ÇAD Devresi ve Simülasyonları | . 45 |
| 5. PROTOTIP VE BASKI DEVRE SERİMLERİ; DU-TCC1209 LAB TESTİ | . 51 |
| 5.1 Çekirdek devrenin serimi | . 51 |
| 5.2 Akım taşıyıcı devresinin Serimi | . 52 |
| 5.3 Prototip devrenin serimi (DU-TCC 1209) | . 56 |
| 5.4 Prototip devrenin eğik ızgaralı veri sınıflandırmada testi | . 60 |
| 6. SINIFLANDIRMA ALGORİTMALARI VE DEVRELERE UYGULANMASI | . 62 |
| 6.1 Fisher Tabanlı Algoritma ile Çift Eşik Doğrularının Bulunmalarına Genel Bakış | . 62 |
| 6.1.1 Fisher tabanlı algoritma ile çift eşik doğrularının bulunması | . 63 |
| 6.1.2 Fisher tabanlı algoritma ile çift eşik doğrularının genel halde bulunması | . 68 |
| 6.2 Eğiticili Perseptron Öğrenme Algoritması ile ÇAD ve CD Parametrelerinin Bulunması | 70 |
| 7. SINIFLANDIRICI DEVRE UYGULAMALARI | .74 |
| 7.1 İris ve Haberman Verilerinin Sınıflandırılması | . 74 |
| 7.1.1 İris verisinin Fisher tabanlı algoritma ile sınıflandırılması | . 74 |
| 7.1.2 Haberman verisinin Fisher tabanlı algoritma ile sınıflandırılması | . 80 |
| 7.1.3 İris verisinin perseptron öğrenme algoritması ile sınıflandırılması | . 85 |
| 7.1.4 Haberman verisinin perseptron öğrenme algoritması ile sınıflandırılması | . 89 |
| 8. SONUCLAR VE ÖNERİLER. | . 94 |
| REFERANSLAR | . 96 |
| EKLER | 101 |

TABLO LÍSTESÍ

| Tablo 2.1 : Çekirdek Yapı Değişkenleri | 14 |
|---|----|
| Tablo 2.2 : Çekirdek devre-1 yapısının MOS tranzistor boyutları. | 18 |
| Tablo 2.3: 1×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları | 20 |
| Tablo 2.4: 2×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları | 21 |
| Tablo 2.5 : CD-2'de kullanılan MOS tranzistor boyutları. | 27 |
| Tablo 2.6 : ÇD-2 ile gerçeklenen 1×1-D kontrol akımları (µA olarak) | 30 |
| Tablo 2.7 : CD-2 ile gerçeklenen 2×1-D sınıflandırıcı devresi kontrol akımları (akımlar µ | ιA |
| olarak ifade edilmiştir). | 30 |
| Tablo 2.8 : Kontrol akımlarını sağlayan direnç değerleri. | 31 |
| Tablo 2.9 : Sınıflandırıcı devrelerin karşılaştırılması. | 32 |
| Tablo 3.1: Kuantalayıcı Yapısında Kullanılan Çekirdek Devre Parametreleri | 34 |
| Tablo 3.2 : Farklı örüntüler için y _i (i=1,,5) çıkış değerleri | 36 |
| Tablo 4.1 : ÇAD devresi MOS tranzistorların boyutları | 50 |
| Tablo 5.1 : Çekirdek devre kontrol akımları | 61 |
| Tablo 7.1 : n çıkışlı akım çoğullayıcı devresi MOS tranzistorların boyutları | 76 |
| Tablo 7.2 : Şekil 7.2'deki iris verisi sınıflandırıcısı test kümesi | 77 |
| Tablo 7.3: İris sınıflandırıcısı ÇAD direnç değerleri. | 78 |
| Tablo 7.4: İris sınıflandırıcısı çekirdek devre kontrol akımları. | 78 |
| Tablo 7.5: İris sınıflandırıcısı ÇAD eşdeğer direnç değerleri. | 79 |
| Tablo 7.6: Şekil 7.8'deki Haberman verisi sınıflandırıcısı test kümesi | 83 |
| Tablo 7.7: Haberman sınıflandırıcısı ÇAD direnç değerleri | 83 |
| Tablo 7.8: Haberman sınıflandırıcısı çekirdek devre kontrol akımları | 83 |
| Tablo 7.9: Haberman sınıflandırıcısı ÇAD eşdeğer direnç değerleri | 84 |
| Tablo 7.10: Şekil 7.11'deki iris verisi sınıflandırıcısı test kümesi | 86 |
| Tablo 7.11: Şekil 7.11'de c1 sınıfı için ÇAD direnç değerleri ve ÇD akımları. | 88 |
| Tablo 7.12: Şekil 7.11'de c2 ve c3 sınıfı için ÇAD direnç değerleri ve ÇD akımları | 88 |
| Tablo 7.13: Sınıflandırıcı yapısında kullanılan ÇAD eşdeğer direnç değerleri | 89 |
| Tablo 7.14: Şekil 7.14'deki haberman verisi sınıflandırıcısı test kümesi | 91 |
| Tablo 7.15: Şekil 7.14'de c1 sınıfı için ÇAD direnç değerleri ve ÇD akımları. | 92 |
| Tablo 7.16: Sınıflandırıcı yapısında kullanılan ÇAD eşdeğer direnç değerleri | 92 |
| Tablo 7.17: Sınıflandırıcı başarımları karşılaştırması | 93 |

ŞEKİL LİSTESİ

<u>Sayfa</u>

| Şekil 1.1 : Donanımsal gerçeklenmiş sınıflandırıcı bloğu (AKSIN, 2005). | 5 |
|--|----|
| Şekil 2.1 : Çekirdek yapı geçiş karakteristiği | 11 |
| Şekil 2.2 : Çekirdek yapının blok diyagramı | 11 |
| Şekil 2.3 : n×1 boyutlu sınıflandırıcı devrenin blok diyagramı. | 12 |
| Şekil 2.4 : Tek boyutlu sınıflandırıcı ile elde edilmek istenilen giriş-çıkış karakteristiği | 13 |
| Şekil 2.5 : 2×1 boyutlu çok seviyeli sınıflandırıcı yapısı blok şeması | 13 |
| Şekil 2.6 : MATLAB programı ile elde edilen 2×1 boyutlu çok seviyeli sınıflandırıcı | |
| devresinin (x_1-x_2) -y karakteristiği | 14 |
| Şekil 2.7 : Akım-modlu çekirdek devre-1 yapısının blok diyagramı | 15 |
| Şekil 2.8 : Akım-modlu çekirdek devre-1 yapısının geçiş karakteristiği | 15 |
| Şekil 2.9 : Çekirdek devre-1 iç yapısının işlevsel diyagramı (YILDIZ, 2007a). | 15 |
| Şekil 2.10 : Çekirdek devre-1'in giriş katı ve evirici. | 16 |
| Şekil 2.11 : NOR kapısı ve çıkış katı. | 17 |
| Şekil 2.12 : Çekirdek devre-1'in I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2} \neq 0$) | 18 |
| Şekil 2.13 : Çekirdek devre-1'in I _{out} akımının I _{in} akımı ile değişim karakteristiği (I _{H2} =0) | 18 |
| Şekil 2.14 : Çekirdek devre-1'in V _{out} geriliminin I _{in} akımı ile değişim karakteristiği | 19 |
| Şekil 2.15 : Çekirdek devre-1 ile gerçeklenen 1×1-D sınıflandırıcı blok diyagramı | 19 |
| Şekil 2.16 : ÇD-1 ile gerçeklenen 1-D sınıflandırıcının giriş-çıkış karakteristiği | 20 |
| Şekil 2.17 : ÇD-1 ile gerçeklenen 2×1-D sınıflandırıcı blok diyagramı. | 21 |
| Şekil 2.18 : ÇD-1 ile gerçeklenen 2×1-D sınıflandırıcı (I _{in1} -I _{in2})-I _{out} karakteristiği | 22 |
| Şekil 2.19 : Akım-modlu ÇD-2'nin blok yapısı. | 22 |
| Şekil 2.20 : Akım-modlu ÇD-2 blok diyagramı iç yapısı (YILDIZ, 2007b) | 23 |
| Şekil 2.21 : Eşik devresinin giriş-çıkış karakteristiği | 23 |
| Şekil 2.22 : ÇD-2 yapısının blok şeması. | 24 |
| Şekil 2.23 : Pozitif geribeslemeli eşik devresi devre şeması | 25 |
| Şekil 2.24 : Pozitif geribeslemeli eşik devresi osiloskop çıktısı. | 25 |
| Şekil 2.25 : Pozitif geribeslemeli eşik devresi simülasyon karakteristiği. | 25 |
| Şekil 2.26 : Eşik devresi devre şeması. | 26 |
| Şekil 2.27 : ÇD-2 sınıflandırıcı devresi giriş-çıkış karakteristiği | 26 |
| Şekil 2.28 : ÇD-2 devre şeması (YILDIZ, 2007b) | 27 |
| Şekil 2.29 : Eşik devresinin giriş-çıkış karakteristiği | 28 |
| Şekil 2.30 : ÇD-2 yapısı giriş-çıkış karakteristiği. | 28 |
| Şekil 2.31 : Tek boyutlu sınıflandırıcının giriş-çıkış karakteristiği. | 29 |
| Şekil 2.32 : ÇD-2 ile gerçeklenen 1×1-D sınıflandırıcı giriş-çıkış simülasyonu. | 29 |
| Şekil 2.33 : ÇD-2 yapısının testinde kullanılan akım kaynağı modeli | 30 |
| Şekil 2.34 : ÇD-2 giriş-çıkış kararteristiği osiloskop sonucu | 31 |
| Şekil 3.1 : Sekiz seviyeli kuantalayıcı devre blok yapısı. | 33 |
| Şekil 3.2 : Sekiz seviyeli kuantalayıcı devresi I _{in} -I _{out} karakteristiği | 33 |
| Şekil 3.3 : Örnek örüntüler. | 35 |

| Şekil 3.4 : Şablon hücre gösterimi. | 35 |
|--|----|
| Şekil 3.5 : Karakter tanıma sınıflandırıcısı blok diyagramı. | 36 |
| Şekil 3.6 : İlk beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları | 37 |
| Şekil 3.7 : Son beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları | 38 |
| Şekil 3.8 : Hatalı test şablonları. | 38 |
| Şekil 3.9 : Karakter tanıma sınıflandırıcısı hata düzeltme benzetim sonucu.10 | 39 |
| Sekil 4.1 : Lineer olarak sınıflandırılamayan veri kümesi. | 40 |
| Şekil 4.2 : Eğik Izgaralı veri sınıflandırıcısı. | 41 |
| Sekil 4.3 : Dik Izgara olmayan veri bölgeleri. | 41 |
| Şekil 4.4 : ÇAD yapısı blok diyagramı sembolik gösterimi. | 42 |
| Şekil 4.5 : Şekil 4.2'deki bloğun iç yapısı | 42 |
| Sekil 4.6 : Paralel bağlanmış sınıflandırıcı devresi. | 43 |
| Sekil 4.7 : Eğik ızgaralı veri sınıfları. | 43 |
| Sekil 4.8 : Eğik ızgaralı sınıflandırıcı çıkışının 3-D I_{out} -(x ₁ -x ₂) karakteristiği | 44 |
| Sekil 4.9 : Verilerin CAD ve CD kullanılarak sınıflandırılması. | 44 |
| Sekil 4.10 : CAD devresi blok seması | 45 |
| Sekil 4.11 : DO-CCII blok diyagramı. | 46 |
| Sekil 4.12 : DO-CCII devre seması | 47 |
| Sekil 4.13 : DO-CCII devresi icin $V_x - V_y$ karakteristiği | 47 |
| Sekil 4.14 : DO-CCII icin I_{7+} ve I_{7-} akımlarının I_x ile değisim karakteristiği. | 48 |
| Sekil 4.15 : CAD yapısının DO-CCII yapıları ile gerçeklemesi. | 49 |
| Sekil 4.16 : CAD devresi icin I_{7+} ve I_{7-} akımlarının V_y gerilimi ile değisim karakteristiği | |
| $(R_2/R_3 = 5)$ | 49 |
| Şekil 4.17 : Çeşitli k= R_2 / R_3 değerleri için ÇAD devresi V _v - I _{z+} karakteristiği | 50 |
| Sekil 5.1. CD-2'nin serim cizimi | 52 |
| Sekil 5.2. Akım tasıvıcı sematiği | 53 |
| Sekil 5.3 · Akım taşıyıcı şerimi | 54 |
| Sekil 5.4 · Tampon devresi sematiği | 55 |
| Sekil 5.5 : Tampon devre serimi | |
| Sekil 5.6 : DU-TCC1209 Bağlantı Divagramı (TOFP 52 paketi icin). | |
| Sekil 5.7 : Entegre devre'in üretici firmava gönderilen son hali. | 57 |
| Sekil 5.8 : Üretici firmadan gelen DU-TCC1209 cip fotoğrafi (Die Photo) | 57 |
| Sekil 5.9 : Üretilmis olan entegre devrenin resmi | 58 |
| Sekil 5.10 : Protop devrenin testi icin kullanılan baskı devre. | |
| Sekil 5.11 : Baskı devrenin gerceklestirilmis sekli | |
| Sekil 5.12 : Giris Cıkıs karakteristiği test sonucu: osiloskop cıktısı. | 59 |
| Sekil 5.13 : Eğik ızgaralı veri sınıflandırıcısı test seması. | 60 |
| Sekil 5.14 : Eğik ızgaralı veri sınıflandırıcısı cıkış akım sekli osiloskop cıktısı. | |
| Sekil 6.1 : Lineer olarak sınıflandırılamayan örnek veri kümesi | |
| Sekil 6.2 · Histogram karakteristiği | 63 |
| Sekil 6.3 : Cift esik doğrularının gösterilimi | 67 |
| Sekil 6.4 : n girisli tek katlı perseptron yapısı | |
| Sekil 7.1 : İzdüsürülmüs iris verilerinin orijine uzaklıkları | |
| Sekil 7.2 · İris verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma ile oluşturulmuş) | 75 |
| Sekil 7.3 · Akım coğullavıcı devre seması (FERRI 2001) | 76 |
| Sekil 7.4 : Sekil 7.2'deki İris verisi sınıflandırıcısı simülasvon sonucu | 79 |
| Sekil 7.5 : İris verisi test sonucu osiloskop cıktısı | 80 |
| Sekil 7.6 : İris verisi test düzeneği. | 80 |
| Sekil 7.7 : İzdüsürülmüs Haberman verilerinin oriiine uzaklıkları | 81 |
| , , , , - J | |

| Şekil 7.8 : Haberman verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu | |
|--|----|
| oluşturulmuştur) | 82 |
| Şekil 7.9 : Şekil 7.8'deki haberman verisi sınıflandırıcısı simülasyon sonucu | 84 |
| Şekil 7.10 : DU-TCC 1209 ile Haberman verisi için test sonucu osiloskop çıktısı | 85 |
| Şekil 7.11 : İris verisi sınıflandırıcı bloğu (öğrenme algoritması sonucu oluşturulmuş) | 87 |
| Şekil 7.12 : Şekil 7.11'deki iris verisi sınıflandırıcısı simülasyon sonucu. | 88 |
| Şekil 7.13 : DU-TCC 1209 ile İris verisi test sonucu osiloskop çıktısı. | 89 |
| Şekil 7.14 : Haberman verisi sınıflandırıcı bloğu (perseptron öğrenme algoritması sonucu | |
| oluşturulmuş) | 90 |
| Şekil 7.15 : Şekil 7.14'daki Haberman verisi sınıflandırıcısı simülasyon sonucu | 92 |
| Şekil 7.16 : DU-TCC 1209 ile Haberman verisi test sonucu osiloskop çıktısı | 93 |
| | |

ÖZET

Sınıflandırma işlemi, benzer özellik taşıyan objelerin farklı özellikte olanlardan ayırt edilmesi şeklinde tanımlanabilir ve otomatik hedef belirleme, yapay zekâ, yapay sinir ağları, analog-sayısal dönüştürücüler, tıbbi tanı, kuantalama, görüntü işleme, istatistik gibi konularda kullanım alanı bulur. Diğer yandan, gerek gerçek dünyada gerekse sayısal dünyada, verilerin sınıflandırılması büyük önem taşımaktadır. Sınıflandırma yöntemleri ilk olarak 1960'lı yıllarda örüntü sınıflandırma adı altında görülmeye başlanmış ve ilişkin yazılımlarda basit yapılar ele alınmıştır; ilk gerçeklenen yapıda en yakın komşu yakınsaması kullanılmıştır.

Bugüne kadar sınıflandırma işlemi, çeşitli algoritmalar yardımıyla genellikle yazılımsal olarak yapılmıştır. Oysaki gerçek zamanda çalışma gerektiren bazı uygulamalarda, sınıflandırma işleminin donanımsal olarak da gerçeklenmesi önem kazanmaktadır.

Bu çalışmada, ayarlanabilir sınıflandırıcı devreleri ve uygulama alanları incelenmiştir. Bu amaçla, çalışmanın donanımsal gerçeklemeyle ilgili kısmında, önce çekirdek devre diye adlandırılan temel bir yapı tasarlanmış ve bu çekirdek devrelerden oluşan çok-girişli çokmimarisi geliştirilmiştir. Bu mimari ile sınıflandırılabilen ve çıkışlı bir sınıflandırıcı sınıflandırılamayan veri kümeleri incelenmiş, sınıflandırılamayan veri kümelerinin ayırt edilebilmesi için çekirdek devre yapıları ile kullanılabilecek Çarpan Devre yapısı gerçekleştirilmiştir. Dolayısıyla gerek sadece çekirdek devre yapıların öncelikli olarak kullanılacak uygulamalarda istenilen çalışma koşullarını çekirdek devre-2'nin sağlayacağı düşünülmüştür, gerekse çarpan devre yapılarını beraber kullanarak veri kümelerinin uygun parametreleri yardımıyla sınıflandırılabileceği kontrol gösterilmiştir. Bu kontrol parametrelerinin bulunmasını sağlayan eğitim algoritmaları da ayrıntılı olarak incelenmiştir.

Geliştirilen devreler simülasyonlarla ve ayrık devre elemanlarıyla gerçeklenerek sınanmış, gerekli değişiklik ve düzeltmeler yapıldıktan sonra 0.35 µm AMS CMOS teknoloji parametreleriyle sınıflandırma işlemlerinde kullanılacak DU-TCC 1209 İntegre Devresinin serimi tasarlanmıştır. Çeşitli sınama aşamalarını geçen serim üretildikten sonra DU-TCC 1209 laboratuarda test edilerek yazılımsal sonuçların ölçümlerle çok başarılı bir uyum halinde olduğu gözlenmiştir.

Anahtar kelimeler: Sınıflandırıcı Devreler, CMOS, Akım Taşıyıcı. Karakter Tanıma

SUMMARY

The aim of classification is to assign an unknown object to a class containing similar objects. Classifier circuits can find applications in various fields of applied science such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks, analog digital converters, quantizers and statistics. Therefore, classification is especially important in the real world applications or in the digital world. First basic classification methods using the nearest neighbour concept date back to 1960 with pattern recognition algorithms.

Nowadays classification is generally achieved with the help of some algorithms in software packages. However, hardware implementation of classifier circuits are important for the applications that require real-time processing.

In this project, new possibilities for CMOS classifier circuits and their applications are investigated. For that reason, firstly a hardware implementation of a basic classification unit called core cell is presented then a multiple-input and multiple-output classification topology is constructed with these core cells. The data sets that can be classified or not-classified with that multiple-input and multiple-output classifier circuits are examined; a Scalor Circuit to be used with core cells is realized to classify data sets unclassifiable with core cells. As a result it is shown that data sets can be classified, using core cells with or without scalor circuits but with appropriate control parameters to provide tunability. Learning algorithms have been investigated, developed and applied to obtain these control parameters.

Developed circuits have been soft and hard-tested with discrete components, corrected and improved, then its layout designed with 0.35 μ m AMS CMOS technology parameters, tested, ameliorated and the resulting IC entitled DU-TCC 1209 manufactured.

Finally, DU-TCC 1209 has been lab tested with several classifier applications and soft versus hard test results have been observed to be in almost perfect agreement.

Key words: Classifier circuits, CMOS, Current Conveyor, Character Recognition.

1. GİRİŞ

Sınıflandırma genellikle otomatik hedef belirleme, yapay zekâ, yapay sinir ağları, analogsayısal dönüştürücüler, tıbbi tanı, kuantalama, görüntü işleme, istatistik gibi konularda kullanılmaktadır (LIU 2000, TZANAKOU 2000). Bu konuda yapılan çalışmalar yazılımsal ve donanımsal olmak üzere iki kısımda incelenebilir. Literatürde gerek yazılımsal gerekse donanımsal sınıflandırıcılarla ilgili birçok yayına rastlanmaktadır (COVER 1967, BISHOP 1996). Çok yaygın olan yapay sinir ağı tabanlı sınıflandırıcılara ise hem yazılımsal hem de donanımsal olarak bakmak daha doğru olur. Genellikle donanımsal olarak tasarlanmış sınıflandırıcılar, farklı ağ yapıları sentezleyen Yapay Sinir Ağlarının (YSA) gerçeklemesidir. YSA algoritmalarının büyük bir çoğunluğu bilgisayara uyarlanabilmektedir. YSA, biyolojik nöron hücresinin yapısı ve öğrenme özelliklerinden esinlenerek geliştirilmiş bir hesaplama sistemi olup sınıflandırmaya çok elverişlidir. Bu ağların mimarisini ve öğrenme algoritmalarını geliştirmeye yönelik literatürde birçok çalışmalar bulunmaktadır (BEIU 1996, DUDA 2000). Bu çalışmaların geliştirdikleri modeller yazılım ve donanım ortamında kullanım sağlamışlardır (JAIN 2000, CHENYZ,2001, ABDEL-ATY-ZOHDY 1999). YSA'nın seçilmesindeki başlıca neden ise, çok karmaşık ve çok fazla parametre içeren durumlarda bile kullanılabiliyor olmalarıdır (RIPLEY 1996). Yazılımsal olarak çalıştırıldıklarında sakıncalarının başında, gerçek zamanda çalıştırılamamaları, öğrenme algoritma süresinin fazla olması, sınıflandırma yapacağı nesnelerin birbirine çok benzer özellikler göstermesi gelir; ayrıca parametre sayısı çok arttığında yavaş çalışır hale gelmeleri ise diğer bir sakıncadır (YAMASAKI 2001). Bütün bunların ana nedeni, paralel çalışma göre ortaya atılmış YSA'nın, ardışıl makinelere yönelik algoritmalarla esasına çalıştırılmasıdır; dolayısıyla mevcut sınıflandırıcı algoritmalarının yazılımsal olmalarına karşın, hızlı ve gerçek zamanda çalışma arzu edildiğinde, donanımsal olarak gerçeklenmeleri gerekliliği doğmuştur (YAMASAKI 2003).

1.1 Literatür Taraması

1.1.1 Yazılımsal yöntemler

Genel bakış açısından, bir sınıflandırıcı tasarımı iki farklı kategoride ele alınabilir: eğitimli ve eğitimsiz tasarım. Eğitimli tasarımda daha önce tanımlanmış bilinen bir giriş sınıfının bilinen çıkışlarıyla, sınıflandırıcının aynı girişlere karşı düşen çıkışları karşılaştırılarak bir hata terimi tanımlanır. Bu terim sıfır oluncaya kadar, sınıflandırıcının parametreleri değiştirilir. Eğitimsiz sınıflandırmada ise bir veri eğitim kümesi bulunmayıp, bir yakınlık kuralına göre verilerin gruplandırılmasının yapılması olarak düşünülebilir (TZANAKOU, 2000). Literatürde kullanılan sınıflandırma yaklaşımlarının çoğunluğu eğitimli sınıflandırma kategorisinde yer almaktadır (REYES, 1995). Bu kategoride yapay sinir ağları ve en yakın komşu gibi yaklaşımlar kullanılmaktadır (COVER, 1967). YSA tabanlı olan sınıflandırma yapıları, günümüzde en yaygın kullanılan, başarımını ispatlamış çok güçlü sınıflandırıcı türleridirler (ZHAO, 2002). YSA'ya yönelik ilk çalışma, 1943 yılında Mc-Culloch ve Pitts'in tarafından yapılmıştır (MCCULLOCH, 1943). Bu çalışmada basit mantık fonksiyonlarının gerçekleştirildiği matematiksel bir model kullanılmıştır. Daha sonraki çalışmalarda farklı mimari ve öğrenme yapısına sahip ağlar geliştirilmiştir (GALUSHKIN, 1992). Bu ağlardan çok katmanlı algılayıcılar ve radyal tabanlı fonksiyon ağları literatürde sınıflandırma gibi pratik uygulamalarda oldukça yaygın olarak kullanılmıştır (MOODY, 1989). Ayrıca YSA'ların genelleme yeteneklerinin olması, çok büyük boyutlu örüntü tanıma uygulamaları açısından da oldukça önemlidir (XIAOQIN, 2003; SUN, 1997).

Sınıflandırma işleminin bir başka yöntemi de, örüntüleri bulundukları uzayda tanımlı bir mesafe fonksiyonuna (metrik) göre, kendilerine en yakın sınıflara minimum hata ile atamaktır (HUNG, 1999).

Örüntü sınıflandırma işlemi temel olarak iki gruba ayrılabilir: bunlardan ilki matematiksel ve istatistiksel tabanlı olanlar, ikincisi ise YSA tabanlı algoritmalardır. Sınıflandırıcılar başarım düzeyi açısından karşılaştırıldığında, YSA'ları kullanılarak elde edilen sınıflandırıcı sonuçlarının istatistiksel tabanlı sınıflandırıcı sonuçlarından daha başarılı olduğu görülmüştür; ancak öğrenme sürelerinin de uzun olduğu bilinmektedir (SUN, 1999; HUNG, 1999; ZHOU, 2000). Bazı çalışmalarda karma yöntemler de önerilmiştir (CAMBONI, 2001).

Diğer bir sınıflandırma yöntemi ise destek vektör makinesidir. Bu yöntem iki sınıflı veriye ait nokta kümesini ayıran en iyi bir soyut-düzlem bulmaya çalışan bir tekniktir. Diğer bir ifade

ile bu yöntem veri sınıflarını uygun soyut-düzlemler ile ayırmaya çalışır. Uygun soyutdüzlemin bulunması ise sınıflandırılacak veriler ile soyut-düzlemin mesafesinin minimizasyonu ile yapılır. Bu yöntemin sakıncalarının başında ise, belirli veriler için en iyi ayıran düzlemini bulduktan sonra yeni veriler geldiğinde, sadece yeni verilere değil eski verilere de ihtiyaç duyulması gelir (JING, 2006).

Literatürde veri sınıflandırması radyal tabanlı fonksiyonlar kullanılarak da yapılmaktadır (OYANG, 2005). Buradaki veri sınıflandırma yönteminde, sınıflandırma sonuçları belli olan belirli veriler için ağ parametreleri saptanır ve bu parametreler benzer dağılım gösteren başka veriler için de kullanılır. Başka bir deyişle tekrar eğitilmesi gerekmemektedir. Ancak radyal tabanlı fonksiyonlar kullanıldığında doğru sonuç elde etme kesinliği azalırken hızlı sonuç alınabilmektedir; oysa destek vektör makinesi ile durum tam tersi olmaktadır.

Örüntü sınıflandırma için kullanılan bir yöntem de vektör kuantalayıcıdır. Bu yöntem örüntü tanımada olduğu kadar, özellikle veri sıkıştırılmasında da kullanılmaktadır. Vektör kuantalayıcı yapılarından, giriş dizisinin örnek dizilerden en çok hangisine benzediğini bulmakta yararlanılır (HUNG, 1999; LUBKIN, 1998). Vektör kuantalama, genellikle daha hızlı bir performans elde etmek için, özellikleri birbirine yakın bir işaret kümesinin, tek bir vektörle temsil edilerek, tek bir kodla kodlanması şeklinde tanımlanabilir.

Bu veri sınıflandırma yöntemlerinden başka literatürde, sınıflandırma yazılımlarında kullanılabilen, k-en-yakın komşu (k-th nearest neighbourhood) ve Voronoi algoritması gibi çeşitli yöntemler de vardır (COVER, 1967). Bu yöntemlerden, Voronoi diyagramı yaklaşımı örüntü sınıflandırmada farklı veri girişlerinin ayırt edilmesi amacıyla kullanılmaktadır (REYES, 1995).

Son olarak, yine literatürde özellikle sınıflandırma işleminin bir alt kümesi olan analogsayısal dönüştürme işlemi, parça parça lineerleştirme yaklaşımı ve de nöro-bulanık sistemlerde kullanılan trapezoidal geçiş fonksiyonu da ayrıntılı olarak incelenmiştir (KACHARE, 2005; BİLGİLİ, 2005; AKSIN, 2000). Trapezoidal geçiş fonksiyonu yardımı ile görüntü işleme konusunda yapılmış çeşitli uygulamalar bulunmaktadır (BİLGİLİ, 2006).

Ayrıca bir çeşit veri sınıflandırma yöntemi olarak düşünülebilecek kuantalama da, kodlama ve veri sıkıştırma uygulamalarında yaygın olarak kullanılmaktadır (GRAY, 1998).

1.1.2 Donanımsal yapılar

Literatürde donanımsal olarak gerçekleştirilmiş sınıflandırıcı yapıları, gerçek zamanlı uygulamalar için elverişli olmaktadırlar. Bu yapılar genellikle yapı birimi bir nöron modeli olan, farklı ağ yapıları sentezleyen, programlanabilen veva programlanamayan devre ve tümdevre mimarileri olarak karşımıza çıkmaktadır. Özellikle yapay sinir ağlarının donanım gerçeklemeleri olan nöroişlemciler, standart mikroişlemciler ile uygulamalarda kullanılmaktadır. Böylece nörol hesaplanmalardan donanımsal olarak faydalanılmış olunur. Ancak bu donanım uygulamaları giriş-çıkış sayısı, ağ tipi, sabit aktivasyon fonksiyonu, sınırlı çalışma aralığı gibi bazı kısıtlamalar içermektedir. Ancak YSA'nın yazılım ortamında elde edilen esneklik, matematik işlem kabiliyetinin üstünlüğü gibi yeteneklere donanım gerçeklemelerinin getirdiği kısıtlamalar nedeniyle tam olarak ulaşılamamaktadır (BEIU, 1996). Bunun temel nedeni, donanımların hücreler arası yoğun arabağlantılara izin verecek kadar gelişmiş olmamasıdır. Ayrıca yazılım ortamında geliştirilen YSA algoritmalarının donanım uygulamaları için uygun olmayışı, işlem karmaşıklığının yüksek oluşu gibi nedenler de gösterilebilir. Literatürde yer alan YSA'nın donanım gerçeklemeleri, uygulama amacına ve kullanılan teknolojiye bağlı olarak çeşitlilik göstermektedir. Bu donanım gerçeklemeleri genel olarak uygulamaya özgü, özel amaçlı tümdevre yapıları ile genel amaçlı tümdevre yapıları olarak karşımıza çıkmaktadır. Belli bir sınıflandırma mimarisi ve algoritmasına uygun tasarlanan tümdevreler dışında standart işlemciler üzerinde de çeşitli topolojiler sentezlenebilmektedir. Uygulamaya özgü tasarlanan tümdevreler, belirli bir uygulamaya göre tasarlandığından, sınırlı büyüklüklere sahip olup böyle bir tümdevrenin başka bir uygulama için kullanımı mümkün değildir (LEONG, 1995). Genel amaçlı tümdevreler bu kısıtlamayı en aza indirmek amacıyla tasarlanmıştır. Gerçeklenen tümdevreler genel veya özel amaçlı olması açısından farklılık göstermesi dışında, uygulanan mimarinin analog, sayısal veya karma tasarlanmış olmasına, VLSI tasarım tekniği ve teknolojisine, eğiticili veya eğiticisiz öğrenme kuralına sahip olmasına, ağırlıkların tüm devre üzerinde saklanabiliyor veya saklanamıyor olmasına, tümdevre üzerinde saklanan ağırlıkların analog veya sayısal olmasına, eğitim işleminin tümdevre üzerinde yapılıp yapılmamasına, öğrenme sürecinde kullanılan algoritmaya göre literatürde çeşitlilik göstermektedir (AYBAY, 1996; ERKMEN, 2007). Ayrıca literatürde yapay sinir ağı donanımları ile ilgili çalışmalar da ayrıntılı bir şekilde ele alınmıştır (AVCI, 2005).

Literatürde kullanılan diğer bir donanımsal sınıflandırıcı yapısı da, yük tabanlı, sabit ağırlıklı olan Hamming sınıflandırıcısıdır (CİLİNGİROĞLU, 1993). Gerçeklenen bu devrenin üstünlükleri, tek besleme gerilimi (+5 V) kullanılıyor ve statik güç tüketimi olmadan da yüksek hızda çalışabiliyor, ancak en önemli sakıncası ise ağırlıkları değiştirilemediğinden sabit programlı olmasıdır.

Sınıflandırmada, iki vektör dizisinin farkını alarak sınıflandırma işlemini gerçekleştiren devreler de mevcuttur. Böyle bir devre 0.35 µm CMOS teknolojisi kullanılarak önerilmiştir (AKSIN, 2005).



Şekil 1.1 : Donanımsal gerçeklenmiş sınıflandırıcı bloğu (AKSIN, 2005).

Donanımsal olarak gerçeklenen iki vektör dizisinin farkını hesaplayan genel bir sınıflandırıcı yapısı Şekil 1.1'de gösterilmiştir. Bu şekildeki $(x_1, \ldots, x_i, \ldots, x_n)$ giriş dizisi vektörünü $((T_{i1}, T_{i2}, \ldots, T_{1n}), i=1, \ldots, m)$ ise giriş vektörüne uzaklıkları hesaplanacak olan sabit örnek vektör dizisini göstermektedir. Kazanan Hepsini Alır (KHA) bloğu ise hesaplanmış uzaklıkların karşılaştırılıp çıkışa yollandığı bloktur. Gerçeklenen bu devreler, genel olarak iki temel bloktan oluşmaktadır. Uzaklık hesaplamalarının yapılmakta olduğu ilk bloğun yapısının özelliği, paralel bağlanmış temel fark alma devrelerinden oluşmasıdır. Bu bloğun yapısı gereği uzaklık hesabı, algoritma olarak sınıflandırmada kullanılan bir yöntemin donanımsal olarak gerçeklenmesiyle yapılmaktadır. İkinci blokta ise genellikle KHA tipinde çıkış katları

kullanılmaktadır (DONCKERS, 1999). Çıkış katlarında kullanılan bu yapıların temel amacı ise basit bir karşılaştırma yapmaktan ibarettir. Bunun nedeni, literatürde önerilen donanımsal sınıflandırıcı yapılarında, iki veri arasındaki Hamming uzaklığının hesaplanması ve elde edilen uzaklık çıktılarının birbirleriyle karşılaştırılması yapılarak sınıflandırma işleminin gerçeklenmesidir (ÇİLİNGİROĞLU, 1998). Literatürde sabit ağırlıklı Hamming sınıflandırıcılarının kullanılmasının bir başka sebebi ise yüksek hızlı uygulamalara elverişli olmalarıdır. Böyle bir sınıflandırıcı devresi 2.4 µm CMOS teknolojisi kullanılarak gerçekleştirilmiş ve tam sayıların tanınmasında kullanılmıştır; sınıflandırma hızı 10 MHz olarak verilmiştir (GRANT, 1994). Bu tip yapılarda ağırlıkların değişmiyor olması başka uygulamalarda kullanılamaması açısından bir dezavantaj olarak karşımıza çıkmaktadır. Hamming uzaklığı kullana sınıflandırıcılar analog olarak donanımsal gerçekleştirildiklerinde eşleşme problemi ile hatalı karar vermeye sebep olmaktadırlar; dolayısıyla eşleşme hataları ve sınırlamaları literatürde incelenmiştir (KUMAR, 1993).

Sınıflandırma işleminde kullanılan diğer bir yaklaşım ise, yakınlık ölçütü olarak Euclid uzaklığının kullanıldığı yöntemlerdir (ONAT, 1997). Bu yöntemler, hücre devrelerinin paralel bağlanmasıyla temel uzaklık hesabı sağlandığı için, alan verimliği açısından çok elverişlidir. Literatürde, temel devre yapısının 4 tranzistor ve 2 kapasite elemanı ile gerçeklendiği örnekler de vardır (ÇİLİNGİROĞLU, 1998). Ancak temel sakınca, bu devrelerde tasarım esnasında kapasite değerlerinin seçilmesi zorunluluğundan dolayı ağırlık katsayılarının kullanım aşamasında değiştirilememesidir. Dolayısıyla ağırlık katsayılarının değiştirilebilir olması sınıflandırıcıların genel amaçlı kullanımı açısından önem taşımaktadır. Tasarımlarda ağırlık değerleri hem analog hemde sayısal olarak tasarlanmıştır. Bu ağırlık değerlerinin sayısal ve tamsayı biçiminde önerildiği ve böylelikle kuantalama hatasının oluşmasının da engellendiği çalışmalar bulunmaktadır (GRAGHICI, 1999; ALMEİDA, 1993).

Sınıflandırmada başka bir yaklaşım da eğitilici öğrenme algoritmasına dayanan destek vektör makinesidir. Destek vektör makinesi doğrusal ayırt edilebilen 2-sınıf problemlerinin çözümünden yola çıkarak doğrusal olarak ayırt edilemeyen veya çoklu sınıf problemlerinin çözümünde kullanılan bir yöntemdir. Gerçek zamanlı sınıflandırma uygulamaları bu öğrenme algoritmasının eğitim süresinin uzun olmasından dolayı zor bir problem olmaktadır. Eğitimde harcanan hesaplama süresinin iyileştirilmesi amacıyla kısıtlı bir eğitim kümesi kullanılarak donanımsal olarak destek vektörleri elde edilmiştir. Bu kısıtlı kümeden elde edilen vektörler sınıflandırma başarımını azaltırken, yapının karmaşıklığının da azalmasını sağlamıştır.

Literatürde bu şekilde tasarlanmış bir devre 0.5 µm CMOS teknolojisi kullanılarak gerçekleştirilmiş ve güç tüketimi 5.9 mW olarak verilmiştir (GENOV, 2003). Sınıflandırmada destek vektör makinesinin eğitim amacıyla kullanıldığı bir başka çalışmada ise, devre mimarisinde yüzen kapılar kullanılarak, analog işlemci bloklarının zayıf evirtimde çalışması sonucu nW'lar ile mW'lar mertebesinde düşük güç tüketim değerlerine ulaşılmıştır (CHAKBARRTTY, 2007). Destek vektör makineleri de radyal tabanlı fonksiyonlar kullanılmasıyla donanımsal olarak gerçeklenmiştir (PENG, 2008). Bu fonksiyonlarda kullanılan Gauss fonksiyonunun ortalama değeri ve varyansı yüzen kapılar kullanılarak ayarlanabilmektedir. Gerçeklemelerin analog tasarımlarında, akım aynaları ve logaritma tabanlı süzgeçler de kullanılmıştır. Ancak bu durum geniş yapay sinir ağlarında çok elverişli olmadığı için, direnç ve kuvvetlendiriciler tercih edilmiştir (PENG, 2008). Ayrıca radyal tabanlı fonksiyon sınıflandırıcısına KHA yapısı da eklenerek analog vektör kuantalayıcıya dönüştürülmüştür (PENG, 2007).

Karar ağaçları da sınıflandırmada kullanılan yöntemler arasında yer almaktadır. Bu yöntem geniş veri kümelerinde dahi yüksek doğrulukla çalışmaktadır. Ancak veri boyutlarının geniş olduğu uygulamalarda yoğun algoritmalardan dolayı işlem süresi çok uzun olmaktadır. Bunun için karar ağacı sınıflandırıcısı donanımsal olarak FPGA (Field Programmable Gate Array) yapıları kullanılarak gerçeklenmiştir (NARAYAN, 2007). Sayısal tasarımların gürültü bağışıklığının iyi olmasından dolayı, yüksek doğruluğa sahip çıkışlar üretebilmektedir. Ayrıca sayısal tasarım tekniklerinden dolayı, donanım tanımlama dilleri (VHDL) kullanılarak gerçekleştirilen sayısal bir tasarım, FPGA yongalarına kolayca aktarılabilmekte ve hızlı prototipler üretilebilmektedir. Donanım tanımlama dilleri kullanarak tasarlanmış sınıflandırıcı mimarileri literatürde bulunmaktadır (REAZ, 2002). Sayısal tasarımların gürültü bağışıklığı ve tasarım kolaylığı gibi avantajlarının yanında, sayısal yapı blokları ile fonksiyon gerçekleştirmek analog devrelere göre çok daha fazla sayıda tranzistor gerektirmektedir. Yapılan çalışmalarda genelde analog ve sayısal tasarım tekniklerinin olumlu özelliklerini bir arada kullanan karma tümdevre tasarımları yer almaktadır.

Bütün bu yukarıda sözü edilen sınıflandırıcı donanımlarına ek olarak literatürde örüntü tanımada kullanılan donanımsal sınıflandırıcı yapısı da karşımıza çıkmaktadır. Donanımsal olarak, 2 µm CMOS teknolojisi ile 2.2×2.2 mm² alan üzerine gerçeklenmiş ve 6 adet nöron içeren örüntü sınıflandırıcı yapısı mevcuttur (WANG, 1991). Bu yapıda kullanılan öğrenme kısmı mikrobilgisayar aracılığı ile gerçekleştirilmiştir ve devre mW'lar mertebesinde güç

tüketmektedir (WANG, 1991). Yüz ve karakter gibi karmaşık görsel örüntülerin sınıflandırılması için resimlerin sayısal bir kamera yardımıyla alınması ve sayısal işaret işleme teknikleri kullanılarak mikroişlemciler aracılığı ile işlenmesi sağlanır. Ancak bu işlemler yüksek güç tüketimi gerektirmektedir. Bunların daha az güç tüketerek gerçekleştirilmesi için analog donanımsal görsel örüntü sınıflandırıcılar önerilmiştir (BRIDGES, 2006). Literatürde +5 V besleme gerilimi ile beslenen, 0.35 μm CMOS teknolojisi ile gerçeklenen ve 1.25 mW güç tüketen uygulamalar bulunmaktadır (BRIDGES, 2006).

Örüntü tanımada birçok sınıflandırıcıyı birleştirerek kullanmak gelişmiş bir yöntem olarak bilinmektedir. Literatürde karar ağaçlarının sınıflandırıcı olarak kullanıldığı örüntü tanıma devreleri de mevcuttur (BERMAK, 2003). Ancak bu tip sınıflandırıcılarda genel başarım yüksek dahi olsa fazla miktarda bellek gereksinimi ve hesaplama süresinin uzunluğu bir sakınca olmakta ve gerçek zamanlı kullanılmalarına engel teşkil etmektedir. Akıllı fotosensörlerde, düşük çözünürlüklü örüntü sınıflandırıcı uygulamalarında kullanılabilirler. Ayrıca yapay sinir ağlarının farklı örnekler için eğitilebilme yetenekleri sayesinde fotosensensör dizileri ile birlikte kullanılarak CMOS devreler ile gerçeklenmiş örüntü sınıflandırma uygulamaları mevcuttur (DJAHANSHAHI, 1996).

Böylece, yukarıda literatürdeki mevcut sınıflandırıcı devreleri incelenmiş ve bu sınıflandırıcıların donanımsal olarak gerçekleştirilmesi için pek çok farklı yapıda donanıma özel öğrenme algoritmaları gerektiği görülmüştür. Donanımsal sınıflandırıcı tasarım teknikleri analog, sayısal ve karma olacak biçimde tümdevre yapıları gerçeklenmiştir. Bu yapıların bir kısmı genel amaçlı, bir kısmı da özel amaçlı tasarımlardır.

Tüm bu tasarımların ışığında, temel sınıflandırıcı yapısı kırmık alanında fazla yer kaplamayacak, güç tüketimi düşük olacak ve üretilmiş olan temel sınıflandırıcı tümdevreleri aralarında bağlanabilecek ve böylelikle çok girişli devrelerin elde edilmesine olanak tanıyacak bir sınıflandırıcı devresinin bulunmadığı, ve böyle bir devreye ihtiyaç olduğu ortaya çıkmıştır. Ayrıca böyle bir tümdevrenin, ağırlıkları dışarıdan ayarlanabilecek şekilde tasarlanmış ve ağırlık katsayılarının bulunması için tümdevreye uygun öğrenme algoritmalarının geliştirilmiş olması da gerekmektedir.

1.2 Projede Kullanılan Yöntem ve Amaç

Bu proje kapsamında kullanılan yöntem ve amaç aşağıdaki gibi özetlenebilir.

- a) Genel amaçlı, esnek bir şekilde kullanılabilecek temel bir sınıflandırıcı yapısı gerçeklenmek istenmektedir. Sözü edilen sınıflandırıcı temel devrelerin kullanım amacına göre paralel, seri, ard arda ya da herhangi bir biçimde bağlanarak daha yüksek boyutlu verilerin sınıflandırılmasına olanak vermelidir.
- b) Bu devreler ayarlanabilir olmalıdır: başka bir deyişle çeşitli veri tiplerine göre, verileri sınıflara ayıran bölgeleri belirleyen devre parametrelerinin kullanıcı tarafından saptanması ve/veya bir uygulamadan ötekine değiştirilebilmesine imkan tanıyacak şekilde tasarlanmalıdır.
- c) Ayrıca bu devreler, veri tiplerinin sınıflandırılmasında ortaya çıkan sınıf bölgelerinin olabildiğince esnek olmasına izin verebilmelidir. Dolaysıyla verileri sınıflandırmak için uygun bir yöntem geliştirmek ve ayrıca bu tasarımlara uyarlanabilecek öğrenme algoritmalarını da oluşturmak ve yapılara uyarlamak gerkmektedir.
- d) Son olarak geliştirilen devrenin IC serimi yapılmalı,, sınanmalı, iyileştirilmeli, nihai serim üretilmeli ve simülasyon/ölçüm sonuçları karşılaştırılmalıdır.

1.3 Çalışmada Yapılanlar

Sınıflandırıcı devrelerinin tasarlanmasında izlenen yol aşağıda özetlenmiştir:

Bölüm 2'de sınıflandırma amacıyla, önce 1.2 a)'da sözü edilen sınıflandırıcı için iki adet temel devre bloğu tasarlanmıştır. Tasarlanan sınıflandırıcı devre blokları temel bir yapı taşı niteliği taşımakta, başla bir deyişle tek boyutlu bir sınıflandırıcı devresi olmaktadırlar. Bu yapıya çekirdek devre denilmiş, hem yazılımsal hem de donanımsal gerçeklenerek simülasyonları ve karşılaştırılmaları sunulmuştur. Bu devreler ile elde edilebilecek sınıflandırına bölgeleri incelenmiş, bu devreler ile kurulan değişik topolojiler sayesinde ne tür farklı veri tiplerinin sınıflandırılabildiği gösterilmiştir. Çekirdek devrelerin çeşitli şekilde bağlanması ile çok boyutlu sınıflandırıcı devreleri gerçekleştirilmiştir. Böylece sınıflandırıcı devrelerin bazı veri tipleri için esnek yapıda tasarlanmış olmasının getirdiği kullanım kolaylıkları da gösterilmiştir. 1.2 b)'de istenen ayarlanabilir olma özelliği, çekirdek devrenin kontrol parametrelerinin kullanıcı tarafından saptanabilmesi ile sağlanmıştır.

Bölüm 3'de, tasarlanmış olan sınıflandırıcı devrenin simülasyonlar yardımıyla kuantalayıcı ve karakter tanıma uygulamaları yapılmıştır; uygun tasarlandıkları durumda karakter tanıyıcı devrenin sembolde oluşan hataları da düzeltebileceği gösterilmiştir.

1.2 c) ile ifade edilen esnekliği sağlamak amacıyla çekirdek devrelerin önüne yeni bir kat eklenerek oluşturulan Çarpan Devresi (ÇAD) ve simülasyonları Bölüm 4'te incelenmiş, ÇAD yapısının nasıl gerçekleştirildiği ve bu ÇAD yapısı ile önerilen devrelerin beraber kullanımı ile hangi tür verilerin sınıflandırılabileceği de gösterilmiştir.

Bölüm 5'te, 0.35 µm AMS CMOS teknoloji parametreleriyle tasarlanan Çekirdek Devrenin, Akım Taşıyıcılarının, DU-TCC 1209 integresinin vb. serimleri, DU-TCC 1209'un testinde kullanılacak baskı devrenin çizimi ve bunların fotografları gösterilmiştir. Bu bölümde ayrıca DU-TCC 1209 ile çekirdek devrenin giriş-çıkış karakteristiğinin, 3 sınıflı bir sınıflandırıcının çıkışlarının simülasyon sonuçlarıyla uyum içinde olduğu gösterilmiş olup, büyük oranda geçmiş ara raporlarda yer almayan sonuçları ihtiva etmektedir.

Önceki bölümlerde önerilen yapılar, özgün ve farklı olduklarından sınıflandırma algoritmaları ve geliştirilen devrelere uygulanması Bölüm 6'da ele alınmıştır. Fisher tabanlı algoritma, perseptron öğrenme algoritmaları incelenmiş, bu algoritmaların önerilen devreler ile nasıl kullanılacağı gösterilmiştir. Çift eşik doğrularının bulunmasında Fisher'in lineer diskriminant analizi yönteminden yararlanılarak yeni bir yöntem geliştirilmiştir.

Gerçekleştirilmiş olan sınıflandırıcının diğer uygulamalarına Bölüm 7'de yer verilmiştir. Geliştirilen sınıflayıcının veri kümelerine uygulanması amacıyla İris ve Haberman verileri Fisher tabanlı ve perseptron öğrenme algoritmaları ile sınıflandırılmış, DU-TCC 1209 ile elde edilen sonuçlar ile simülasyon sonuçları karşılaştırılarak uyum içinde oldukları gösterilmiştir; bu bölüm de diğer ara raporlarda bulunmayan sonuçları içermektedir.

8. bölüm proje kapsamında yapılan çalışmalar, elde edilen gelişmelere ilişkin sonuç ve yorumlar ile izleyebilecek ArGe çalışmalarının verilmesine ayrılmıştır.

2. SINIFLANDIRICI DEVRESİ TASARIMI VE BENZETİMLERİ

2.1 Sınıflandırıcı Yapısı Blok Diyagramı

Proje çalışmasında öncelikle, basit donanımlarla sınıflandırıcıların nasıl gerçeklenebileceği ele alınmış ve sınıflandırma işlemi için temel bir çekirdek yapı ve bu yapının sınıflandırabileceği veri kümeleri incelenmiştir. Bu temel çekirdek yapılar ile oluşturulan topolojilerle daha karmaşık veri kümelerinin de sınıflandırılabileceği gösterilmiş ve gerçeklemeleri yapılmıştır. Çekirdek Yapı (ÇY) taşı tek boyutlu bir sınıflandırıcı yapısı olarak düşünülmüş ve tasarlanmıştır. Gerçeklenmesi istenen geçiş eğrisi Şekil 2.1 ve blok diyagramı da Şekil 2.2'de verilmiştir. Bu çekirdek yapı temel sınıflandırıcı yapısını oluşturmaktadır ve çok boyutlu sınıflandırıcı yapılarının elde edilmesinde kullanılmıştır.



Şekil 2.1 : Çekirdek yapı geçiş karakteristiği.



Şekil 2.2 : Çekirdek yapının blok diyagramı.

Tasarımın, Şekil 2.1'deki karakteristikteki r_1 , r_2 , ve k parametreleri kullanıcı tarafından kolayca değiştirilebilecek ve değerleri uygulanacakları sınıflama probleminden kolayca elde edilebilecek biçimde yapılmıştır. Giriş-çıkış karakteristiği aşağıdaki gibi ifade edilir:

$$y = \begin{cases} k & r_1 < x < r_2 \\ 0 & di \breve{g} er \ haller de \end{cases}$$
(2.1)

Bu yaklaşımın bir üstünlüğü de çekirdek yapıların çeşitli şekillerde aralarında bağlanarak farklı tipteki ve boyuttaki verileri sınıflandırmak için kullanılabilmesidir; örneğin Şekil 2.3'de gösterildiği gibi paralel bağlanarak $n \times 1$ boyutlu bir sınıflandırıcı sistemi gerçekleştirilebilir. Daha karmaşık topolojiler ileriki aşamalarda verilecektir.



Şekil 2.3 : $n \times 1$ boyutlu sınıflandırıcı devrenin blok diyagramı.

Şekil 2.3'deki yapının çeşitli özel halleri göz önüne alınabilir; bir özel durum

$$x_1 = x_2 = \dots = x_n = x, \qquad r_1 < r_2 < r_3 < \dots < r_{(2n-1)} < r_{2n}$$
(2.2)

şeklinde verilebilir.

Bu durum için Şekil 2.4'de gösterilen giriş-çıkış karakteristiği elde edilir. Şekil 2.4'deki yapıdan da görüldüğü gibi bu şekilde gerçeklenmiş bir topoloji, tek boyutlu verileri, çıkış

büyüklüğünün genliğine bakarak *n* tane farklı sınıfa ayırabilmektedir. Bu yapıdan aynı zamanda *x* büyüklüğünün kuantalanmasında da yararlanılabilir.



Şekil 2.4 : Tek boyutlu sınıflandırıcı ile elde edilmek istenilen giriş-çıkış karakteristiği.

Sınıflandırıcı yapılarının diğer bir uygulaması olarak, iki boyutlu ve çok seviyeli bir sınıflandırıcı tasarımı Şekil 2.5'deki topoloji ile elde edilebilir. Bu şekilde verilen topolojide çekirdek yapı-I ve II'ye x_1 giriş değişkeni, çekirdek yapı-III ve IV'e x_2 giriş değişkeni uygulanmıştır. Tasarlanan devreler akım modunda çalıştıklarından çıkışta bir toplama devresi yoktur. Böylece elde edilen sınıflandırıcının giriş-çıkış karakteristiğini daha iyi görebilmek amacıyla kontrol değişkenlerinin

Tablo 2.1'de verilen değerleri kullanılarak MATLAB simülasyonu yapılmıştır.



Şekil 2.5 : 2×1 boyutlu çok seviyeli sınıflandırıcı yapısı blok şeması.

| Çekirdek Yapı | | | | | | | | | | | |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | Ι | | | II | | | III | | | IV | |
| r_1 | r_2 | k_1 | r_3 | r_4 | k_2 | r_5 | r_6 | k_3 | r_7 | r_8 | k_4 |
| 10 | 20 | 10 | 40 | 50 | 40 | 10 | 20 | 30 | 40 | 50 | 50 |

Tablo 2.1 : Çekirdek Yapı Değişkenleri

Bu durumda Şekil 2.6'daki gibi bir giriş-çıkış karakteristiği elde edilebilir. Bu karakteristikten görüldüğü üzere 2 girişli, 1 çıkışlı ve çok seviyeli (bu örnekte 8 farklı sınıf, 0 düzlemi de dahil edilirse 9 farklı sınıf) sınıflandırıcı yapısı elde edilir.



Şekil 2.6 : MATLAB programı ile elde edilen 2×1 boyutlu çok seviyeli sınıflandırıcı devresinin (x_1 - x_2)-y karakteristiği.

Tasarlanmış ve matematiksel modeli verilmiş devrenin gerçeklenmesi Bölüm 2.2'de ele alınmıştır.

2.2 Çekirdek Devreler

2.2.1 Çekirdek devre-1

Çalışmada tek boyutlu bir sınıflandırıcı devresi olarak, akım modunda çalışmak üzere tasarlanan ve çekirdek devre-1 olarak adlandırılan yapının işlevsel davranışını gösteren blok diyagramı ve geçiş karakteristiği sırasıyla Şekil 2.7 ve Şekil 2.8'de verilmiştir (YILDIZ, 2007a). Şekil 2.8'deki geçiş karakteristiğinden de görüldüğü gibi giriş değerleri I_1 ve I_2 arasında ise çıkış I_{H1} değerini, bunların dışında ise çıkış I_{H2} değerini almaktadır.



Şekil 2.7 : Akım-modlu çekirdek devre-1 yapısının blok diyagramı.



Şekil 2.8 : Akım-modlu çekirdek devre-1 yapısının geçiş karakteristiği.

Tasarlanmış olan çekirdek devre-1'in işlevsel blok diyagramı Şekil 2.9'da verildiği gibidir. Bu blok diyagramında giriş katı, giriş akımını gerilime dönüştürmek için kullanılırken NOR kapısı çıkışın, giriş $I_1 < I_2$ akımlarının arasındayken I_{H1} değerini, değilken I_{H2} değerini almasını sağlamaktadır. Çıkış değişkenin akım olması ise çıkış katı ile sağlanmaktadır.





Şekil 2.8'deki giriş-çıkış karakteristiğinin tanım bağıntısı (2.3) ile gösterildiği biçimdedir:

$$I_{out} = \begin{cases} I_{H1} & I_1 < I_{in} < I_2 \\ -I_{H2} & di ger \ haller de \end{cases}$$
(2.3)

Şekil 2.9'daki giriş katı ile birlikte tasarlanan evirici devresi Şekil 2.10'da verilmiştir.



Şekil 2.10 : Çekirdek devre-1'in giriş katı ve evirici.

Buradaki diyot bağlı M₁ tranzistoru ve I_{BIAS} akımı giriş akımını gerilime dönüştürmek için kullanılır. M₂ ve M₃ tranzistorları $I_{BIAS}+I_1$ ve $I_{BIAS}+I_2$ akımları ile kutuplanmıştır. Dolayısıyla devrede kullanılacak olan iki farklı eşik değeri elde edilmiş olur. Diğer bir deyişle buradaki M₁, M₂ ve M₃ tranzistorlarından oluşan kat, hem bir eşik devresi hem de akımı gerilime dönüştürmek amacıyla kullanılmıştır. M₄ ve M₅ tranzistorları ise evirici olarak çalışmaktadır. Giriş katı ile V₁ ve V₂' gerilimleri elde edilmiştir.

Devrede kullanılan NOR kapısı ve çıkış katı gerçeklenmesi Şekil 2.11'de verilmiştir. Çıkış karakteristiğinde sıfır olmayan kısmın elde edilmesi için NOR kapısı kullanılmıştır. Bu yapıdaki NOR kapısı M₆, M₇, M₈ ve M₉ tranzistorlarından oluşmaktadır. Dolayısıyla *V*_{out} çıkışı,

$$V_{out} = \begin{cases} V_{DD} & I_1 < I_2 \\ Vss & diger hallerde \end{cases}$$
(2.4)

ifadesi ile verilir.

Son olarak, kullanılan M_{10} , M_{11} , M_{12} ve M_{13} tranzistorları çıkış katını oluşturmaktadır. Çıkış katında kullanılan I_{H1} ve I_{H2} akım kaynakları geçiş karakteristiğindeki tepe noktalarının ayarlanmasını sağlamaktadır.



Şekil 2.11 : NOR kapısı ve çıkış katı.

Çıkış katındaki I_o^+ ve I_o^- akımları da V_{out} gerilimi cinsinden aşağıdaki ifadeler ile verilir:

$$I_{o}^{+} = \begin{cases} I_{H1} & V_{out} = V_{DD} \\ -I_{H2} & V_{out} = V_{SS}, \end{cases}$$
(2.5)

$$I_{o}^{-} = \begin{cases} -I_{H2} & V_{out} = V_{DD} \\ I_{H1} & V_{out} = V_{SS} \end{cases}$$
(2.6)

Dolayısıyla, tasarlanmış olan devrede giriş katında kullanılan I_1 ve I_2 akımları geçiş karakteristiğinin sıfır olmayan bölgesinin ayarlanmasına olanak tanımaktadır.

Tasarlanan devrenin SPICE simülasyonları için çıkış akımı I_{out} için I_o^+ seçilmiş ve 0.35 µm AMS CMOS SPICE teknoloji parametreleri kullanılmıştır (parametreler Ek A'da verilmiştir). Besleme gerilimi V_{DD} ve $V_{SS} \pm 1.25$ V olarak alınmış, kutuplama akımı $I_{BIAS}=10$ µA ve kontrol akımları $I_1=40$ µA, $I_2=80$ µA ve $I_{HI}=I_{H2}=20$ µA olarak seçilmesi halinde I_{out} akımının I_{in} akımı ile değişim karakteristiği Şekil 2.12'de verilmiştir. Çekirdek devre-1 yapısının oluşturulmasında kullanılan tranzistor boyutları Tablo 2.2'de verilmiştir.



Şekil 2.12 : Çekirdek devre-1'in I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2} \neq 0$).

Çalışmanın devamında I_{H2} akımı 0 alınmış, I_{H1} akımı I_H olarak gösterilmiş ve simülasyonlar yapılmıştır. Kontrol akımları I_1 =40 µA, I_2 =80 µA ve I_H =20 µA seçilerek I_{out} akımının I_{in} akımı ile değişim karakteristiği çizdirilmiş ve Şekil 2.13'de verilmiştir. Çıkışın gerilim olması durumunda, V_{out} geriliminin I_{in} akımı ile değişim karakteristiği Şekil 2.14'de gösterilmiştir.

Tablo 2.2 : Çekirdek devre-1 yapısının MOS tranzistor boyutları.

| MOSFET | W [µm] | L [µm] |
|--|--------|--------|
| $M_1, M_2, M_3, M_4, M_6, M_7, M_8, M_9, M_{10}, M_{12}$ | 10.5 | 0.7 |
| $M_5, M_{11}, M_{13},$ | 5.25 | 0.7 |



Şekil 2.13 : Çekirdek devre-1'in I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2}=0$).



Şekil 2.14 : Çekirdek devre-1'in V_{out} geriliminin I_{in} akımı ile değişim karakteristiği.

1×1 boyutlu bir sınıflandırıcı devre örneği için, dört adet çekirdek devre-1 yapısı Şekil 2.15'de gösterildiği biçimde paralel olarak bağlanır. Bu tek boyutlu sınıflandırıcı yapısında kontrol akımları ile, uygun bölgeye düşen veri ayırt edilmektedir. Çekirdek devre-1 yapılarına uygulanan kontrol akımları Tablo 2.3'de verilmiştir.



Şekil 2.15 : Çekirdek devre-1 ile gerçeklenen 1×1-D sınıflandırıcı blok diyagramı.

| Çekirdek Devre-1 | I_1 (μ A) | I_2 (μ A) | $I_H(\mu A)$ |
|------------------|------------------|------------------|--------------|
| Ι | 20 | 80 | 130 |
| II | 130 | 180 | 90 |
| III | 230 | 310 | 100 |
| IV | 370 | 440 | 140 |

Tablo 2.3 : 1×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.

Çekirdek devre-1 yapıları ile oluşturulmuş 1×1 boyutlu sınıflandırıcının SPICE simülasyonu sonucu elde edilen I_{out} akımının I_{in} akımı ile değişim karakteristiği Şekil 2.16'da verilmiştir.



Şekil 2.16 : ÇD-1 ile gerçeklenen 1-D sınıflandırıcının giriş-çıkış karakteristiği.

Çekirdek devre-1 yapıları kullanılarak 1×1 boyutlu veriler dışında 2×1 hatta daha büyük $n \times m$ boyutlu veriler de sınıflandırılabilmektedir. Çok boyutlu verilerin sınıflandırılabilmesi amacıyla alt bölüm 2.1'de MatLab simülasyonu yapılan topolojiden yararlanarak çekirdek devre-1 blokları ile kurulan yapı Şekil 2.17'de verilmiştir. Bu yapıda 4 adet çekirdek devre-1 yapısı birbirine bağlanmış ve çekirdek devre I ve II'ye birinci veri kümesi, III ve IV'e ise ikinci veri kümesi giriş olarak uygulanmıştır. Böylece 2×1 boyutlu veri sınıflandırıcı yapısı oluşturulmuştur.



Şekil 2.17 : ÇD-1 ile gerçeklenen 2×1-D sınıflandırıcı blok diyagramı.

İki boyutlu sınıflandırıcının SPICE simülasyonunu yapmak amacıyla Tablo 2.4'de verilen kontrol parametreleri seçilmiş olup her bir çekirdek devre-1'in güç tüketimleri de ayrı ayrı gösterilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişmektedir.

| Çekirdek Devre-1 | I_1 (μ A) | I_2 (μ A) | $I_{H}(\mu A)$ | Güç Tüketimi |
|------------------|------------------|------------------|----------------|--------------|
| Ι | 70 µA | 140 µA | 60 µA | 0.46 mW |
| II | 210 µA | 280 µA | 120 µA | 1.03 mW |
| III | 60 µA | 140 µA | 100 µA | 0.51 mW |
| IV | 280 µA | 360 µA | 80 µA | 1.21 mW |

Tablo 2.4 : 2×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.

Çekirdek devre-1 yapıları ile oluşturulmuş 2×1 boyutlu sınıflandırıcı devresinin (I_{in1} - I_{in2})- I_{out} karakteristiğinin simülasyon sonucu Şekil 2.18'de verilmiştir.



Şekil 2.18 : ÇD-1 ile gerçeklenen 2×1 -D sınıflandırıcı (I_{in1} - I_{in2})- I_{out} karakteristiği.

Şekil 2.18'de görüldüğü gibi Şekil 2.17'deki gibi tasarlanmış iki boyutlu sınıflandırıcı devresi ile 8 farklı veri kümesi ayırt edilebilmektedir. Eğer sıfır düzlemi de bir sınıflama sonucu olarak kabul edilirse, 9 farklı sınıfa ait veri sınıflandırılmış olur.

2.2.2 Çekirdek devre-2

Şekil 2.1'deki giriş-çıkış karakteristiği farklı bir görüş açısıyla tasarlanmıştır. Devrenin blok yapısı Şekil 2.19'da verilmiştir. Gerçekleştirilen devre ilk yapıdan farklı olarak eşik devreleri kullanılarak oluşturulmuştur.



Şekil 2.19 : Akım-modlu ÇD-2'nin blok yapısı.

Çekirdek devre işlevsel diyagramı Şekil 2.20'de gösterildiği biçimde iki adet eşik devresi ve bir fark devresinden oluşmaktadır.



Şekil 2.20 : Akım-modlu ÇD-2 blok diyagramı iç yapısı (YILDIZ, 2007b).

Eşik devresinin giriş-çıkış karakteristiği Şekil 2.21'de gösterilmiş ve tanım bağıntısı

$$I_{out} = \begin{cases} I_H & I_1 < I_{in} \\ 0 & di \breve{g} er & haller de \end{cases}$$
(2.7)

şeklinde verilmiştir.



Şekil 2.21 : Eşik devresinin giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının gerçekleştirilmesi için Şekil 2.22'de verilen blok diyagram kullanılmıştır. Bu yapı ise iki eşik devresi ve bir fark alma devresinden oluşmaktadır. İki eşik devresinin çıkış akımlarının farkı alınarak Şekil 2.22'de gösterildiği gibi sınıflandırma bölgesi oluşturulmuştur. Bu bölgenin oluşturulması için eşik devrelerinin I_H akımları eşit seçilmeli ve $I_2 > I_1$ koşulu sağlanmalıdır.



Şekil 2.22 : ÇD-2 yapısının blok şeması.

Çekirdek devre-2'nin gerçekleştirilmesi için kullanılan eşik devresi Şekil 2.23 ile gösterilmiştir (MORGÜL 2005). Devrede I_1 akımı eşik değeridir. Eğer giriş akımı I_{in} , I_1 akımından büyükse devrenin çıkışındaki akım I_H ve M₃ tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur; giriş akımı I_{in} eşik akımından küçük olduğu zaman ise M₃ tranzistorunun savak gerilimi yaklaşık olarak VDD'ye eşittir. Diğer taraftan M4 tranzistorunun akımı M3 tranzistorunun savak gerilimi ile de kontrol edilmektedir. M4 ve M5 (M4 açıkken M5 kapalı, M₅ açık iken M₄ kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M₄ ya da M₅ tranzistorundan akmaktadır. Şekil 2.23'deki devre CD4007 CMOS tranzistor entegresi kullanılarak sınanmış ve osiloskop ekranında kullanılan eşik devresinde histerezis karakteristiği oluştuğu gözlenmiştir. Devrenin test düzeneği kurulurken akım kaynakları yerine direnç bağlanmıştır. Bu dirençlerin uçlarına uygulanan gerilim ile kontrol akım ve giriş akım değerleri elde edilmiştir. Pozitif geribeslemeli eşik devresini sınamak için I_1 akımı $R_1=20$ k Ω direnç üzerinden $V_{R1}=1.7$ V olacak şekilde uygulanmıştır. Benzer şekilde $R_H=20 \text{ k}\Omega$, $V_H=1.5 \text{ V}$ alınarak uygulanmış ve böylece uygun akım değerleri $I_1=85 \text{ }\mu\text{A}$ ve $I_H=$ 75 μ A olacak şekilde sağlanmıştır. Çıkışa bağlanan 100 k Ω 'luk direnç ile akımın değişimi osiloskopta gözlenerek akım değeri ölçülmüştür. Kurulan devrenin giriş-çıkış karakteristiği test sonuçları osiloskobun X-Y özelliği kullanılarak Şekil 2.24'de verilmiştir. Bu şekilde X ekseni için osiloskop skalası 0.5 V/div, Y ekseni içinse 1 V/div seçilmiştir.

Devrenin simülasyon sonucu için kontrol akımları I_1 =85 µA ve I_H =75 µA olacak şekilde ayarlanmış, devrenin aynı histerezis davranışını gösterdiği, giriş önce arttırılarak, sonra da azaltılarak uygulandığında elde edilmiş ve sonucu Şekil 2.25'de verilmiştir.

Şekil 2.23 : Pozitif geribeslemeli eşik devresi devre şeması.



Şekil 2.24 : Pozitif geribeslemeli eşik devresi osiloskop çıktısı.



Şekil 2.25 : Pozitif geribeslemeli eşik devresi simülasyon karakteristiği.

Şekil 2.23'deki devrede pozitif geribeslemeyi sağlayan M₃ tranzistoru çıkartılarak histerezis karakteristiğinin oluşmamasının sağlandığı hem donanım hem de simülasyonla sınanmış, gerçekleştirilen yeni eşik devresinin şeması Şekil 2.26'da sunulmuştur. Bu yapıda kullanılan kaynaklar basit akım aynaları olarak devreye uygulanmıştır.



Şekil 2.26 : Eşik devresi devre şeması.

Devrede I_1 akımı eşik değeridir. Eğer giriş akımı I_{in} , I_1 akımından büyükse devrenin çıkışındaki akım I_H ve M₂ tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur. Giriş akımı I_{in} eşik akımından küçük olduğu zaman da M₂ tranzistorunun savak gerilimi yaklaşık olarak V_{DD} 'ye eşittir. Diğer taraftan M₃ tranzistorunun akımı M₂ tranzistorunun savak gerilimi ile de kontrol edilmektedir. M₂ tranzistorunun savak gerilimi yaklaşık V_{SS} olduğu zaman M₃ tranzistoru kesime girerek I_H akımının M₄ üzerinden akması sağlanır. Benzer şekilde M₂ tranzistorunun savak gerilimi yaklaşık V_{DD} olduğu zaman M₃ tranzistoru iletime geçerek I_H akımının M₃ üzerinden akması sağlanır. M₃ ve M₄ (M₃ açıkken M₄ kapalı, M₄ açık iken M₃ kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M₃ ya da M₄ tranzistorundan akmaktadır.

Böylece eşik devreleri kullanılarak gerçekleştirilmiş olan çekirdek devre-2 yapısının girişçıkış karakteristiği Şekil 2.27'de verilmiştir.



Şekil 2.27 : ÇD-2 sınıflandırıcı devresi giriş-çıkış karakteristiği.

Giriş-çıkış karakteristiği Şekil 2.27'de verilen devrenin tanım bağıntısı da,

$$I_{out} = \begin{cases} I_H & I_1 < I_{in} < I_2 \\ 0 & di \check{g} er & haller de \end{cases}$$
(2.8)

şeklinde verilir.

Çekirdek devre-2 yapısının şeması Şekil 2.28'de verilmiştir. Çekirdek devre-2 yapısında kullanılan kontrol ve giriş akımları basit akım aynaları kullanılarak devreye uygulanmıştır. Şekil 2.28'de M_1 - M_4 ve M_8 - M_{11} tranzistorları eşik devrelerini oluşturmaktadır. Çıkış akımlarının farkını almak için kullanılan fark devresi ise basit akım aynası (M_6 ve M_7) kullanılarak gerçeklenmiştir. M_{13} , M_{14} ve M_{15} tranzistorları eşik devrelerine aynı I_H akımını uygulamak için kullanılmıştır. Benzer şekilde M_{16} , M_{17} ve M_{18} tranzistorları da girişlere aynı I_{in} akımını uygulamak için kullanılmıştır.

Önerilen çekirdek devre-2 yapısının SPICE simülasyonunda 0.35 μ m AMS CMOS teknoloji parametreleri kullanılmış ve besleme gerilimleri V_{DD} ve $V_{SS} \pm 1.65$ V olarak alınmıştır. Tranzistor boyutları Tablo 2.5'de verilmiştir.



Şekil 2.28 : ÇD-2 devre şeması (YILDIZ, 2007b).

Tablo 2.5 : CD-2'de kullanılan MOS tranzistor boyutları.

| MOSFET | W [µm] | L [µm] |
|--|--------|--------|
| M ₁ , M ₂ , M ₃ , M ₄ , M ₅ , M ₈ , M ₉ , M ₁₀ , M ₁₁ , M ₁₂ | 21 | 1.05 |
| $M_6, M_7, M_{13}, M_{14}, M_{15}, M_{16}, M_{17}, M_{18}, M_{19}, M_{20}, M_{21}, M_{22}$ | 67.9 | 1.05 |

 $I_1=50 \ \mu\text{A}$ ve $I_H=20 \ \mu\text{A}$ için eşik devresinin simülasyon sonuçları Şekil 2.29'da verilmiştir.


Şekil 2.29 : Eşik devresinin giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının simülasyon sonuçları I_1 =40 µA, I_2 =80 µA ve I_H =20 µA için Şekil 2.30'da gösterilmiştir. Devrenin güç tüketimi 0.38 mW olarak elde edilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.



Şekil 2.30 : ÇD-2 yapısı giriş-çıkış karakteristiği.

Tek boyutlu verilerin sınıflandırılması için, Şekil 2.15'de gösterildiği gibi, çekirdek devre-2 yapıları paralel olarak bağlanır ve

$$I_{in1} = I_{in2} = \dots = I_{inn} = I_{in}$$
(2.9)

$$I_1 < I_2 < I_3 < \dots < I_{(2n-1)} < I_{2n}$$
(2.10)

ifadeleri ile verilen koşullar sağlanacak şekilde kontrol akımları seçilir.

Kontrol akımlarının uygun olarak seçildiği durumda 1×1 boyutlu sınıflandırıcının giriş-çıkış karakteristiği Şekil 2.31'de verildiği biçimde elde edilir.



Şekil 2.31 : Tek boyutlu sınıflandırıcının giriş-çıkış karakteristiği.

1×1 boyutlu devrenin simülasyonu için 4 adet çekirdek devre-2 yapısı paralel olarak bağlanmıştır. Simülasyon sonucu Şekil 2.32'de, kontrol akımları da Tablo 2.6'da verilmiştir. Simülasyon sonuçlarından görüldüğü gibi "0" bölgesi de dahil beş farklı tipte veri sınıflandırılmaktadır. Devrenin toplam güç tüketimi 0.95 mW'tır.



Şekil 2.32 : ÇD-2 ile gerçeklenen 1×1-D sınıflandırıcı giriş-çıkış simülasyonu.

Tablo 2.6 : ÇD-2 ile gerçeklenen 1×1 -D kontrol akımları (μ A olarak).

| 1 | ÇD-1 | 1 | | ÇD-2 | 2 | | ÇD-3 | 3 | | ÇD-4 | 1 |
|-------|-------|----------|-------|-------|----------|-------|-------|----------|-------|-------|----------|
| I_1 | I_2 | I_{H1} | I_3 | I_4 | I_{H2} | I_5 | I_6 | I_{H3} | I_7 | I_8 | I_{H4} |
| 20 | 30 | 10 | 40 | 50 | 20 | 60 | 70 | 30 | 80 | 90 | 40 |

İki boyutlu sınıflandırıcı devrenin simülasyonu için Şekil 2.17'de verilen blok şeması çekirdek devre-2 yapıları kullanılarak kurulmuştur. Simülasyonda kullanılan her bir çekirdek devre-2 yapısının kontrol akım ve güç tüketim değerleri Tablo 2.7'de verilmiştir.

Tablo 2.7 : ÇD-2 ile gerçeklenen 2×1-D sınıflandırıcı devresi kontrol akımları (akımlar μA olarak ifade edilmiştir).

| Çekirdek Devre-1 | I_1 | I_2 | I_H | Güç Tüketimi |
|------------------|--------|--------|--------|--------------|
| Ι | 70 µA | 140 µA | 60 µA | 0.86 mW |
| II | 210 µA | 280 µA | 120 µA | 1.83 mW |
| III | 60 µA | 140 µA | 100 µA | 1.19 mW |
| IV | 280 µA | 360 µA | 80 µA | 1.75 mW |

Çekirdek devre-2 ile gerçekleştirilen 2×1-D sınıflandırıcı yapısının SPICE simülasyon sonucu Şekil 2.18'dekine benzer biçimde elde edilir. Çekirdek yapılar paralel bağlanarak daha fazla sayıda verinin sınıflandırılması sağlanabilir.

2.2.2.1 Çekirdek devre-2'nin ayrık elemanlar ile gerçeklenmesi

Şekil 2.28'deki devre CD4007 CMOS tranzistor entegresi kullanılarak kurulmuş, V_{DD} =5 V, V_{SS} = -5 V seçilmiştir. Devre içindeki her akım kaynağı yerine Şekil 2.33'de gösterildiği gibi seri bir direnç konularak kontrol ve giriş akımları direnç uçlarına bağlanan gerilim kaynakları ile sağlanmıştır. Gerekli akımları sağlayan direnç değerleri Tablo 2.8'de verilmiştir.



Şekil 2.33 : ÇD-2 yapısının testinde kullanılan akım kaynağı modeli.

Tablo 2.8 : Kontrol akımlarını sağlayan direnç değerleri.

| I _{in} | I_1 | I_2 | I_H |
|-----------------------------|--------------------------|----------------------------|----------------------|
| $R_{in}=20 \text{ k}\Omega$ | $R_1=20 \text{ k}\Omega$ | $R_2 = 20 \text{ k}\Omega$ | R_H =39 k Ω |

Çekirdek devre-2'yi sınamak için, V_{R1} = 1.7 V, dolayısıyla I_1 = 85 µA, V_{R2} =2.4 V, I_2 =120 µA ve I_H = 40 µA olacak şekilde ayarlanmış; çıkışa da 100 kΩ'luk direnç bağlanmıştır. Bu direnç, akımın değişimini osiloskopta gözlemek amacıyla akımı gerilime dönüştürmek için kullanılmıştır. Kurulan devrenin giriş-çıkış karakteristiğinin test sonuçları osiloskobun X-Y özelliği kullanılarak çıkartılmış ve Şekil 2.34'de verilmiştir; osiloskobun X ekseni skalası 1 V/div, Y ekseni içinse 2 V/div seçilmiştir.



Şekil 2.34 : ÇD-2 giriş-çıkış kararteristiği osiloskop sonucu.

Projede tasarlanmış olan sınıflandırıcı devrelerin literatürde bulunan benzer özellikler taşıyan devreler ile karşılaştırılması yapılmıştır. Tablo 2.9'da özetlenmiş olan karşılaştırmada (YILDIZ, 2007a) kaynağındaki devrenin, I_1 =40 µA, I_2 =80 µA ve I_H =20 µA seçilmesi durumunda güç tüketimi 0.12 mW ve yayılma gecikmesinin 5 ns seviyesinde elde edildiği görülürken aynı akım değerleri için (YILDIZ, 2008) numaralı kaynakta verilen devrenin güç tüketimi 0.21 mW, yayılma gecikmesi 3 ns olmuştur. Çekirdek devrelerinin sınıflandırıcı olarak kullanılacağı düşünüldüğünden, öncelikli olarak kullanılacak uygulamada istenilen çalışma değerleri göz önüne alınarak yapı seçilmelidir. Bunun için önerilen devrelerde yayılma gecikmeleri, güç tüketimi ve besleme gerilimi incelenmiştir. Her ne kadar devrenin güç tüketiminde az bir artış olsa dahi yayılma gecikmesinde bir iyileşme sağlanmıştır. Dolayısıyla sınıflandırma için önem taşıyan özellikler göz önüne alınarak uygun devre de seçilebilir. Ancak her iki çekirdek devre topolojisi de önerilmiş olan uygulama alanlarında kullanılabilir. Öncelikli olarak kullanılacak uygulamalarda istenilen çalışma koşullarını çekirdek devre-2'nin sağlayacağı düşünülmüştür. Projenin devamında çekirdek devre olarak çekirdek devre-2

yapısı seçilmiştir. Gerek uygulamalarda, gerekse algoritma anlatımlarında kullanılan ve şekillerde çekirdek devre olarak gösterilen blok çekirdek devre-2 yapısıdır.

| Kaynak | Üretim Teknolojisi | Besleme Gerilimi | Güç Tüketimi | Yayılma Gecikmesi | Açıklama |
|--------------------|-----------------------|---------------------|--------------|----------------------|---|
| (LIU, 2000) | 0.6 μm CMOS | 3.3 V | 14.95 mW | - | - |
| (PENG, 2007) | 0.5 μm CMOS | 3.3 V | 90 μW-160 μW | 20 μs-40 μs | - |
| (LEONG, 1993) | 1.2 μm CMOS | 3 V | 25 mW | - | |
| (BRIDGES, 2006) | 0.35 μm CMOS | 5 V | 1.25 mW | - | - |
| (LU, 2002) | 0.5 μm CMOS | ± 2.5 V | 10 mW | 0.4 μs-0.6 μs | - |
| (GATT, 2002) | 0.35 μm CMOS | $\pm 2 V$ | 2.5 mW | - | - |
| (AKSIN, 2005) | 0.35 μm CMOS | 5 V | - | - | - |
| (GRANT, 1994) | 0.5 μm CMOS | 4 V | 80 nW-840 nW | - | - |
| (YILDIZ, 2007a) | 0.35 μm CMOS | ± 1.25 V | 0.12 mW | 5 ns | ÇD-1, I_1 =40 μ A, I_2 =80 μ A, I_H =20 μ A |
| (YILDIZ, 2008) | 0.35 μm CMOS | ± 1.65 V | 0.21 mW | 3 ns | ÇD-2, I_1 =40 μ A, I_2 =80 μ A $I_{\rm H}$ =20 μ A |

Tablo 2.9 : Sınıflandırıcı devrelerin karşılaştırılması.

3. SINIFLANDIRICI DEVRE UYGULAMALARI (KUANTALAYICI VE KARAKTER TANIMA)

3.1 Kuantalayıcı

Tasarlanan çekirdek devre yapıları kullanılarak kuantalayıcı devre oluşturulabilmektedir.

Sekiz seviyeli kuantalayıcı devre oluşturmak için sekiz adet çekirdek devre Şekil 3.1'de gösterildiği biçimde paralel olarak bağlanmış ve girişlerine aynı giriş işareti (kuantalanacak işaret) uygulanmıştır.



Şekil 3.1 : Sekiz seviyeli kuantalayıcı devre blok yapısı.

Kuantalayıcı yapı, aynı zamanda tek boyutlu 8 farklı veri türünü ayırt etmek için kullanılabilecek bir sınıflandırıcı yapısıdır. Ancak yapının kuantalayıcı olarak çalışması için her bir çekirdek devrenin I_1 , I_2 ve I_H kontrol parametrelerinin Tablo 3.1'de verildiği biçimde uygulanmış olması gerekir. Çizelgede verilen parametreler doğrultusunda kuantalayıcı yapının benzetimi için devrenin girişine üçgen bir dalga uygulanmıştır. Devrenin giriş-çıkış karakteristiği Şekil 3.2'de verilmiştir. Ayrıca kuantalama devresinde kuanta aralıkları ve kuanta seviyesi kontrol akımlarına bağlı olarak değiştirilebilir.



Şekil 3.2 : Sekiz seviyeli kuantalayıcı devresi *I_{in}-I_{out}* karakteristiği.

| ÇD | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|------------------|-----|----|----|----|----|----|----|----|
| I_1 (μ A) | 0 | 5 | 10 | 15 | 20 | 25 | 30 | 35 |
| I_2 (μ A) | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 |
| $I_H(\mu A)$ | 0.1 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |

Tablo 3.1 : Kuantalayıcı Yapısında Kullanılan Çekirdek Devre Parametreleri.

Gerçekleştirilen kuantalayıcı yapı ile çekirdek devrenin kontrol parametrelerinin dışarıdan değiştirilebiliyor ve ÇD bloklarının paralel bağlanabiliyor olmasının sağladığı avantajlar, bu çalışmada ele alınan yaklaşımın üstünlüğünü göstermesi açısından da yararlı olmaktadır.

3.2 Karakter Tanıma

Gerçekleştirilen devrenin bir başka uygulaması olarak çok kullanılmakta olan "Karakter Tanıma" problemi şablon uydurma metodu kullanılarak ele alınmıştır (YILDIZ, 2010b). Karakter tanıma çok tipik bir sınıflandırma örneği olarak gösterilmektedir. Buradaki sınıflandırma işlemi ele alınan verinin bir şablona uyup uymadığının saptanması şeklindedir. Burada verilen örnekte şablonlar sayı karakterleri olarak düşünülmüş ve Şekil 3.3'de verilmiştir. Karakter tanıma, örüntü tanımanın özel bir şeklidir. Şablon uydurmadan büyük bir resmin içindeki küçük resimleri tanımada yararlanılır. Burada şablonun küçük resimler olduğu düşünülebilir. Hatta bu şablon herhangi bir karakter, resim ya da sayı olabilir. Her bir şablon Şekil 3.4'de gösterildiği gibi 4×5=20 hücreye bölünmüştür. Oluşabilecek bir satır hatasının düzeltilmesi için fazladan bir kolon kullanılmış ve karakterlerin yerleri özel olarak seçilmiştir.

Şekil 3.5'de gösterilen karakter tanıma sınıflandırıcısı, örnek şablonun, her hücresinin diğer örnek şablonlar ile karşılaştırılabileceği şekilde tasarlanmıştır (YILDIZ, 2008). Bu şekil aynı zamanda Dört Boyutlu Sınıflandırıcı Blokları (DBSB) içermektedir. DBSB yapısı, çıkışları aynı yere bağlanmış ve çıkış fonksiyonu $f(x_1, x_2, x_3, x_4)$ ile ifade edilen dört adet çekirdek devrenin paralel bağlanmasından oluşmuştur. Her şablonun, her satırının farklı bir sonuç vermesi için DBSB yapısının çıkışı,

$$f(x_1, x_2, x_3, x_4) = x_1 2^0 + x_2 2^1 + x_3 2^2 + x_4 2^3$$
(3.1)

şeklinde seçilmiştir. Her DBSB'nin çıkışı da, geçiş karakteristiği $g_i(y)$ fonksiyonu ile ifade edilen bir çekirdek devreye uygulanmıştır. Giriş örnek örüntüsünün satırları Şekil 3.5'de gösterildiği biçimde DBSB bloklarına uygulanır. Uygulanan giriş örüntüsüne göre, a_i , b_i , c_i , d_i (i=1,...,5) değerleri sayısal olarak "1" veya "0" olabilir ve DBSB yapısının çıkışı (3.1) ifadesine göre değerler alır. Bu değerler Tablo 3.2'de verilmektedir. Bu çizelgeden görüldüğü gibi çıkışlara ilişkin satırların biri, diğeri ile çakışmamaktadır. Böylece y_i (i=1,...,5) çıkışlarına bakarak giriş şablonunun hangi sayı olduğu söylenebilir. Karakter tanıma işlemi bu aşamada gerçekleşmektedir. İstenen sadece karakter tanıma işleminin gerçekleştirilmesi ise, çıkışlar toplanarak tek bir çıkış ile elde edilebilir; Σy_i ile gösterilen bu toplam her bir giriş için farklı sonuç vermektedir.



Şekil 3.3 : Örnek örüntüler.

| <i>a</i> ₁ | b 1 | <i>c</i> ₁ | <i>d</i> ₁ |
|-----------------------|-----------------------|-----------------------|-----------------------|
| <i>a</i> ₂ | b ₂ | c_2 | <i>d</i> ₂ |
| <i>a</i> ₃ | b 3 | <i>c</i> ₃ | <i>d</i> ₃ |
| <i>a</i> ₄ | b 4 | <i>c</i> ₄ | d_4 |
| <i>a</i> ₅ | b 5 | <i>c</i> ₅ | <i>d</i> ₅ |

Şekil 3.4 : Şablon hücre gösterimi.

Karakter tanıma sınıflandırıcısında, $g_i(y_i)$ fonksiyonu ile gösterilen bloğun kontrol akımları, seçilmek istenen şablonu gösterecek şekilde ayarlanır. Ayırt edilmek istenen örüntünün girişe uygulanması durumunda, ya da $g_i(y_i)$ çekirdek devresinin girişi uygun bölgede ise, z_i çıkışı sayısal olarak "1" değerini aksi durumda ise "0" değerini verecektir. Sonuç olarak, bulunmak istenen doğru şablon girişe uygulandığında "z" çıkışı genlik olarak "5" değerini, farklı bir şablon uygulandığında da "5" den daha küçük bir değeri verecektir.



Şekil 3.5 : Karakter tanıma sınıflandırıcısı blok diyagramı.

Ayrıca Tablo 3.2'de, herhangi iki farklı giriş için en fazla üç adet y_i (i=1,2,...,5) çıkışının birbiri ile çakıştığı açık bir şekilde görülmektedir. Bu durumdan hata düzeltilmesinde yararlanılabilir.

Tablo 3.2 : Farklı örüntüler için y_i (i=1,...,5) çıkış değerleri.

| Şablonlar | \mathcal{Y}_1 | \mathcal{Y}_2 | <i>Y</i> 3 | \mathcal{Y}_4 | <i>Y</i> 5 | Σy_i |
|-----------|-----------------|-----------------|------------|-----------------|------------|--------------|
| 0 | 7 | 5 | 5 | 5 | 7 | 29 |
| 1 | 8 | 8 | 8 | 8 | 8 | 40 |
| 2 | 14 | 8 | 14 | 2 | 14 | 52 |
| 3 | 14 | 8 | 12 | 8 | 14 | 56 |
| 4 | 10 | 10 | 14 | 8 | 8 | 50 |
| 5 | 7 | 1 | 7 | 4 | 7 | 26 |
| 6 | 1 | 1 | 7 | 5 | 7 | 21 |
| 7 | 7 | 4 | 4 | 4 | 4 | 23 |
| 8 | 14 | 10 | 14 | 10 | 14 | 62 |
| 9 | 7 | 5 | 7 | 4 | 4 | 27 |

Sınıflandırıcı devresinin, şablon uydurma yöntemi ile karakter tanıma uygulaması için, Şekil 3.3'de verilen şablonlar ile benzetimi yapılmıştır. Bunun için bir şablon seçilmiş, 0 ms ile 1ms arasında şablon "0", 1 ms ile 2 ms arasında şablon "1" ve diğer şablonlar 10 ms olana kadar benzer şekilde uygulanmış; benzetim 10 şablon seçilerek tekrarlanmıştır. Benzetimde "1" genliği için 10 µA değeri seçilmiştir. Karakter tanıma uygulamasının benzetim sonuçları, ilk beş şablon için Şekil 3.6'da son beş şablon için ise Şekil 3.7' de verilmiştir.



Şekil 3.6 : İlk beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları.

Bu şekillerden de görüldüğü gibi 0 ms ile 10 ms arasında sadece 1 ms'lik bir kısmın genliği 10 μ A olmakta ve devrenin istenildiği gibi çalıştığı görülmektedir. Tablo 3.2'deki sadece bir satırın, bir veya daha çok hücresinde hata oluştuğunda z_i çıkışı genlik olarak "4" değerini alacaktır. Oluşan hatanın düzeltilmesi için sınıflandırıcı bloğunun sonuna g(z) fonksiyonunu sağlayan bir adet çekirdek devre eklenebilir. Bu çekirdek devre oluşan girişine, "4" veya "5" genlikli bir işaret geldiğinde çıkışta genlik olarak "1" verilmesini dolaysıyla da hatanın düzeltilmesini sağlamış olacaktır.



Şekil 3.7 : Son beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları.

Karakter tanıma sınıflandırıcısının, aynı zamanda hata düzeltme işlemini de gerçekleştirdiğini göstermesi açısından, Şekil 3.8'de verilen hatalı şablonlar, 1 ms ile 2 ms arasında şablon "1", 2 ms ile 3 ms arasında şablon "2" ve diğerleri, 7 ms olana kadar benzer şekilde bloğa uygulanmıştır. Benzetim sonucu Şekil 3.9'da gösterilmiştir. Sonuçtan da görüldüğü gibi test şablonlarının tek bir satırında, tek ya da daha çok hücrede, hata olması durumunda doğru sınıflandırma işlemi gerçeklenmiştir (YILDIZ, 2008).



Şekil 3.8 : Hatalı test şablonları.



Şekil 3.9 : Karakter tanıma sınıflandırıcısı hata düzeltme benzetim sonucu.10

Bu örnekle, sınıflandırma uygulamaları için gerçekleştirilmiş olan sınıflandırıcı devrenin tek satır hatasını düzelterek karakter/şablon tanıma ve benzeri uygulamalarda da kullanılabileceği gösterilmiştir.

4. EĞİK IZGARALI SINIFLANDIRICILAR

4.1 Bölgelerin Oluşturulması

Literatürde, "perseptron" türü yapay sinir ağlarının sınıflandırabileceği verilerin ne türden olacağı araştırılırken bunların lineer olarak ayrılabilen (bölgelerin bir hiperdüzlemle ayrıştırılabildiği özel durum) veriler olduğu gösterilmiştir (SUN, 1997; ZENG, 2000). Ancak bir çok veri türünün lineer olarak ayrıştırılamadığı bilindiğinden, çalışmanın bu kısmında daha genel dağılımı olan verilerin sınıflandırıcı devreler ile nasıl ayırt edilebileceği ele alınmıştır. Lineer olarak ayrıştırılamayan klasik bir örnek,(XOR kapısı) verilerin Şekil 4.1'de gösterildiği biçimde taranmış bölgelere dağıldığı iki sınıftan oluşmaktadır (YILDIZ, 2009).



Şekil 4.1 : Lineer olarak sınıflandırılamayan veri kümesi.

Şekil 4.1'deki gibi bir veri dağılımı ele alındığında bu türden bir dağılım için bölgeleri ayıran eğrilerin bir hiperdüzlem (2-D durumunda eğimi tamamen keyfi doğrular) olması gerekir. Oysa bölüm2'de geliştirilen çekirdek devrelerin birbirine bağlanmasıyla elde edilen yapıların sağlayacağı eğimler sadece 0 veya ∞ olabilir (dik ızgara yapısı). Gerekli eğimlerin sağlanması amacıyla devrenin girişine uygulanacak verilerin ($x_1, x_2, ..., x_n$) belli katsayılarla ile çarpılmış (ağırlıklı) toplamını (lineer kombinasyonu) uygulamak gerekecektir.

Ağırlık katsayıları ve çekirdek devrenin kontrol akımları uygun bir sınıflandırma algoritması ile belirlenerek, Şekil 4.3'de gösterilen bölgelerin oluşması sağlanır.



Şekil 4.2 : Eğik Izgaralı veri sınıflandırıcısı.

Şekil 4.2'de gösterilen yapıda I_{in} girişine uygulanacak x_1 girişinin belli bir katsayı ile çarpılmasını sağlayan Çarpan Devresi (ÇAD) yapısı Şekil 4.4'de ve tanım bağıntısı da,

 $y = wx \tag{4.1}$

olarak verilir. Bu bağıntıdaki *w* bir sabit olup, giriş büyüklüğünü sadece ölçeklemektedir. Gerçekleştirilmiş olan çekirdek devre yapıları akım-modlu olduğu için (4.1) ifadesindeki *wx* terimi de akım modludur. Buradaki *x* terimi sınıflandırılacak verinin özelliklerini belirleyen bir değişkeni göstermektedir.



Şekil 4.3 : Dik Izgara olmayan veri bölgeleri.



Şekil 4.4 : ÇAD yapısı blok diyagramı sembolik gösterimi.

Şekil 4.2'deki bloğun ÇAD yapıları kullanılarak elde edilen blok diyagramı Şekil 4.5'de gösterilmiştir.



Şekil 4.5 : Şekil 4.2'deki bloğun iç yapısı.

Şekil 4.2'deki yapılar paralel bağlanarak Şekil 4.6'daki eğik ızgaralı sınıflandırıcı elde edilir; eğik ızgaralı sınıflandırıcının ayırdığı iki boyutlu veri bölgelerinin gösterilimi Şekil 4.7'de, bu bölgelerdeki verileri işleyen sınıflandırıcı çıkışının üç boyutlu I_{out} -(x_1 - x_2) karakteristiği ise Şekil 4.8'de verilmiştir. Buradaki üç boyutlu şekilde yükseklik değeri çekirdek devrelerin I_H parametresi ile ayarlanmaktadır. Diğer bir deyiş ile, veriler bu yüksekliklerin aldığı değerlere göre sınıflanmaktadır. Dolayısıyla tipik biri örnek olarak Şekil 4.7'de ve Şekil 4.8'de her renk farklı bir veri sınıfına karşı gelmektedir.



Şekil 4.6 : Paralel bağlanmış sınıflandırıcı devresi.



Şekil 4.2'deki yapılardan ikiden fazla sayıda paralel yapı bağlanarak ya da girişlerine 2 veri değişkeninden daha fazlasının lineer kombinezonu uygulanarak, daha farklı ve çok sayıda bölgeler de oluşturulabilir. ÇAD devresinin donanımsal gerçeklenmesi Bölüm 3.2'de incelenmiştir.



Şekil 4.8 : Eğik ızgaralı sınıflandırıcı çıkışının 3-D I_{out} - (x_1-x_2) karakteristiği.

Şekil 4.9'da gösterilen, lineer olarak sınıflandırılamayan ve de dikdörtgen ızgara bölgeler ile ayrıştırılamayan veriler, ÇAD ve ÇD yapıları kullanılarak sınıflandırılabilmektedir. Bu verilerin sınıflandırılması biçimsel olarak aşama aşama Şekil 4.9 a-d'de gösterilmiştir.



Şekil 4.9 : Verilerin ÇAD ve ÇD kullanılarak sınıflandırılması.

Dolayısıyla bu çeşit bir veri kümesinin sınıflandırılabilmesi için uygun doğruların, başka bir deyişle ÇAD devresinin parametreleri olan w_i 'ler (ağırlık katsayıları) ve ÇD parametreleri I_1 , I_2 (kontrol akımları) değerlerinin bulunması gerekmektedir. Şekil 4.9'da kesikli çizgiler ile gösterilen bu doğrulara çift eşik doğruları denilmektedir. Şekil 4.9 b'de ilk çift eşik doğruları bulunmuştur ve görülmektedir ki çift eşik doğrularını sağında ve solunda kalan veriler ayırt edilmiştir, Şekil 4.9 c'de ise önceki aşamada sınıflandırma dışında kalan veriler için tekrar çift eşik doğruları bulunmuş ve verilerin ayırt edilmesi sağlanmıştır. Bu çalışmada çift eşik doğrularının elde edilmesi için Fisher'in lineer diskriminant analiz yönteminden faydalanılmıştır (QIAN 2007). Şekil 4.8'de gösterilen yüksekliklerin sayısal değerleri ÇD parametresi I_H ile ayarlanmaktadır. Dolayısıyla her farklı yükseklik değeri ayrı bir sınıfı göstermektedir. Şekil 4.6'da gösterilmiş bir sınıflandırıcı yapısı uygulaması Bölüm 5.3'de verilmiştir.

4.2 ÇAD Devresi ve Simülasyonları

Çarpan devresinin blok diyagramı Şekil 4.4'te gösterilmişti; tanım bağıntısı (4.1) ifadesi ile verilen bu yapı, uygulamalarda kullanım kolaylığı sağlanması için giriş değişkeni gerilim, çıkış değişkeni akım olarak tasarlanmıştır. Dolayısıyla gerçekleştirilmiş olan devre blok şeması olarak Şekil 4.10'da gösterilmiştir.



Şekil 4.10 : ÇAD devresi blok şeması.

Bu devrenin tanım bağıntıları da

$$I_{out}^{+} = \frac{V_{in}}{R_1} \frac{R_2}{R_3}$$
(4.2)

$$I_{out}^{-} = -\frac{V_{in}}{R_1} \frac{R_2}{R_3}$$
(4.3)

ifadeleri ile verilmiştir.

Şekil 4.10'da kullanılan R_1 direnci, V_{in} giriş geriliminin akıma dönüştürülmesini sağlamak amacıyla kullanılmıştır. Ayrıca, aynı direnç uygulamalarda karşılaşılacak giriş verilerinin uygun değerlere normalizasyonu için de kullanılmaktadır. Diğer taraftan R_2/R_3 oranı ise giriş geriliminin, dolayısıyla akımının belli bir çarpan ile çıkışa aktarılmasını sağlamaktadır. Burada kullanılan elemanların direnç olması gerek çarpım gerekse normalizasyon işlemi sırasında esneklik sağlamaktadır. Ayrıca (4.2) ve (4.3) ifadelerinden görüldüğü gibi çıkış değişkeni, akım boyutunda olduğundan çekirdek devre yapılarına uygulanabilmektedir.

ÇAD devresinin devre şeması iki adet Çift Çıkışlı İkinci Kuşak Akım Taşıyıcı (DO-CCII) elemanı kullanılarak gerçekleştirilmiştir (FERRI 2001). DO-CCII yapısında z ucu çıkış kabul edilmek üzere, tanım bağıntısı aşağıdaki gibi verilir.

$$\begin{bmatrix} V_x \\ I_y \\ I_{z+} \\ I_{z-} \end{bmatrix} = \begin{bmatrix} 0 & \beta & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \alpha_1 & 0 & 0 & 0 \\ -\alpha_2 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_x \\ V_y \\ V_{z+} \\ V_{z-} \end{bmatrix}$$
(4.4)

Aynı yapının blok diyagramı Şekil 4.11'de ve devre şeması Şekil 4.12'de gösterilmiştir.



Şekil 4.11 : DO-CCII blok diyagramı.



Şekil 4.12 : DO-CCII devre şeması.

DO-CCII devresinin simülasyonunda 0.35 µm AMS CMOS teknoloji parametreleri ve besleme gerilimleri de V_{DD} ve $V_{SS} \pm 1.65$ V olarak kullanılmıştır. Kutuplama akımının 18 µA olması için V_B gerilimi 0.8 V olarak seçilmiştir. Şekil 4.12'de verilen devre için V_x geriliminin V_y ile değişim karakteristiği Şekil 4.13'de verilmiştir.



Şekil 4.13 : DO-CCII devresi için V_x - V_y karakteristiği.

İdeal olarak DO-CCII devresinde $\alpha_1 = \alpha_2 = 1$ ve $\beta = 1$ 'dir. Ayrıca aynı devrede I_{z+} ve I_{z-} akımlarının I_x akımını takip etme başarımlarının gösterildiği karakteristik ise Şekil 4.14'de verilmiştir.



Şekil 4.14 : DO-CCII için I_{z+} ve I_{z-} akımlarının I_x ile değişim karakteristiği.

Karakteristiklerden görüldüğü gibi I_{z+} ve I_{z-} akımları giriş akımını -300 μ A ile 300 μ A aralığında çok iyi takip etmektedir.

Çalışmada sınıflandırma için kullanılacak temel yapılardan biri olan ÇAD devresi, Şekil 4.11'deki yapı kullanılarak Şekil 4.15'de gösterildiği biçimde oluşturulmuştur. Bu yapının akım gerilim bağıntısı aşağıdaki ifadeler ile verilir:

$$I_{z+} = \frac{V_y}{R_1} \frac{R_2}{R_3}$$
(4.5)

$$I_{z-} = -\frac{V_y}{R_1} \frac{R_2}{R_3}$$
(4.6)

Çıkış değişkeni olarak I_{z+} ve I_{z-} akımları kullanılmış ve uygulamalarda (4.2) ifadesinde gösterildiği şekilde I_{out} çıkışı olarak bunlardan işareti arzu edilen alınmıştır.



Şekil 4.15 : ÇAD yapısının DO-CCII yapıları ile gerçeklemesi.

Şekil 4.15'de verilen devrenin simülasyonu için R_1 =25 k Ω , R_2 =25 k Ω ve R_3 =5 k Ω seçilerek V_y ucuna -0.8 V ile 0.8 V arasında değişen gerilim uygulanmış ve R_1 direncinin akımı giriş akımı olarak alınmıştır; dolayısıyla giriş akımı -32 µA ile 32 µA aralığında değiştirilmiştir. $R_2/R_3 = 5$ için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği simülasyon sonucu Şekil 4.16'da verilmiştir.



Şekil 4.16 : ÇAD devresi için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği $(R_2 / R_3 = 5).$

ÇAD devresinde R_2/R_3 oranlarının farklı seçilmesi durumunda giriş-çıkış karakteristiği için yapılan simülasyon sonucu Şekil 4.17'de verilmiştir. Bu simülasyonda $R_2=25$ k Ω alınmış ve R_3 değerleri şekildeki *k* değerlerini sağlayacak biçimde seçilmiştir.



Şekil 4.17 : Çeşitli k= R_2 / R_3 değerleri için ÇAD devresi V_y - I_{z+} karakteristiği.

Önerilen ÇAD devresinin simülasyonunda kullanılan tranzistor boyutları Tablo 4.1'de verilmiştir.

| MOSFET | W [µm] | L [µm] |
|---|--------|--------|
| M_1, M_2, M_9 | 21 | 5.25 |
| M ₃ , M ₄ | 6.3 | 5.25 |
| M ₆ , M ₈ , M ₁₀ , M ₁₁ , M ₁₂ | 11.2 | 5.25 |
| M ₅ , M ₇ , M ₁₃ , M ₁₄ , M ₁₅ | 47.2 | 5.25 |

Tablo 4.1 : ÇAD devresi MOS tranzistorların boyutları.

5. PROTOTİP VE BASKI DEVRE SERİMLERİ; DU-TCC1209 LAB TESTİ

5.1 Çekirdek devrenin serimi

Devre şeması Şekil 2.28'de verilmiş olan yapının serimi MENTOR programı kullanılarak 0.35 µm AMS CMOS teknoloji parametreleriyle, çeşitli denemlerden sonra Şekil 5.1'deki biçimde tasarlanmıştır. Serimi tamamlanan ve yazılımla tüm sınamaları yapılan DU-TCC1209 Integre Devresinin prototipi Fransa'daki Multi-Project Circuit şirketinde üretilmiştir. Çizim sırasında dikkat edilmesi gereken önemli hususlar vardır; en önemlisi devrenin serimi sırasında geçerli olacak fiziksel kuralların iyi bilinmesidir. Bu kurallar Design Rule Check (DRC) adı altında toplanmıştır. Serimin çizimi sırasında bu kurallara uyulması ayrıca devrenin performansını etkileyebilecek durumlara da dikkat edilmesi gerekmektedir. Örnek olarak; gereğinden fazla kapasite yaratmak (poly üstünden metal geçirmek), akım aynalarında eşleşmelere dikkat etmek veya çok büyük boyutlu transistorların parçalı halde oluşturulması gibi. Serim tasarımı bittikten sonra serimin "Calibre" programı yardımı ile çeşitli testlerden geçmesi gerekmektedir. Bunlar şu şekilde sıralanabilir:

- DRC, Design Rule Check: AMS 0.35 µm için belirlenmiş fiziksel kuralların kontrolü,
- LVS, Layout Versus Schematic: Devrenin şematiği ile çizilen serimin birbirlerine denk olmasının kontrolü. Örneğin, bağlantı, net ve pad isimlerinin kontrolü gibi,
- PEX, Parasitic Extraction: Serim sonrası fiziksel yolla oluşabilecek parazitik etkenlerin (direnç ve kapasite gibi) oluşturulması ve devreye etkileri.

Bu üç ana kontrolden sonra serim sonrası simülasyon yapılmaya hazır demektir. Serim sonrası simülasyonu gerçek zamanlı teste en yakın simülasyondur bu yüzden çok önemlidir. Üretim sırasında oluşabilecek yarıiletken etkilerin oluşturacağı üstünlük veya sakıncaların önceden gözlenmesi ve üretim öncesi yapılacak son değişikler bu şekilde belirlenir. Teknoloji parametreleri AMS 0.35 tarafından belirlenmiştir. Çekirdek devrenin seriminin alanı yaklaşık olarak 90x90 (µm)² dir. AMS 0.35, 4 metal kullanma seçeneği sunmaktadır ve bunların her biri farklı katmanlardadır. Akım aynaları iç içe ve birbirine yakın konumlandırılmış ve genel olarak bütün transistorlar parçalı halde birleştirilmiştir. Besleme gerilimleri metal2 (sarı renk)

ile çizilmiştir. Diğer bağlantılar da metal1 kullanılmıştır. Gövdelerin (bulk) beslemelere olan bağlantıları via adı verilen kontaklarla gerçekleştirilmiştir.



Şekil 5.1 : ÇD-2'nin serim çizimi.

5.2 Akım taşıyıcı devresinin Serimi

Akım taşıyıcıları çok çeşitli uygulamalarda kullanılabilirler. Kullanılan akım taşıyıcısının, Şekil 5.2'de görüldüğü gibi, üç girişi ve iki çıkışı vardır. Bu girişlerden V_{bias} olarak gösterileni devredeki kutuplama akımını sağlayacak gerilimin uygulandığı uçtur. Çıkış akımları aynı büyüklükte fakat ters işaretli olacak şekilde, ve ideal çalışma koşullarının sağlanması amacıyla giriş direnci küçük, çıkış direnci büyük değerli olarak tasarlanmıştır. Yer sorunu olmadığından, akım aynalarının eşleşmelerini en iyi şekilde sağlamak amacıyla tranzistörlerin uzunlukları büyük seçilmiştir.



Şekil 5.2 : Akım taşıyıcı şematiği.

MOS tranzistor boyutları aşağıdaki gibi seçilmiştir:

- Tranzistorlerin L boyutları 5.25 µm,
- M3-M4 tranzistorleri 6.3 µm,
- M1-M2-M9 tranzistorleri 21 µm,
- M6-M8-M10-M11-M12 tranzistorleri 11,2 μm,
- M5-M7-M13-M14-M15 tranzistorleri 47,25 μm, .

Serim sonrası simülasyon ile giriş-çıkış dirençleri test edilmiş ve şu sonuçlara ulaşılmıştır; giriş direnci doğru akım çalışmasında 3,3 Ω (10 kHz'de 30 Ω) gibi ideale yakın derecede küçük olmaktadır. Çıkış direnci de 360 k Ω olarak bulunmuştur.

Devrenin serimi ise Şekil 5.3'de gösterilmiştir. Serimin kapladığı alan 100x80 $(\mu m)^2$ dir. Serimden de anlaşılacağı üzere tranzistorler parçalı halde birleştirilmiştir. Serim mümkün olduğunca kare şeklinde tasarlanmaya çalışılmış, minimum boşluk olacak şekilde çizilmiştir. Transistorlerin eşik gerilimlerini ideale yakın tutabilmek için gövde (bulk) ile kaynak (source) arasında ki gerilim farkını en aza indirmek gerekir (idealde 0 V kabul edilir). Bu sebepten gövde ile kaynağın bağlı olduğu besleme gerilimlerinin arasındaki kısa devreyi kuvvetlendirmek için via sayısı mümkün olduğunca fazla tutulmaya çalışılır. Şekil 5.3'deki mavi renkli kutular via'lardır.



Şekil 5.3 : Akım taşıyıcı serimi.

Bazı uygulamalarda kullanılmak üzere özel bir tampon devresi de serime dahil edilmiştir. Tek girişli 3 çıkışlı olan bu tampon devresinin giriş direnci 6.8 k Ω olarak ayarlanmıştır. 3 çıkışlı olmasının sebebi 3 sınıftan oluşan bir veri kümesini sürebilmesi içindir. Çıkış direnci ise beklenildiği gibi çok yüksektir. Bu tampon devresi, akım taşıyıcısının çıkışına bağlanarak, girilen akımın minimum zayıflamayla taşınmasını sağlamak içindir. Kullanıldığı zaman akım taşıyıcısının giriş direnci de 6.8 k Ω olarak ayarlanırsa en iyi sonucu verir. Şekil 5.4'de ve Şekil 5.5'de bu devrenin devre şeması ve serimi verilmektedir.



Şekil 5.4 : Tampon devresi şematiği.



Şekil 5.5 : Tampon devre serimi.

5.3 Prototip devrenin serimi (DU-TCC 1209)

Devrelerin tasarımı, serimi ve serim sonrası simülasyonları yapıldıktan sonra üretim için son aşama olan PAD'lerin (dış dünya ile devreler arasındaki bağlantılar) ve devre bloklarının (çekirdek devre, akım taşıyıcı,ve tampon devre) yerleştirilmesi ele alınmıştır. Şekil 5.6 ve Şekil 5.7'de PAD'ler ve diğer dış elemanlarla beraber devre bloklarının yerleştirilmiş, üretim için hazır hale gelmiş şekli görülmektedir. Şekil 5.6'da gösterilen V_i ve R_i girişleri sırası ile akım taşıyıcının Şekil 4.12'deki Y ve X ucunu göstermektedir. Devre bloklarının giriş ve çıkış uçlarının sayısı çok fazla olduğundan, bununla beraber devre bloklarının kapladığı alan ile PAD'lerin kapladığı alan arasında fark olduğundan çip alanında boşluklar gözükmektedir. Bu tarz çiplere literatürde "PAD LIMITED" denilmektedir. Başka bir değişle tasarlanan çok PAD'lerin kapladığı alan çipe hükmetmektedir. Devre bloklardan blokları yerleştirildikten sonra tekrar DRC kontrolü yapılıp "bonding" (bağlantı) diyagramı oluşturulmuştur. Bu bağlantı diyagramı hangi uçların nereye bağlanacağı bilgisini taşımakta olup üretici firmaya bilgi amaçlı gönderilmiştir. Şekil 5.6'da üretici firmaya gönderilen bu diyagram, Şekil 5.8'de ise üretici firmanın (CMP) son kontrollerinden sonra hazırladığı çip fotoğrafi yer almaktadır.



Şekil 5.6 : DU-TCC1209 Bağlantı Diyagramı (TQFP 52 paketi için).



Şekil 5.7 : Entegre devre'in üretici firmaya gönderilen son hali.



Şekil 5.8 : Üretici firmadan gelen DU-TCC1209 çip fotoğrafi (Die Photo).

Prototip entegre devre'nin içerisinde 9 adet akım taşıyıcı, 3 adet çekirdek devre ve 3 adet tampon devresi vardır. Bir adet akım taşıyıcısı çip içerisinde tampon devrelerden bir tanesine doğrudan bağlıdır. Tampon devrenin de çıkışlarından bir tanesi doğrudan çekirdek devreye bağlanmıştır. Diğer bütün devre bloklarının giriş ve çıkışları dış dünya ile bağlantılıdır. Şekil 5.6'da bu bağlantılar, hangi ucun (pin) nereye karşılık geldiği, gösterilmektedir. DU-TCC 1209 52 bacaklı olup TQFP adı verilen bir paketin içerisinde üretilecektir. Paket 1cm² büyüklüğünde ve çok incedir. Paketin ayrıntılı bilgileri PCB çizimi için gereklidir ve üretici firma tarafından temin edilmiştir. Şekil 5.8'de üretici firma (CMP)'ya son kontroller için

gönderilen tasarım görülmektedir. Üretimden gelen prototip devrenin fotografi Şekil 5.9'da verilmiştir.



Şekil 5.9 : Üretilmiş olan entegre devrenin resmi.

Prototip devrenin testi için Şekil 5.10'da gösterilen baskı devre tasarlanmış ve üretilen devrenin resmi Şekil 5.11'da gösterilmiştir



Şekil 5.10 : Protop devrenin testi için kullanılan baskı devre.



Şekil 5.11 : Baskı devrenin gerçekleştirilmiş şekli.

Şekil 5.11'deki baskı devre kullanılarak çekirdek devrenin giriş çıkış karakteristiğinin testi gerçekleştirilmiştir. Test sonucu, Şekil 2.30'da verilmiş olan SPICE simülasyonu ile uyumlu olarak elde edilmiştir. Test için I_H=80 μ A, I₁=70 μ A ve I₂=170 μ A olacak şekilde ayarlanmış ve akım değerleri devrenin çıkışına 20k Ω direnç bağlanarak akım karakteristiğinin çıktısı osiloskop ile gözlenmiştir. Osiloskopda X ekseni 1.4V/Div ve Y ekseni 0.5V/Div olarak ayarlanmıştır. Test sonucu osiloskop çıktısı Şekil 5.12'de verilmiştir. Baskı devrede gözüken hassas potansiyometreler ayarlanabilir ağırlık katsayılarını veren direnç değerlerini sağlamak üzere konulmuşlardır.



Şekil 5.12 : Giriş Çıkış karakteristiği test sonucu; osiloskop çıktısı.

5.4 Prototip devrenin eğik ızgaralı veri sınıflandırmada testi

Bölüm 3'de eğik ızgaralı verilerin sınıflandırılması incelenmiş ve çekirdek devre yapılarının uygun şekilde bağlanmasıyla bir çok sınıflandırma işleminin yapılabileceği simülasyonlar ile gösterilmişti. Her ne kadar bu yapılar ilerki bölümlerde daha karmaşık sınıflandırıcı uygulamalarında kullanılacaksa da, eğik ızgaralı verilerin sınanması amacıyla, Şekil 5.13'de verilen devre DU-TCC1209 ile kurularak test ölçümleri yapılmıştır.

Bu test düzeneğinde 3 adet çekirdek devre kullanarak kontrol akımları Tablo 5.1'deki gibi verilmiş ve yapının girişine genliği zamanla lineer artan bir işaret uygulanmıştır. Elde edilen çıkış karakteristiği Şekil 5.14'de verilmiştir. Çıkış karakteristiğinden de görülmektedir ki, giriş işareti hangi çekirdek devre sınıflandırıcısının uygun kontrol akım bölgesine düşer ise o bölgeye ait çıkış akımı değeri üretilmektedir. Osiloskobun X ekseni 0.5V/div, Y ekseni ise 0.5V/Div olarak ayarlanmıştır. Test devresinin çıkış akım karakteristiğinin gözlenmesi için çıkışa 10 k Ω 'luk bir direnç bağlanarak osiloskop görüntüsü alınmıştır.



Şekil 5.13 : Eğik ızgaralı veri sınıflandırıcısı test şeması.

| Kontrol akımları | ÇD-1 | ÇD-2 | ÇD-3 |
|------------------|--------|--------|--------|
| I_1 | 25 μΑ | 100 µA | 150 µA |
| I_2 | 50 µA | 125 µA | 175 µA |
| I_H | 200 µA | 100 µA | 150 µA |

Tablo 5.1 : Çekirdek devre kontrol akımları



Şekil 5.14 : Eğik ızgaralı veri sınıflandırıcısı çıkış akım şekli osiloskop çıktısı.

6. SINIFLANDIRMA ALGORİTMALARI VE DEVRELERE UYGULANMASI

6.1 Fisher Tabanlı Algoritma ile Çift Eşik Doğrularının Bulunmalarına Genel Bakış

Sınıflandırılacak veriler 2 boyutlu uzayda, c_1 ve c_2 sınıflarına ait olmak üzere, Şekil 6.1'de gösterildiği gibi verilmiş olsun. Fisher'in lineer diskriminant analizi geliştirilecek ve verilerin ayrıştırılmasına, \vec{v} vektörü ile doğrultusu belirlenen ve orijinden geçen bir doğru üzerine izdüşürülmelerinden yararlanarak uygulanacaktır (QIAN 2007).



Şekil 6.1 : Lineer olarak sınıflandırılamayan örnek veri kümesi.

Bu doğru üzerine izdüşürülen verilerin merkeze (orijine) uzaklıkları göz önüne alınarak Şekil 6.2'de gösterildiği gibi histogramları çizilir; bu şekilde μ_{p1} ve μ_{p2} verilerin orijine olan uzaklıklarının ortalamasını, σ_1 ve σ_2 ise standart sapmalarını göstermektedir.



Şekil 6.2 : Histogram karakteristiği.

Histogram eğrilerinden faydalanılarak iki eşik doğrusu elde edilir. Ancak izdüşüm doğrusu, verilerin en iyi ayrıştırılmasının sağlanması açısından:

a) μ_{p1} ve μ_{p2} birbirinden maksimum uzaklıkta,

b) σ_1 ve σ_2 de olabildiğince küçük olma

koşullarını sağlayacak şekilde seçilmelidir.

6.1.1 Fisher tabanlı algoritma ile çift eşik doğrularının bulunması

 \vec{x} vektörünü

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$
(6.1)

ile verilen iki boyutlu ve iki farklı sınıftan oluşan veri kümesinin bir elemanı olarak alalım; \vec{x} örneklerinin, \vec{v} birim vektörü ile temsil edilen izdüşüm doğrusuna izdüşümleri $\vec{v}^T \vec{x}$ iç çarpımıyla gösterilir. μ_{p1} ve μ_{p2} sırası ile birinci ve ikinci sınıfa ait verilerin orijine olan ortalama izdüşüm uzaklıkları, $\vec{\mu}_1$ ve $\vec{\mu}_2$ ise birinci ve ikinci sınıf veri değerlerinin aritmetik ortalamalarından oluşan vektörler olsun. Bu durumda

$$\mu_{p1} = \vec{v}^T \vec{\mu}_1 \tag{6.2}$$

$$\mu_{p2} = \vec{v}^T \vec{\mu}_2 \tag{6.3}$$

bağıntıları geçerlidir (QIAN 2007).

Ayrıca
$$y = \vec{v}^T \vec{x}$$
(6.4)

ifadesi orijine olan izdüşüm uzaklıklarını gösterdiğine göre, birinci ve ikinci sınıf verileri için sırası ile saçılmalar (standart sapmanın eşdeğeri):

$$s_{p1}^{2} = \sum_{y_{i} \in C_{1}} (y_{i} - \mu_{p1})^{2}$$
(6.5)

$$s_{p2}^{2} = \sum_{y_{i} \in C_{2}} (y_{i} - \mu_{p2})^{2}$$
(6.6)

ifadeleri ile verilir.

Bu ifadelerde verilen y_i 'ler eğitim kümesindeki her bir elemanın orijine olan izdüşüm uzaklığıdır.

Sınıflandırma işleminin en az hata ile gerçekleştirilebilmesi için,

$$J(\vec{v}) = \frac{(\mu_{p1} - \mu_{p2})^2}{(s_{p1} + s_{p2})^2}$$
(6.7)

ifadesinin maksimum kılınması gerekir; dolayısıyla öyle bir \vec{v} bulunmalıdır ki $J(\vec{v})$ ifadesi maksimum olsun.

Birinci ve ikinci sınıf verilerin sınıf içi saçılma matrisleri:

$$S_{1} = \sum_{\vec{x} \in C_{1}} (\vec{x} - \vec{\mu}_{1}) (\vec{x} - \vec{\mu}_{1})^{T}$$
(6.8)

$$S_{2} = \sum_{\vec{x} \in C_{2}} (\vec{x} - \vec{\mu}_{2}) (\vec{x} - \vec{\mu}_{2})^{T}$$
(6.9)

ifadeleri ile tanımlanır ve S_W matrisi

$$S_w = S_1 + S_2$$
 (6.10)

olarak yazılırsa, (6.5) ifadesi ile verilen saçılma terimleri:

$$s_{p1}^{2} = \sum_{y_{i} \in C_{1}} (y_{i} - \mu_{p1})^{2} = \sum_{y_{i} \in C_{1}} (\vec{v}^{T} \vec{x} - \vec{v}^{T} \vec{\mu}_{1})^{2} = \sum_{y_{i} \in C_{1}} (\vec{v}^{T} (\vec{x} - \vec{\mu}_{1}))^{T} (\vec{v}^{T} (\vec{x} - \vec{\mu}_{1}))$$
(6.11)

$$s_{p1}^{2} = \sum_{y_{i} \in C_{1}} \left(\left(\vec{x} - \vec{\mu}_{1} \right)^{T} \vec{v} \right)^{T} \left(\left(\vec{x} - \vec{\mu}_{1} \right)^{T} \vec{v} \right) = \sum_{y_{i} \in C_{1}} \vec{v}^{T} \left(\vec{x} - \vec{\mu}_{1} \right) \left(\vec{x} - \vec{\mu}_{1} \right)^{T} \vec{v} = \vec{v}^{T} S_{1} \vec{v}$$
(6.12)

biçiminde ifade edilebilirler.

Benzer şekilde (6.6) ifadesi de şöyle yazılabilir:

$$s_{p2}^2 = \vec{v}^T S_2 \vec{v}$$
(6.13)

Bu durumda saçılmaların toplamı

$$S_{p1}^{2} + S_{p2}^{2} = \vec{v}^{T} S_{1} \vec{v} + \vec{v}^{T} S_{2} \vec{v} = \vec{v}^{T} S_{w} \vec{v}$$
(6.14)

olur.

İki sınıfın ortalamaları arasındaki ayrılabilirliğin bir ölçütü olan sınıflar arası saçılma matrisi

$$S_B = (\vec{\mu}_1 - \vec{\mu}_2)(\vec{\mu}_1 - \vec{\mu}_2)^T$$
(6.15)

ifadesi ile verilir.

Diğer taraftan (6.7) ifadesinin pay terimi aşağıdaki şekilde yazılır:

$$(\mu_{p1} - \mu_{p2})^2 = (\vec{v}^T \vec{\mu}_1 - \vec{v}^T \vec{\mu}_2)^2 = \vec{v}^T (\vec{\mu}_1 - \vec{\mu}_2) (\vec{\mu}_1 - \vec{\mu}_2)^T \vec{v} = \vec{v}^T S_B \vec{v}$$
(6.16)

Dolayısıyla (6.14) ve (6.16) eşitlikleri kullanılarak oluşturulmuş

$$J(\vec{v}) = \frac{(\mu_{p1} - \mu_{p2})^2}{(S_{p1}^2 + S_{p2}^2)} = \frac{\vec{v}^T S_B \vec{v}}{\vec{v}^T S_w \vec{v}}$$
(6.17)

ifadesinden görüldüğü gibi sınıflandırma işleminin en uygun şekilde yapılması sınıflar arası saçılma matrisine (S_B) ve sınıf içi saçılma matrisine (S_W) bağlı çıkmaktadır.

Böylece problem $J(\vec{v})$ ölçütünü maksimum kılan \vec{v} vektörünü, yani izdüşüm doğrusunu bulmak şeklinde ifade edilebilir hale gelir. Bunun için ölçütün gradyenini sıfıra eşitleyen

$$\frac{\partial}{\partial v}J(\vec{v}) = \frac{\left(\frac{\partial}{\partial v}\vec{v}^T S_B \vec{v}\right)\vec{v}^T S_w \vec{v} - \left(\frac{\partial}{\partial v}\vec{v}^T S_w \vec{v}\right)\vec{v}^T S_B \vec{v}}{\left(\vec{v}^T S_w \vec{v}\right)^2} = 0$$
(6.18)

ifadesi kullanılarak

$$\frac{\partial}{\partial v}J(\vec{v}) = \frac{(2S_B\vec{v})\vec{v}^T S_w\vec{v} - (2^T S_w\vec{v})\vec{v}^T S_B\vec{v}}{\left(\vec{v}^T S_w\vec{v}\right)^2} = 0$$
(6.19)

$$\vec{v}^{T} S_{w} \vec{v} (S_{B} \vec{v}) - \vec{v}^{T} S_{B} \vec{v} (S_{w} \vec{v}) = 0$$
(6.20)

$$\frac{\vec{v}^{T} S_{w} \vec{v} (S_{B} \vec{v})}{\vec{v}^{T} S_{w} \vec{v}} - \frac{\vec{v}^{T} S_{B} \vec{v} (S_{w} \vec{v})}{\vec{v}^{T} S_{w} \vec{v}} = 0$$
(6.21)

ara işlemlerinden sonra

$$(S_B\vec{v}) - \frac{\vec{v}^T S_B \vec{v} (S_w \vec{v})}{\vec{v}^T S_w \vec{v}} = (S_B \vec{v}) - \lambda (S_w \vec{v}) = 0$$
(6.22)

eşitliği elde edilir.

(6.22) ifadesindeki denklemin çözümü ise genelleştirilmiş özdeğer probleminden başka bir şey değildir. Bu özdeğer problemi

$$S_B \vec{v} = \lambda S_w \vec{v} \tag{6.23}$$

denklemi ile verilir.

Bu denklemin çözümünden bulunacak olan maksimum değere sahip özdeğere karşılık gelen \vec{v} vektörü verilerin izdüşürüleceği doğrunun eğimini vermektedir (QI, 1997). Bu birim vektör Şekil 6.3'de gösterilen izdüşüm doğrusu ile aynı yöndedir (QIAN, 2007).

Şekil 6.3'de açık renkli olarak çizilmiş histogram eğrisi birinci sınıfa, koyu renkli olarak çizilmiş histogram eğrisi ikinci sınıfa ait veriler olsun; bu durumda \vec{v} vektörü bulunduktan sonra histogram eğrileri kullanılarak *a* ve *b* noktaları bulunur. Burada koordinatları (x_{a1} , x_{a2}) olan *a* noktası, izdüşüm doğrusu üzerinde ikinci sınıfa ait olabilecek son noktadır; başka bir deyişle, izdüşüm doğrusu üzerinde ikinci sınıfa ait verilerin orijine uzaklığı en küçük olan noktadır.



Şekil 6.3 : Çift eşik doğrularının gösterilimi.

Benzer şekilde koordinatları (x_{b1} , x_{b2}) olan *b* noktası ise birinci sınıfa ait verilerde orijine olan izdüşüm uzaklığı maksimum olan değerdir. Bu durumda verilerin ayırt edilmesini sağlayacak doğruların (hiperdüzlemlerin) denklemleri ise aşağıdaki ifadeler ile verilir:

$$\vec{v}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \vec{v}^T \begin{pmatrix} x_{a1} \\ x_{a2} \end{pmatrix} = 0$$
(6.24)

$$\vec{v}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \vec{v}^T \begin{pmatrix} x_{b1} \\ x_{b2} \end{pmatrix} = 0$$
(6.25)

Bu denklemler izdüşüm doğrusu üzerinde bulunan a ve b noktalarına dik çıkan Şekil 6.3'de kesik çizgiler ile gösterilmiş doğru denklemleridir. Bu doğru denklemleri bulunduktan sonra a ile b noktaları arasında kalan veri kümesi için aynı algoritma tekrar uygulanır ve benzer doğrular arada kalan veri kümesi için tekrar hesaplanır. Verileri kesin olarak ayrıştıran doğrular bulunduktan sonra, yöntem durdurulur. Bu (6.24) ve (6.25) ifadeleri ile bulunmuş olan denklemler çift eşik doğrularıdır.

6.1.2 Fisher tabanlı algoritma ile çift eşik doğrularının genel halde bulunması

Çift eşik doğrularının çizilmesi için anlatım kolaylığı açısından iki boyutlu ve iki sınıftan oluşan veri kümeleri kullanıldı. Çift eşik doğrularının çizilmesinde genelleştirilmiş durum, $d \ge c$ koşulu altında d boyutlu c adet farklı sınıf olsun. c=2 olması durumu için Altbölüm 5.1.1'de anlatılan yöntem hiç bir değişikliğe uğramaksızın uygulanır; aksi halde (c-1) adet birim izdüşüm vektörü bulunur ve bunlarla

$$V = \begin{bmatrix} \vec{v}_1 | \vec{v}_2 | \dots | \vec{v}_{c-1} \end{bmatrix}$$
(6.26)

matrisi oluşturulur.

Bu durumda izdüşüm vektörlerinin bulunması için öncelikle genelleştirilmiş haldeki sınıf içi saçılma matrisi (S_W) ve sınıflar arası saçılma matrisi (S_B) elde edilmelidir. n_i , *i*. sınıfa ait eğitim kümesindeki veri sayısı ise, S_W ve S_i matrisleri

$$S_w = \sum_{i=1}^c S_i$$
(6.27)

$$S_{i} = \sum_{x \in C_{i}} (\vec{x} - \vec{\mu}_{i}) (\vec{x} - \vec{\mu}_{i})^{T}$$
(6.28)

ifadeleri ile, $\vec{\mu}_i$ ortalaması da

$$\vec{\mu}_i = \frac{1}{n_i} \sum_{x \in C_i} \vec{x}$$
(6.29)

eşitliği ile verilir. Dolayısıyla

$$S_{w} = \sum_{i=1}^{c} \sum_{C_{i}} (\vec{x} - \vec{\mu}_{i}) (\vec{x} - \vec{\mu}_{i})^{T}$$
(6.30)

olur.

(6.28) ifadesinde bulunan $\vec{\mu}_i$, *i*. sınıfa ait verilerin ortalaması, *n* eğitim kümesindeki toplam veri sayısıdır. Tüm verilerin ortalaması olan $\vec{\mu}$ vektörü,

$$\vec{\mu} = \frac{1}{n} \sum_{\forall x} \vec{x}$$
(6.31)

ifadesi ile verilir.

Sınıflar arası saçılma matrisi S_B aşağıdaki eşitlikteki gibidir:

$$S_{B} = \sum_{i=1}^{c} n_{i} (\vec{\mu}_{i} - \vec{\mu}) (\vec{\mu}_{i} - \vec{\mu})^{T}$$
(6.32)

 $\vec{\mu}_{pi}$, *i*. sınıfa ait verilerin ortalama izdüşüm uzaklığından oluşan vektör, $\vec{\mu}_p$ ise $\vec{\mu}_{pi}$ 'lerin ortalamasıdır. Dolayısıyla

$$\vec{\mu}_p = \frac{1}{n} \sum_{\forall y} y \tag{6.33}$$

olarak yazılabilir. Buradaki y orijine olan uzaklığı göstermektedir.

Sınıf içi saçılma matrisi (S_{PW}) ve izdüşürülmüş sınıflar arası saçılma matrisi (S_{PB}) sırası ile

$$S_{PW} = \sum_{i=1}^{c} \sum_{y \in C} (\vec{y} - \vec{\mu}_{pi}) (\vec{y} - \vec{\mu}_{pi})^{T}$$
(6.34)

olmak üzere,

$$S_{PW} = V^T S_W V \tag{6.35}$$

ile, ve

$$S_{PB} = \sum_{i=1}^{c} n_i (\vec{\mu}_{pi} - \vec{\mu}_p) (\vec{\mu}_{pi} - \vec{\mu}_p)^T$$
(6.36)

$$S_{PB} = V^T S_B V \tag{6.37}$$

ifadeleriyle verilir.

Sınıflandırma işleminin optimum şekilde yapılması için, izdüşürülmüş sınıflar arası saçılma matrisi (S_{PB}) ile izdüşürülmüş sınıf içi saçılma matrisi (S_{PW}) skaler olmadıklarından, pay ve payda terimlerinin determinant olduğu

$$J(V) = \frac{\det(V^T S_B V)}{\det(V^T S_w V)}$$
(6.38)

biçiminde J(V) ölçütü ile tanımlanır. Bu ölçüt maksimize edilerek en iyi ayıran hiperdüzlemler bulunur (JIANG 1999). (6.38) ifadesinin maksimum kılınması, aşağıda verilen

$$(S_B - \lambda_i S_w) \vec{v}_i = 0 \tag{6.39}$$

genelleştirilmiş özdeğer probleminin çözümüne indirgenir (ÇEVİKALP 2006). S_B matrisi için maksimum(rank(S_B))=*c*-1 olmaktadır, ispatı Ek B'de verilmiştir. **(6.39)** ifadesi ile verilen özdeğer probleminin özdeğerleri ve karşı düşen özvektörleri \vec{v}_i , *V* izdüşüm matrisinin sütunlarını oluştururlar. Bu matrisler reel ve yarı kesin pozitif olduklarından özdeğerleri de pozitif reeldir ve maksimum değere sahip olanın özvektörü, verilerin sınıflandırılmasında en iyi ayrımı sağlamaktadır (QIAN 2007). Birden fazla sayıda ve birbirine eşit maksimum özdeğer çıkması durumunda, bu maksimum özdeğerlerden elde edilen özvektörlerden herhangi biri seçilip veri sınıflandırılmasında kullanılabilir (farklı özvektörlerin hepsi veriler arasında aynı derecede ayırım sağlamaktadır (ÇEVİKALP 2006)). Maksimum özdeğere karşı düşen izdüşüm vektörü elde edildikten sonra sınıflandırıma devresi ile kullanılması Bölüm 5.3'de ele alınmıştır.

6.2 Eğiticili Perseptron Öğrenme Algoritması ile ÇAD ve ÇD Parametrelerinin Bulunması

Çift eşik hiperdüzlemlerinin oluşturulması için kullanılabilecek bir yöntem de yapay sinir ağlarında kullanılan eğiticili perseptron öğrenme algoritmasıdır (AKSIN 2000). Bölüm 3'de ÇAD ve ÇD yapıları kullanılarak eğik ızgaralı verilerin, çift eşik hiperdüzlemi ile nasıl sınıflandırılacağı incelenmişti. Çift eşik hiperdüzlemi belirleyen katsayılar tek katlı perseptron öğrenme algoritmasından faydalanılarak da elde edilebilmektedir. Dolayısıyla perseptron öğrenme algoritması da Bölüm 3'de ele alınan verilerin sınıflandırılması için kullanılabilir; Şekil 6.4'de *n* girişli tek katlı perseptron yapısı verilmiştir.



Şekil 6.4 : *n* girişli tek katlı perseptron yapısı.

Şekil 6.4 ile verilen bu yapının tanım bağıntısı da şöyledir:

$$y = \begin{cases} 1 & \sum_{i=1}^{n} x_i w_i \ge k & i \ cin \\ 0 & \sum_{i=1}^{n} x_i w_i < k & i \ cin \end{cases}$$
(6.40)

Sınıflandırmada kullanılacak katsayılar (6.40) eşitliğindeki w ve k değerleridir. Bu w ve k değerleri ile hiperdüzlem denklemleri oluşturulur. Sınıflandırma için gerekli olan çift eşik hiperdüzlem denklemleri her farklı sınıf için aşağıdaki ifadeler kullanılarak elde edilir:

$$y_{i} = \begin{cases} 1 & \vec{x} \, \vec{v}_{i}^{T} - a \ge 0 \\ 0 & \vec{x} \, \vec{v}_{i}^{T} - a < 0 \end{cases}$$
(6.41)

$$y_{i} = \begin{cases} 1 & \vec{x} \, \vec{v}_{i}^{T} - b \le 0 \\ 0 & \vec{x} \, \vec{v}_{i}^{T} - b > 0 \end{cases}$$
(6.42)

(6.41) ve (6.42) ifadelerinde kullanılan \vec{v} vektörü, *a* ve *b* parametrelerinin değerleri perseptron öğrenme algoritmasına göre hesaplanır.

Öğrenme algoritması sonucu, hiperdüzlem denklemleri

$$\vec{x}\,\vec{v}_i^T - a = 0 \tag{6.43}$$

$$\vec{x}\,\vec{v}_i^T - b = 0 \tag{6.44}$$

biçiminde yazılabilir.

Bu yöntemde veri, sınıflara uygun sayıda çift eşik hiperdüzlemleri ile ayrıştırılarak sınıflandırılmaktadır.

 c_i (i=1,2,..,k) veri sınıfları için uygun sayıda çift eşik hiperdüzlem denkleminin bulunması aşağıdaki yöntemi adım uygulayarak yapılır:

Çift eşikli Perseptron Algoritması

- 1. Herhangi bir c_i sınıfını diğer tüm sınıflardan ayıran uygun bir adet çift eşik hiperdüzlemi olup olmadığı (5.45-5.47) ile güncelleme kuralları verilen perseptron öğrenme algoritmasıyla kontrol edilir.
 - i. Eğer varsa, *c_i* sınıfı için ağırlık katsayıları kaydedilir, ikinci adıma geçilir,
 - ii. Eğer bulunamıyorsa üçüncü adıma geçilir.
- 2. Bir sonraki sınıfa geçilir.
 - i. Eğer bir sonraki sınıf sonuncu *k* sınıfı ise sınıflandırma işlemi sonlandırılır,
 - ii. Değil ise birinci adıma dönülür.
- 3. Çift eşik hiperdüzlem sayısı bir arttırılır ve c_i sınıfının diğer sınıflardan ayrılıp ayrılmadığı aynı algoritmayla kontrol edilir.
 - Ayrılabiliyor ise çift eşik hiperdüzlemlerinin ağırlık katsayıları kaydedilir,.
 ikinci adıma geçilir,
 - ii. Ayrılamıyor ise üçüncü adım tekrarlanır.

Perseptron yapısında kullanılan aktivasyon fonksiyonu keskin eşik fonksiyonu biçiminde olduğu için ağırlık katsayıları perseptron öğrenme algoritmasına göre güncellenerek elde edilir (ZHAO 2002).

Perseptron öğrenme algoritması,

$$v_i(n+1) = v_i(n) - \eta (y_d - y_o) x_i$$
(6.45)

$$a_i(n+1) = a_i(n) - \eta (y_d - y_o)$$
(6.46)

$$b_i(n+1) = b_i(n) - \eta (y_d - y_o)$$
(6.47)

biçimindedir buradaki y_d elde edilmek istenilen sonuç, y_o ise elde edilen sonuç değeridir. η ise öğrenme katsayısı olarak isimlendirilir; ve 0 ile 1 arasında değer alır. Güncelleme işleminde $y_d = y_o$ olması durumunda ağırlık katsayısı değişmez. Öğrenme algoritması bütün ağırlık katsayıları değişmediği durumda durur (GENÇ 1998).

Eğitim kümesindeki verilerden c_i sınıfına ait veriler kullanılarak güncelleme işlemi bittikten sonra (6.41) ve (6.42) ifadeleri kullanılarak bulunmuş olan çift eşik hiperdüzlemi, $a_i \leq \vec{x} \vec{v}_i^T \leq b_i$ bölgesini göstermektedir. Öğrenme algoritması ile bulunan \vec{v} vektörü ÇAD parametreleri olan w'lara (ağırlık katsayıları), a ve b ise çekirdek devrenin I_1 , I_2 (kontrol akımları) değerlerine karşılık gelmektedir ve ÇD'nin I_H parametresinden yararlanılarak veri sınıfı belirlenmektedir. Şayet veri kümesindeki bir sınıf, iki veya daha çok sayıda çift eşik hiperdüzlemi gerektirirse, diğer bir deyişle iki veya daha çok sayıda bölgede aynı sınıfın olması durumunda, farklı bölgelere karşılık düşen aynı verilerin ÇD'lerinin I_H parametreleri aynı seçilerek veri sınıfının doğru sınıflanması sağlanır. Dolayısıyla uygun sayıda ÇAD ve ÇD bloğu kullanılarak sınıflandırma yapısı oluşturulur. Perseptron öğrenme algoritması kullanılarak üç ve dört boyutlu iki farklı veri kümesi Bölüm 6'da önce yazılımla sınıflandırılmış, sonra da DU-TCC1209'ye uygulanarak,sonuçları verilmiştir.

7. SINIFLANDIRICI DEVRE UYGULAMALARI

7.1 İris ve Haberman Verilerinin Sınıflandırılması

7.1.1 İris verisinin Fisher tabanlı algoritma ile sınıflandırılması

Literatürde çok kullanılan veri kümelerinden biri iris veri kümesidir. Bu veri kümesi 1936 yılında Fisher tarafından oluşturulmuştur (FISHER, 1936). Sınıflandırma uygulamalarında kullanılan en temel veri sınıflarındandır. Veri kümesi dört adet özellik, diğer bir ifadeyle dört boyutlu veri (x_1, x_2, x_3, x_4) ve üç sınıftan (c_1, c_2, c_3) oluşmaktadır. Bu sınıflar Setosa, Versicolor ve Virginica isimli bitki tipleridir. Veri kümesinin sınıflandırılması bitkinin taç ve çanak yapraklarının uzunluk ve kalınlık değerlerine bakılarak gerçekleştirilir. İris veri kümesi 150 adet veri içermektedir. Bu veriler üç sınıfa eşit sayıda dağılmıştır. Veri kümesi detaylı olarak Ek C1'de verilmiştir. Sınıflandırma için kullanılacak veri kümesinin özellikleri santimetre birimi cinsinden verilmiş ve gerçekleştirilmiş olan sınıflandırıcı ile test edilmiştir. Bu testte sınıflandırıcının kontrol parametrelerinin, Bölüm 4.1'de geliştirilmiş olan Fisher tabanlı algoritma ile hesaplanması için, her sınıftan 40 veri ve toplamda 120 veri kullanılmıştır. Fisher tabanlı algoritma sonucunda veriler arasında en iyi ayrımı sağlayacak izdüşüm vektörü $\vec{v} = (0.57 - 0.80 \ 0.10 \ 0.14)$ olarak elde edilmiştir. Bu vektör (6.4) ifadesinde belirtildiği şekilde kullanılarak iris veri kümesinin orijine olan izdüşüm uzaklığı hesaplanarak verilerin orijine olan izdüşüm uzaklıkları Şekil 7.1'de gösterilmiştir. Bu şekil üstündeki, yıldız işaretleri c_1 , artı işaretleri c_2 ve küçük daireler de c_3 sınıfına ait verilerdir. Şekil 7.1'den görüldüğü gibi yatay eksen, verilerin orijine olan izdüşüm uzaklıklarını ve dikey eksen ise veri numarasını göstermektedir. Dolayısıyla Şekil 7.1'deki verilerin uzaklıkları ile iris veri kümesi sınıflandırılabilmektedir. Bu sınıflandırma işlemi için ÇAD ve ÇD yapılarından faydalanılmıştır. Şekil 7.2'de iris verisi sınıflandırıcısı blok diyagramı verilmiştir. Bu yapıdaki her bir ÇAD yapısının ağırlık katsayısı w, Fisher tabanlı algoritma ile elde edilen \vec{v} vektörünün elemanlarına denk düşmektedir. ÇAD yapılarının çıkış değişkeni akım boyutunda olduğundan çıkış uçları birbirlerine bağlanarak toplama işlemi gerçeklenir. Bu toplam, veri kümesinin orijine olan izdüşüm uzaklığını gösterir. Dolayısıyla sınıflandırma işlemi için izdüşüm uzaklıkları (akım değerleri) üç çıkışlı Akım Çoğullayıcı (AÇ) devre ile çekirdek devre girişlerine uygulanmıştır. Şekil 7.1'de gösterildiği biçimde, her bir sınıfın ayrıldığı noktalar çekirdek devrenin kontrol akım değerlerine karşılık gelmektedir.



Şekil 7.1 : İzdüşürülmüş iris verilerinin orijine uzaklıkları.



Şekil 7.2 : İris verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma ile oluşturulmuş).

Şekil 7.2'deki yapının gerçekleştirilmesi için kullanılan akım çoğullayıcının n çıkışlı devre şeması Şekil 7.3'de ve tanım bağıntısı da (7.1) ifadesi ile verilmiştir.

Akım çoğullayıcı sayesinde, ÇAD akımlarının toplamı tekrar çekirdek devrelere uygulanarak, Şekil 7.1'de gösterilen bölgelere ait veriler, çıkışın üç farklı değerden birini almasıyla seçilir.

$$\begin{bmatrix} V_{x} \\ I_{z-1} \\ I_{z-2} \\ I_{z-3} \\ \vdots \\ I_{z-n} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & . & 0 \\ 1 & 0 & 0 & 0 & . & 0 \\ 1 & 0 & 0 & 0 & . & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & . & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & . & 0 \end{bmatrix} \begin{bmatrix} I_{x} \\ V_{z-1} \\ V_{z-2} \\ V_{z-3} \\ \vdots \\ V_{z-n} \end{bmatrix}$$
(7.1)

Şekil 7.2'deki devrenin simülasyonunda kullanılan n çıkışlı akım çoğullayıcı yapı için 0.35 μ m AMS CMOS teknoloji parametreleri ve besleme gerilimleri V_{DD} ve $V_{SS} \pm 1.65$ V olarak kullanılmıştır. Kutuplama akımının 18 μ A olması için V_B gerilimi 0.8 V olarak alınmıştır. Akım çoğullayıcı devresinde kullanılan tranzistor boyutları da Tablo 7.1'de verilmiştir.



Şekil 7.3 : Akım çoğullayıcı devre şeması (FERRI, 2001).

| Tablo 7.1 : <i>n</i> (| çıkışlı akım | çoğullayıcı | devresi MOS | tranzistorların | boyutları. |
|------------------------|--------------|-------------|-------------|-----------------|------------|
| | , , | , 0 , | | | 2 |

| MOSFET | W [µm] | L [µm] |
|--|--------|--------|
| M_1, M_3, M_4 | 21 | 1.05 |
| M_5, M_6 | 6.3 | 1.05 |
| M ₂ , M _{a-1} , M _{a-2} ,, M _{a-n} | 42.7 | 1.05 |
| M ₇ , M _{b-1} , M _{b-2} ,, M _{b-n} | 11.2 | 1.05 |

Sınıflandırıcının başarımı için her sınıftaki test kümesinden geri kalan 10, toplamda 30 adet veri kullanılarak Şekil 7.2'de gösterilen yapının SPICE simülasyonu yapılmıştır. Simülasyon için kullanılan test verileri Tablo 7.2'de gösterilen aralıklarla Şekil 7.2'deki yapıya uygulanmıştır.

| Verilerin uygulanma süresi | x_1 | x_2 | <i>x</i> ₃ | x_4 | Sınıf |
|----------------------------|-------|-------|-----------------------|-------|-------|
| 0 ms - 1 ms | 4.3 | 2.3 | 1.4 | 0.2 | c_1 |
| 1 ms - 2 ms | 5.7 | 2.7 | 3.9 | 1.1 | c_2 |
| 2 ms - 3 ms | 5.7 | 2.7 | 4.8 | 1.8 | c_2 |
| 3 ms - 4 ms | 4.9 | 2.2 | 6 | 2.5 | c_3 |
| 4 ms - 5 ms | 5.6 | 2.5 | 5.1 | 1.9 | c_3 |
| 5 ms - 6 ms | 4.6 | 3 | 1.7 | 0.4 | c_1 |
| 6 ms - 7 ms | 4.7 | 3.1 | 1.5 | 0.1 | c_1 |
| 7 ms - 8 ms | 6.1 | 2.9 | 3.8 | 1.1 | c_2 |
| 8 ms - 9 ms | 6.4 | 3 | 4 | 1.3 | c_2 |
| 9 ms - 10 ms | 4.9 | 2 | 4.7 | 1.4 | c_2 |
| 10 ms - 11 ms | 4.8 | 3.2 | 1.2 | 0.2 | c_1 |
| 11 ms - 12 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 12 ms - 13 ms | 6.3 | 2.8 | 6.5 | 1.8 | c_3 |
| 13 ms - 14 ms | 6.7 | 3 | 6.4 | 2 | c_3 |
| 14 ms - 15 ms | 7.2 | 3.2 | 5.4 | 2.1 | c_3 |
| 15 ms -16 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |
| 16 ms - 17 ms | 5.3 | 3.7 | 1.5 | 0.2 | c_1 |
| 17 ms - 18 ms | 5.4 | 3.8 | 1.3 | 0.3 | c_1 |
| 18 ms - 19 ms | 6.3 | 3.0 | 4.4 | 1.3 | c_2 |
| 19 ms - 20 ms | 6.3 | 3.0 | 4.1 | 1.3 | c_2 |
| 20 ms - 21 ms | 6.9 | 3.1 | 5.6 | 2.4 | c_3 |
| 21 ms - 22 ms | 7.6 | 3.3 | 5.7 | 2.5 | c_3 |
| 22 ms - 23 ms | 7.7 | 3.4 | 5.2 | 2.3 | c_3 |
| 23 ms - 24 ms | 6.4 | 3.0 | 4.0 | 1.3 | c_2 |
| 24 ms - 25 ms | 6.4 | 3.0 | 4.4 | 1.2 | c_2 |
| 25 ms - 26 ms | 6.5 | 3.0 | 4.0 | 1.2 | c_2 |
| 26 ms - 27 ms | 7.7 | 3.4 | 5.0 | 1.9 | c_3 |
| 27 ms - 28 ms | 7.7 | 3.6 | 5.2 | 2.0 | c_3 |
| 28 ms - 29 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 29 ms - 30 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |

Tablo 7.2 : Şekil 7.2'deki iris verisi sınıflandırıcısı test kümesi.

Tablo 7.2'de verilmiş olan iris verisinin özellikleri olan x_i (i=1,2,...4) değerleri cm boyutundadır. Ancak simülasyon için bu değerler ÇAD sayesinde gerekli normalizasyon yapılarak akıma dönüştürülmüş ve çekirdek devreye uygulanmıştır. Tablo 7.2'deki 0-1 ms arasındaki x_1 =4.3 verisinin ÇAD-1 yapısında kullanılması şöyle özetlenebilir: Şekil 4.15'deki yapıda V_y =0.43 V ve R_1 =10 k Ω alınması durumunda giriş büyüklüğü V_y/R_1 = 43 µA olarak akıma çevrilir. Ayrıca Fisher tabanlı algoritma ile bulunan $\vec{v} = (0.57 - 0.80 \ 0.10 \ 0.14)$ vektörünün ilk elemanı 0.57 olduğu için R_2/R_3 = 57 k Ω / 100 k Ω alınarak (4.5) ifadesine göre çıkış akımı elde edilir. Bu çıkış değeri elde edilirken x_1 =4.3 olan giriş verisi µA boyutunda gibi düşünülüp 10 kat büyütülerek 43 µA olarak \vec{v} vektörünün elemanı ile çarpılmıştır. Benzer durum diğer giriş verileri içinde uygulandığından, orijine olan izdüşüm uzaklığı Şekil 7.1'de gösterilmiş olan uzaklığın 10 katı olarak elde edilmiştir. Bu durumda Şekil 7.1'de c_1 sınıfına ait verilerin orijine olan izdüşüm uzaklığı 0.1 ile 0.8 arasında olduğu halde verilerin ayırt edilebilmesi için Şekil 7.2'deki çekirdek devre-1'in kontrol akım değerleri I_1 =1 µA, I_2 =8 µA olacak şekilde seçilmiştir. Şekil 7.2'de kullanılan diğer ÇAD ve ÇD parametreleri benzer şekilde bulunmuş ve sırası ile Tablo 7.3'de ve Tablo 7.4'de verilmiştir. Ayrıca Şekil 7.2'de kullanılan ÇAD çıkışları \vec{v} vektörünün elemanlarının pozitif ya da negatif olmasına göre seçildiğinden, Şekil 4.15'den farklı olarak tek çıkışları bulunmaktadır. Diğer bir ifadeyle \vec{v} vektörünün ilk elemanı pozitif, ikinci elemanı negatiftir, dolayısıyla Şekil 7.2'de kullanılan ÇAD-1 yapısı için pozitif çıkış ucu, ÇAD-2 yapısı için negatif çıkış ucu kullanılmıştır.

Tablo 7.3: İris sınıflandırıcısı ÇAD direnç değerleri.

| Dirençler | ÇAD-1 | ÇAD-2 | ÇAD-3 | ÇAD-4 |
|-----------|--------|-----------------------|--------|--------|
| R_1 | 10 kΩ | 10 kΩ | 10 kΩ | 10 kΩ |
| R_2 | 57 kΩ | $80 \text{ k}\Omega$ | 10 kΩ | 14 kΩ |
| R_3 | 100 kΩ | $100 \text{ k}\Omega$ | 100 kΩ | 100 kΩ |

Şekil 7.2'deki yapının çıkış akımı çekirdek devrelerin I_H akımları ile kontrol edildiğinden her çekirdek devrenin I_H akımı farklı seçilerek veri kümesinin sınıflandırılması sağlanır. Ayrıca iris kümesindeki veriler sınıflandırıcıya uygulanmadan önce sınıflandırıcının uygun çalışma bölgesi göz önüne alınarak 10 kat büyütülmüştür. Böylece ÇAD yapısı ile sınıflandırıcı devrenin çalışma bölgesi dışındaki veri kümelerinin de ayırt edilebildiği gösterilmiştir.

| Kontrol akımları | ÇD-1 | ÇD-2 | ÇD-3 |
|------------------|-------|-------|-------|
| I_1 | 1 μΑ | 16 µA | 21 µA |
| I_2 | 8 μΑ | 19 µA | 27 μΑ |
| I_H | 10 µA | 20 µA | 30 µA |

Tablo 7.4: İris sınıflandırıcısı çekirdek devre kontrol akımları.

Tablo 7.2'de sınıflandırıcı sonuçlarının ne çıkması gerektiği gözükmektedir. Şekil 7.2'deki yapının simülasyon sonucu Şekil 7.4'de verilmiştir; buna göre simülasyon sonucunda çıkış değerinin 10 μ A olması durumunda sonuç c_1 , 20 μ A olması durumunda sonuç c_2 , 30 μ A olması durumunda sonuç c_3 olmaktadır.



Şekil 7.4 : Şekil 7.2'deki İris verisi sınıflandırıcısı simülasyon sonucu.

İris verisinin DU-TCC 1209 kullanılarak Fisher tabanlı algoritma ile testi için Şekil 7.2'deki blok yapısının deney düzeneği kurulmuştur. Bu düzenek kurulurken simulasyonlarda kullanılan ÇAD yapısından farklı olarak (test düzeneğini kurulum kolaylığı için) tek bir CCII elemanı kullanılmıştır. Bu ÇAD yapısında ise simulasyon için elde edilen direnç değerleri alınmış, ancak $\frac{V_y}{R_1} \frac{R_2}{R_3}$ ifadesindeki dirençler için $\frac{1}{R_1} \frac{R_2}{R_3}$ nin eşdeğer direnci kullanılmıştır. Eşdeğer direnç değerleri Tablo 7.5'de verilmiştir.

Tablo 7.5: İris sınıflandırıcısı ÇAD eşdeğer direnç değerleri.

| Dirençler | ÇAD-1 | ÇAD-2 | ÇAD-3 | ÇAD-4 |
|-----------|---------|---------|--------|---------|
| R | 17.5 kΩ | 12.5 kΩ | 100 kΩ | 71.4 kΩ |

DU-TCC 1209'un sınanması için değişken işaretler üretebilen bir işaret üreteci kullanılarak simülasyondakine benzer şekilde 1ms aralıklarla uygun giriş gerilimleri direnç üzerinden uygulanmıştır. Dolayısıyla sınıflandırıcı yapısına uygun akım değerlerinin girilmesi sağlanmıştır. Simülasyonlarda belirtilen kontrol akım değerleri de uygun dirençler sayesinde ayarlanmıştır. Sınıflandırıcı yapısının çıkışına 10k Ω 'luk direnç bağlanılarak çıkış akımı ölçülmüş ve çıkış geriliminin değişim karakteristiği Şekil 7.5'de verilmiştir. Osiloskopda X ekseni 3ms/Div ve Y ekseni 0.1V/Div olarak ayarlanmıştır.



Şekil 7.5 : İris verisi test sonucu osiloskop çıktısı.

Fisher tabanlı algoritma ile iris verisinin DU-TCC 1209 ile sınıflandırılması için laboratuarda kurulan test düzeneğinin resmi Şekil 7.6'da gösterilmiştir.



Şekil 7.6 : İris verisi test düzeneği.

Test sonucunda elde edilen osiloskop çıktısı ile Şekil 7.4'de simulasyon sonuçları uyum içindedir. Dolayısıyla görüldüğü gibi 30 adet iris test verisi doğru olarak sınıflandırılmıştır ve bu veri için Bölüm 4.1'de verilen Fisher tabanlı algoritma kullanılarak veri kümesinin sınıflandırılabilmesi sağlanmıştır.

7.1.2 Haberman verisinin Fisher tabanlı algoritma ile sınıflandırılması

Literatürde, sınıflandırmada çok kullanılan diğer bir veri kümesi de Haberman veri kümesidir (HABERMAN 1967). Bu veri kümesi üç boyutlu (x_1, x_2, x_3 ,) iki sınıftan (c_1, c_2) oluşmaktadır. Bu küme Haberman tarafından Şikago Üniversitesi Billing Hastanesinde göğüs kanseri teşhisi ile ameliyat olmuş hastalardan yararlanılarak 1958-1970 yılları arasında oluşturulmuştur.

Küme, ameliyat sonrasında beş yıldan daha uzun yaşayan ya da beş yıl içinde ölen hastaları ayıırmaktadır. Sınıflandırma işlemi için hastanın yaşı, ameliyat tarihi ve pozitif netice veren göğüs kanserli hasta sayısı kullanılmıştır; veri kümesi toplam 306 adet veri içermekte ve bu verilerin 255 adeti c_1 sınıfına, 81 adeti ise c_2 sınıfına ait bulunmaktadır. Sınıflandırıcıların başarımlarının karşılaştırılmasında kolaylık sağlaması açısından iki sınıfa ait eşit dağıtılmış rastgele 104 veri alınmıştır. Bu verilerden de her sınıftan 42 ve toplamda 84 veri kullanılarak, Bölüm 4.1'de ele alınan Fisher tabanlı algoritma Haberman veri kümesine uygulanmıştır.

Fisher tabanlı algoritma sonucunda izdüşüm vektörü $\vec{v} = (0.23 \ 0.50 \ 0.80)$ olarak elde edilmiştir. Böylece bu izdüşüm vektörü (6.4) ifadesinde gösterildiği şekilde kullanılarak Haberman veri kümesinin orijine olan izdüşüm uzaklıkları bulunmuş ve Şekil 7.7'deki şekilde çizdirilmiştir. Bu şekilde, küçük yuvarlaklar c_1 ve artı işaretleri c_2 sınıfına ait verilerdir.



Şekil 7.7 : İzdüşürülmüş Haberman verilerinin orijine uzaklıkları.

Şekil 7.7'den görüldüğü gibi yatay eksen verilerin orijinden uzaklıklarını dikey eksen ise veri numarasını göstermektedir. Dolayısıyla iris veri kümesinde olduğu gibi, aynı yöntemle Şekil 7.7'deki verilerin uzaklıkları ile de Haberman veri kümesi de sınıflandırılabilmektedir.

Bu sınıflandırma işleminin simülasyonunda ÇAD ve ÇD yapılarından faydalanılmıştır. Şekil 7.8'de Haberman verisi sınıflandırıcısı blok diyagramı verilmiştir. ÇAD çıkışlarındaki akımların toplamı, girişe uygulanan verinin orijine olan izdüşüm uzaklığına karşılık gelmektedir. Dolayısıyla sınıflandırma işlemi için, izdüşüm uzaklığı (akım değeri) iki çıkışlı akım çoğullayıcı devre ile iki çekirdek devre girişine uygulanmıştır. Bu sayede çekirdek devreler ile Şekil 7.7'de gösterilen bölgelerin seçilmesi sağlanmaktadır.



Şekil 7.8 : Haberman verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuştur).

Tablo 7.6'da verilmiş olan Haberman verisinin özellikleri olan x_i (i=1,2,3) değerleri μ A akım boyutunda düşünülmüştür. Simülasyon için Tablo 7.6'daki 0-1 ms arasında x_1 =34 verisinin ÇAD-1 yapısında kullanılması şu şekilde özetlenebilir: Şekil 4.15'deki yapıda V_y =34 mV ve R_1 =10 k Ω alınması durumunda giriş büyüklüğü akımı V_y/R_1 = 34 μ A olarak seçilmiştir. Ayrıca Fisher tabanlı algoritmanın uygulanmasıyla bulunan $\vec{v} = (0.23 \ 0.50 \ 0.80)$ vektörünün ilk elemanı 0.23 olduğu için, $R_2/R_3 = 23 k\Omega/100 k\Omega$ alınarak (4.5) ifadesine göre çıkış akımı elde edilir. Diğer taraftan Şekil 7.7'de c_1 sınıfına ait verilerin orijine olan izdüşüm uzaklığı 38 ile 45 arasında olduğu için Şekil 7.8'deki çekirdek devre-1'in kontrol akımı değerleri I₁=38 μ A, I₂=45 μ A olacak şekilde seçilmiştir. Benzer şekilde Şekil 7.8'de kullanılan ÇAD ve ÇD parametreleri bulunmuş ve sırası ile Tablo 7.7'de ve Tablo 7.8'de verilmiştir. Ayrıca her çekirdek devrenin I_H akımı farklı seçilmiştir. Bu sayede Şekil 7.8'deki yapının çıkış akımına bakılarak veri sınıfının bulunması sağlanmıştır.

| Verilerin uygulanma süresi | x_1 | x_2 | <i>x</i> ₃ | Sınıf |
|----------------------------|-------|-------|-----------------------|-------|
| 0 ms - 1 ms | 34 | 60 | 1 | c_1 |
| 1 ms - 2 ms | 61 | 68 | 1 | c_2 |
| 2 ms - 3 ms | 51 | 59 | 3 | c_2 |
| 3 ms - 4 ms | 37 | 59 | 6 | c_1 |
| 4 ms - 5 ms | 54 | 58 | 1 | c_1 |
| 5 ms - 6 ms | 61 | 62 | 5 | c_2 |
| 6 ms - 7 ms | 42 | 63 | 1 | c_1 |
| 7 ms - 8 ms | 53 | 61 | 1 | c_1 |
| 8 ms - 9 ms | 48 | 67 | 7 | c_2 |
| 9 ms - 10 ms | 65 | 66 | 15 | c_2 |
| 10 ms - 11 ms | 60 | 59 | 17 | c_2 |
| 11 ms - 12 ms | 42 | 59 | 2 | c_1 |
| 12 ms - 13 ms | 30 | 62 | 3 | c_1 |
| 13 ms - 14 ms | 65 | 62 | 22 | c_2 |
| 14 ms - 15 ms | 41 | 60 | 23 | c_2 |
| 15 ms - 16 ms | 46 | 58 | 3 | c_1 |
| 16 ms - 17 ms | 42 | 61 | 4 | c_1 |
| 17 ms - 18 ms | 72 | 67 | 3 | c_1 |
| 18 ms - 19 ms | 47 | 63 | 23 | c_2 |
| 19 ms - 20 ms | 43 | 58 | 52 | c_2 |

Tablo 7.6: Şekil 7.8'deki Haberman verisi sınıflandırıcısı test kümesi.

Tablo 7.7: Haberman sınıflandırıcısı ÇAD direnç değerleri.

| Dirençler | ÇAD-1 | ÇAD-2 | ÇAD -3 |
|-----------|--------|-------|--------|
| R_1 | 10 kΩ | 10 kΩ | 10 kΩ |
| R_2 | 2.3 kΩ | 5 kΩ | 8 kΩ |
| R_3 | 10 kΩ | 10 kΩ | 10 kΩ |

Tablo 7.8: Haberman sınıflandırıcısı çekirdek devre kontrol akımları.

| Kontrol akımları | ÇD-1 | ÇD-2 |
|------------------|-------|-------|
| I_1 | 38 µA | 45 μΑ |
| I_2 | 46 µA | 82 µA |
| I_H | 10 µA | 20 µA |
| | | |

Sınıflandırıcının başarımını sınamak için her sınıftan geri kalan 10 adet, toplamda 20 adet veri kullanılarak Şekil 7.8'deki sınıflandırıcı yapının SPICE simülasyonu yapılmıştır. Simülasyon için kullanılan test verileri Tablo 7.6'da gösterilen aralıklar ile Şekil 7.8'deki yapıya uygulanmıştır. Tablo 7.6'da sınıflandırıcı sonuçlarının ne olması gerektiği görülmektedir. Şekil 7.9'daki simülasyon sonucu, çıkış değeri 10 μ A ise verinin c_1 , 20 μ A ise verinin c_2 sınıfına ait olduğunu göstermektedir.



Şekil 7.9 : Şekil 7.8'deki haberman verisi sınıflandırıcısı simülasyon sonucu.

DU-TCC 1209 yardımıyla Haberman verisinin Fisher tabanlı algoritma ile testi için Şekil 7.8'deki blok yapısının deney düzeneği kurulmuştur. Bu düzenek kurulurken simulasyonlarda kullanılan ÇAD yapısından farklı olarak (test düzeneğini kurulum kolaylığı için) tek bir CCII elemanı kullanılmıştır. Bu ÇAD yapısında ise simulasyonda kullanılan direnç değerleri için $\frac{V_y}{R_1}\frac{R_2}{R_3}$ ifadesindeki dirençler yerine $\frac{1}{R_1}\frac{R_2}{R_3}$ nin eşdeğer direnç değeri alınmıştır. Eşdeğer direnç değerleri Tablo 7.9'da tabloda verilmiştir.

Tablo 7.9: Haberman sınıflandırıcısı ÇAD eşdeğer direnç değerleri.

| Dirençler | ÇAD-1 | ÇAD-2 | ÇAD -3 | |
|-----------|---------|-------|---------|--|
| R | 43.5 kΩ | 20 kΩ | 12.5 kΩ | |

Sınıflandırıcının DU-TCC 1209 ile testi için değişken işaretler üretebilen bir işaret üreteci kullanılarak simülasyondakine benzer şekilde 1ms aralıklarla uygun giriş gerilimleri direnç üzerinden uygulanmıştır. Dolayısıyla sınıflandırıcı yapısına uygun akım değerlerinin girilmesi sağlanmıştır. Simülasyonlarda belirtilen kontrol akım değerleri de uygun dirençler sayesinde ayarlanmıştır. Çıkış karakteristiğinde c₁ sınıfına ait olan verilerin çıkış akımı 10 μ A ve c₂ sınıfına ait olan verilerin içinse 30 μ A çıkış verecek şekilde test düzeneği kurulmuştur. Sınıflandırıcı yapısının çıkışına 10k Ω 'luk direnç bağlanılarak çıkış akımı ölçülmüş ve çıkış geriliminin değişim karakteristiği Şekil 7.10'da verilmiştir. Osiloskopda X ekseni 2ms/Div ve Y ekseni 0.1V/Div olarak ayarlanmıştır.



Şekil 7.10 : DU-TCC 1209 ile Haberman verisi için test sonucu osiloskop çıktısı.

Şekil 7.9'da simülasyon sonucu verilen 20 adet Haberman verisinin Şekil 7.10'da verilmiş olan test sonucu ile de uyum içinde olduğu ve Fisher tabanlı algoritma ile doğru olarak sınıflandırıldığı gösterilmiştir.

7.1.3 İris verisinin perseptron öğrenme algoritması ile sınıflandırılması

Bu kez, iris veri kümesi, Bölüm 5.2'de ele alınan ÇAD ve ÇD parametrelerinin perseptron öğrenme algoritması kullanılarak bulunmasından faydalanılarak sınıflandırılmıştır;.ÇAD ve ÇD parametrelerinin hesaplanması için her sınıftan 40 veri, toplamda 120 veri kullanılmıştır. Veri kümesi dört boyutlu ve sınıf sayısı üç olduğu için, Bölüm 5.2'de incelenen öğrenme algoritmasına göre üç adet çift eşik hiperdüzlemi elde edilmiş olup, bu çift eşik hiperdüzlemlerinden biri c_1 sınıfını diğer ikisi ise c_2 sınıfının bulunduğu bölgeleri göstermektedir. Üç çift eşik hiperdüzleminin belirlediği bölgelerin hiçbirinde bulunmayan veriler ise üçüncü sınıfa ait veri olarak sınıflandırılmıştır. Bu algoritma sonucu, her sınıf için gereken \vec{v} vektörü (ÇAD parametreleri w_i 'ler) ve a,b sayıları (ÇD parametreleri, I_1 ve I_2) bulunmuştur. Böylece c_1 sınıfına ait veriler için $\vec{v}_{c1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$, a=1 ve b=3.5 elde edilmiştir. Benzer şekilde c_2 sınıfına ait veriler için $\vec{v}_{c2-1} = (0.3 \ 1.0 \ -0.2 \ 3.7)$, $a_1=7.8$ ve $b_1=9.3$ ve $\vec{v}_{c2-2} = (0.4 \ 2.0 \ -0.1 \ 3.5)$, $a_2=14$ ve $b_2=15$ bulunmuştur. Elde edilen bu sonuçlar için oluşturulmuş olan iris sınıflandırıcısı bloğu Şekil 7.11'de verilmiştir.

Sınıflandırıcının başarımını sınamak için her sınıftan geri kalan 10, toplamda 30 adet veri kullanılarak Şekil 7.11'de blok diyagramı verilen devrenin SPICE simülasyonu yapılmıştır. Simülasyon için kullanılan test verileri Tablo 7.10'da gösterilen aralıklar ile Şekil 7.11'deki devreye uygulanmıştır.

| Verilerin Uygulanma Süresi | x_1 | x_2 | <i>x</i> ₃ | x_4 | Sınıf |
|----------------------------|-------|-------|-----------------------|-------|-----------------------|
| 0 ms - 1 ms | 4.3 | 2.3 | 1.4 | 0.2 | c_1 |
| 1 ms - 2 ms | 5.7 | 2.7 | 3.9 | 1.1 | c_2 |
| 2 ms - 3 ms | 5.7 | 2.7 | 4.8 | 1.8 | c_2 |
| 3 ms - 4 ms | 4.9 | 2.2 | 6 | 2.5 | c_3 |
| 4 ms - 5 ms | 5.6 | 2.5 | 5.1 | 1.9 | c_3 |
| 5 ms - 6 ms | 4.6 | 3 | 1.7 | 0.4 | c_1 |
| 6 ms - 7 ms | 4.7 | 3.1 | 1.5 | 0.1 | c_1 |
| 7 ms - 8 ms | 6.1 | 2.9 | 3.8 | 1.1 | c_2 |
| 8 ms - 9 ms | 6.4 | 3 | 4 | 1.3 | c_2 |
| 9 ms - 10 ms | 4.9 | 2 | 4.7 | 1.4 | <i>c</i> ₂ |
| 10 ms - 11 ms | 4.8 | 3.2 | 1.2 | 0.2 | c_1 |
| 11 ms - 12 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 12 ms - 13 ms | 6.3 | 2.8 | 6.5 | 1.8 | <i>c</i> ₃ |
| 13 ms - 14 ms | 6.7 | 3 | 6.4 | 2 | <i>c</i> ₃ |
| 14 ms - 15 ms | 7.2 | 3.2 | 5.4 | 2.1 | <i>c</i> ₃ |
| 15 ms -16 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |
| 16 ms - 17 ms | 5.3 | 3.7 | 1.5 | 0.2 | c_1 |
| 17 ms - 18 ms | 5.4 | 3.8 | 1.3 | 0.3 | c_1 |
| 18 ms - 19 ms | 6.3 | 3.0 | 4.4 | 1.3 | c_2 |
| 19 ms - 20 ms | 6.3 | 3.0 | 4.1 | 1.3 | c_2 |
| 20 ms - 21 ms | 6.9 | 3.1 | 5.6 | 2.4 | c_3 |
| 21 ms - 22 ms | 7.6 | 3.3 | 5.7 | 2.5 | c_3 |
| 22 ms - 23 ms | 7.7 | 3.4 | 5.2 | 2.3 | c_3 |
| 23 ms - 24 ms | 6.4 | 3.0 | 4.0 | 1.3 | c_2 |
| 24 ms - 25 ms | 6.4 | 3.0 | 4.4 | 1.2 | c_2 |
| 25 ms - 26 ms | 6.5 | 3.0 | 4.0 | 1.2 | c_2 |
| 26 ms - 27 ms | 7.7 | 3.4 | 5.0 | 1.9 | c_3 |
| 27 ms - 28 ms | 7.7 | 3.6 | 5.2 | 2.0 | <i>C</i> ₃ |
| 28 ms - 29 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 29 ms - 30 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |

Tablo 7.10: Şekil 7.11'deki iris verisi sınıflandırıcısı test kümesi.

Tablo 7.10'da verilmiş olan iris verisinin özellikleri olan x_i (*i*=1-4) değerleri cm boyutundadır. Ancak simülasyonda bu değerler ÇAD sayesinde gerekli normalizasyon yapılarak akıma dönüştürülmüş ve çekirdek devreye uygulanmıştır. Simülasyon yapılırken, Tablo 7.10'daki 0-1 ms arasında x_1 =4.3 verisinin, ÇAD-1 yapısında kullanılması şu şekilde özetlenebilir: Şekil 4.15'deki devrede V_y =0.43 V ve R_1 =10 k Ω alınarak giriş büyüklüğü V_y/R_1 = 43 µA olacak şekilde akım boyutuna dönüştürülür: ayrıca öğrenme algoritması ile c_1 sınıfı için bulunan \vec{v}_{c1} = (0.25 0.96 -1.50 -0.63) vektörünün ilk elemanı 0.25 olduğundan, R_2/R_3 = 25 k Ω / 100 k Ω alınarak (4.5) ifadesine göre çıkış akımı elde edilir. Bu çıkış değeri elde edilirken, x_1 =4.3 olan giriş verisi µA boyutunda gibi düşünülüp 10 kat büyütülmüş ve 43 µA olarak \vec{v}_{c1} vektörünün ilk elemanı ile çarpılmıştır. Bu durumda c_1 sınıfı için bulunmuş olan a=1 ve b=3.5 değerleri (çekirdek devre I_1 , I_2 parametresi) 10 kat büyük olacaktır. Diğer bir ifade ile Şekil 7.11'deki çekirdek devre-1'in kontrol akım değerleri $I_1=10 \ \mu$ A, $I_2=35 \ \mu$ A olarak seçilmiştir. Benzer durum göz önünde alınarak Şekil 7.11'de kullanılan ÇAD ve ÇD parametreleri, c_1 ve c_2 sınıfı için bulunarak sırası ile Tablo 7.11'de ve Tablo 7.12'de verilmiştir. Ayrıca Şekil 7.11'deki ÇAD yapılarının çıkışları \vec{v}_{ci} vektörünün elemanlarının pozitif ya da negatif olmasına göre seçilmiştir. Bu nedenle Şekil 4.15'den farklı olarak tek çıkış bulunmaktadır. Diğer bir ifadeyle $\vec{v}_{c1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$ vektörünün ilk ve ikinci elemanı pozitif, üçüncü ve dördüncü elemanı negatiftir. Dolayısıyla Şekil 7.11'de kullanılan ÇAD-1 ve ÇAD-2 yapısı için pozitif çıkış ucu, ÇAD-3 ve ÇAD-4 yapısı için negatif çıkış ucu kullanılmıştır.



Şekil 7.11 : İris verisi sınıflandırıcı bloğu (öğrenme algoritması sonucu oluşturulmuş).

Tablo 7.11: Şekil 7.11'de c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

| ÇAD-1 | ÇAD -2 | ÇAD -3 | ÇAD -4 | ÇD-1 |
|---------------------------|---------------------------|--------------------------|---------------------------|------------------------------|
| $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | <i>I</i> ₁ =10 μA |
| $R_2=2.5 \text{ k}\Omega$ | $R_2=9.6 \text{ k}\Omega$ | $R_2=15 \text{ k}\Omega$ | $R_2=6.3 \text{ k}\Omega$ | <i>I</i> ₂ =35 μA |
| $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | <i>I_H</i> =10 μA |

Tablo 7.12: Şekil 7.11'de c_2 ve c_3 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

| ÇAD-5 | ÇAD-6 | ÇAD-7 | ÇAD-8 | ÇD-2 | ÇAD-9 | ÇAD-10 | ÇAD-11 | ÇAD-12 | ÇD-3 |
|-----------------------------|-----------------------------|-----------------------------|-----------------------------|------------------------------|-----------------------------|-----------------------------|-----------------------------|---------------------|-------------------|
| $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | <i>I</i> ₁ =78 μA | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1=10$ k Ω | $I_1 = 140 \mu A$ |
| $R_2=3 \text{ k}\Omega$ | $R_2=10$ k Ω | $R_2=2$ k Ω | $R_2=37$ k Ω | <i>I</i> ₂ =93 μA | $R_2=4$ k Ω | $R_2=20$ k Ω | $R_2=1 \text{ k}\Omega$ | $R_2=35$ k Ω | $I_2 = 150 \mu A$ |
| $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $I_H = 20 \mu A$ | $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $I_H=20 \ \mu A$ |

Şekil 7.11'deki yapının çıkış akımı çekirdek devrelerin I_H akımları ile kontrol edilmektedir. Dolayısıyla her çekirdek devrenin I_H akımı farklı seçilerek veri kümesinin sınıflandırılması sağlanmıştır: çıkış akımının 10 µA olması durumunda sonuç c_1 , 20 µA olması durumunda ise c_2 , 0 µA olması durumunda da c_3 olarak değerlendirilmiştir. Simülasyon için kullanılan test verileri, Tablo 7.10'da gösterilen aralıklarla Şekil 7.11'deki yapıya uygulanmıştır; Tablo 7.10'da sınıflandırıcı sonuçlarının ne çıkması gerektiği gözükmektedir. Simülasyon sonucu Şekil 7.12'de verilmiştir.



Şekil 7.12 : Şekil 7.11'deki iris verisi sınıflandırıcısı simülasyon sonucu.

DU-TCC 1209 ile iris verisinin perseptron öğrenme algoritması kullanılarak testi için Şekil 7.11'deki blok yapısının deney düzeneği kurulmuştur. Bu düzenek kurulurken simulasyonlarda kullanılan ÇAD yapısından farklı olarak (test düzeneğini kurulum kolaylığı için) tek bir CCII elemanı kullanılmıştır. Bu ÇAD yapısında ise simulasyonda kullanılan direnç değerleri kullanılmıştır, ancak $\frac{V_y}{R_1} \frac{R_2}{R_3}$ ifadesinde kullanılan dirençler yerine $\frac{1}{R_1} \frac{R_2}{R_3}$ nin eşdeğer direnç değeri alınmıştır. Eşdeğer direnç değerleri Tablo 7.13'de verilmiştir.

| | | , | , | , | , | , | | |
|--------|-------------|-------|-------|---------|--------|---------|--------|--------|
| | | R | 40 kΩ | 10.5 kΩ | 6.7 kΩ | 15.8 kΩ | | |
| | | | | | | | | |
| | | | | | | | | |
| Direnç | ÇAD-5 | ÇAD-6 | ÇAD-7 | ÇAD-8 | ÇAD-9 | ÇAD-10 | ÇAD-11 | ÇAD-12 |
| R | $33k\Omega$ | 10kΩ | 50kΩ | 2.7kΩ | 25kΩ | 5kΩ | 100kΩ | 2.8kΩ |

Tablo 7.13: Sınıflandırıcı yapısında kullanılan ÇAD eşdeğer direnç değerleri.

Direnc CAD-1 CAD-2 CAD-3 CAD-4

Tablo 6.10'daki girişleri sırasıyla uygulamak için değişken işaretler üretebilen bir üretec ile simülasyondakine benzer şekilde 1ms aralıklarla uygun giriş gerilimleri direnç üzerinden uygulanmış böylece sınıflandırıcı yapısına uygun akım değerlerinin girilmesi sağlanmıştır. Simülasyonlarda belirtilen kontrol akım değerleri de uygun dirençler sayesinde ayarlanmıştır. Çıkış karakteristiğinde c₁ sınıfına ait olan verilerin çıkış akımı 10 μ A, c₂ sınıfına ait olan verilerin içinse 20 μ A ve c3 sınıfına ait veriler için 0 μ A çıkış verecek şekilde test düzeneği kurulmuştur. Sınıflandırıcı yapısının çıkışına 10k Ω 'luk direnç bağlanılarak çıkış akımı ölçülmüş ve çıkış geriliminin değişim karakteristiği Şekil 7.13'de verilmiştir. Osiloskopta X ekseni 3ms/Div ve Y ekseni 0.05V/Div olarak ayarlanmıştır.



Şekil 7.13 : DU-TCC 1209 ile İris verisi test sonucu osiloskop çıktısı.

Şekil 7.12'deki verilen simülasyon sonucu ile Şekil 7.13'deki test sonucu uyum içindedir. Dolayısıyla test sonucunda 30 adet test verisi doğru olarak sınıflandırılmıştır.

7.1.4 Haberman verisinin perseptron öğrenme algoritması ile sınıflandırılması

Haberman veri kümesi, Bölüm 4.2'de ele alınan ÇAD ve ÇD parametrelerinin perseptron öğrenme algoritması kullanılarak bulunmasından faydalanılarak sınıflandırılmıştır. Öğrenme algoritması yardımı ile ÇAD ve ÇD parametrelerinin hesaplanması için her sınıftan 42, toplamda 84 adet veri kullanılmıştır. Veri kümesi üç boyutlu ve sınıf sayısı iki olduğu için, Bölüm 4.2'de incelenen öğrenme algoritmasına göre 2 adet çift eşik hiperdüzlemi elde

edilmiştir. Bu çift eşik hiperdüzlemlerinin belirlediği bölge içinde kalanlar c_1 sınıfına ait veriler, kalmayanlar ise c_2 sınıfına ait veriler olarak sınıflandırılmıştır. Öğrenme algoritması sonucu \vec{v} vektörleri (ÇAD parametreleri, w_i 'ler) ve a, b sayıları (ÇD parametreleri I_1 ve I_2) bulunmuştur. Bu durumda c_1 sınıfına ait veriler için ilk $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörü, $a_1=11$ ve $b_1=69$ ve ikinci olarak $\vec{v}_2 = (-1.3 \ 2.2 \ -10)$ vektörü, $a_2=20$ ve $b_2=24$ değerleri elde edilmiştir. Bu değerler için oluşturulmuş Haberman sınıflandırıcısı bloğu Şekil 7.14'de verilmiştir.



Şekil 7.14 : Haberman verisi sınıflandırıcı bloğu (perseptron öğrenme algoritması sonucu oluşturulmuş).

Sınıflandırıcının başarımı için her sınıftan geri kalan 10 adet, toplamda 20 adet veri kullanılarak Şekil 7.14'deki yapıya karşılık düşen devre ile SPICE simülasyonu yapılmış, kullanılan test verileri Tablo 7.14'de gösterilen aralıklarla Şekil 7.14'deki yapıya uygulanmıştır.

| Verilerin Uygulanma Süresi | x_1 | x_2 | x_3 | Sınıf |
|----------------------------|-------|-------|-------|-------|
| 0 ms - 1 ms | 34 | 60 | 1 | c_1 |
| 1 ms - 2 ms | 61 | 68 | 1 | c_2 |
| 2 ms - 3 ms | 51 | 59 | 3 | c_2 |
| 3 ms - 4 ms | 37 | 59 | 6 | c_1 |
| 4 ms - 5 ms | 54 | 58 | 1 | c_1 |
| 5 ms - 6 ms | 61 | 62 | 5 | c_2 |
| 6 ms - 7 ms | 42 | 63 | 1 | c_1 |
| 7 ms - 8 ms | 53 | 61 | 1 | c_1 |
| 8 ms - 9 ms | 48 | 67 | 7 | c_2 |
| 9 ms - 10 ms | 65 | 66 | 15 | c_2 |
| 10 ms - 11 ms | 60 | 59 | 17 | c_2 |
| 11 ms - 12 ms | 42 | 59 | 2 | c_1 |
| 12 ms - 13 ms | 30 | 62 | 3 | c_1 |
| 13 ms - 14 ms | 65 | 62 | 22 | c_2 |
| 14 ms - 15 ms | 41 | 60 | 23 | c_2 |
| 15 ms - 16 ms | 46 | 58 | 3 | c_1 |
| 16 ms - 17 ms | 42 | 61 | 4 | c_1 |
| 17 ms - 18 ms | 72 | 67 | 3 | c_1 |
| 18 ms - 19 ms | 47 | 63 | 23 | c_2 |
| 19 ms - 20 ms | 43 | 58 | 52 | c_2 |

Tablo 7.14: Şekil 7.14'deki haberman verisi sınıflandırıcısı test kümesi.

Tablo 7.14'de verilmiş olan Haberman verisinin özellikleri olan x_i (*i*=1,2,3) değerleri μ A boyutunda olarak düşünülmüştür. Simülasyon için Tablo 7.14'deki 0-1 ms arasındaki x_1 =34 verisinin, ÇAD-1 yapısında kullanılması şu şekilde özetlenir: Şekil 4.15'deki yapıda V_y =34 mV ve R_1 =10 k Ω alınması durumunda giriş büyüklüğü V_y/R_1 = 34 μ A olarak akım boyutuna dönüştürülür. Ayrıca öğrenme algoritması ile c_1 sınıfı için bulunan $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörünün ilk elemanı -1.4 olduğu için $R_2/R_3 = 14 \ k\Omega / 10 \ k\Omega$ alınarak (4.6) ifadesine göre çıkış akımı elde edilir. Bu durumda c_1 sınıfı için bulunmuş olan a=11 ve b=69 değerleri (çekirdek devre I_1 , I_2 parametreleri) kullanılarak Şekil 7.14'deki çekirdek devre-1'in kontrol akım değerleri I_1 =11 μ A, I_2 =69 μ A olarak alınmıştır. Şekil 7.14'de kullanılan ÇAD yapılarının çıkışları \vec{v} vektörünün elemanlarının pozitif ya da negatif olmasına göre seçilmiştir. Diğer bir ifadeyle $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörünün ikinci elemanı pozitif, ilk ve üçüncü elemanı negatiftir. Dolayısıyla Şekil 7.14'de kullanılan ÇAD-2 bloğu için pozitif çıkış ucu, ÇAD-1 ve ÇAD-3 bloğu için ise negatif çıkış ucu kullanılmıştır. Şekil 7.14'deki çekirdek deir. Dolayısıyla çekirdek devrelerinin I_H akımı, uygulanan verinin c_1 sınıfına ait olup olmadığını

belirlemektedir. Şekil 7.14'de kullanılan ÇAD ve ÇD devrelerinin parametreleri c_1 sınıfı için Tablo 7.15'de verilmiştir.

| ÇAD-1 | ÇAD-2 | ÇAD-3 | ÇD-1 | ÇAD-4 | ÇAD-5 | ÇAD-6 | ÇD-2 |
|--------------------------|--------------------------|---------------------------|-----------------------------|--------------------------|--------------------------|---------------------------|------------------------------|
| $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $I_1 = 11 \ \mu A$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | <i>I</i> ₁ =20 μA |
| $R_2=14 \text{ k}\Omega$ | $R_2=22 \text{ k}\Omega$ | $R_2=160 \text{ k}\Omega$ | $I_2 = 69 \mu A$ | $R_2=13 \text{ k}\Omega$ | $R_2=22 \text{ k}\Omega$ | $R_2=100 \text{ k}\Omega$ | $I_2 = 24 \mu A$ |
| $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | <i>I_H</i> =10 μA | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | <i>I_H</i> =10 μA |

Tablo 7.15: Şekil 7.14'de c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

Simülasyon için kullanılan test verileri Tablo 7.14'de gösterilen sırada Şekil 7.14'deki sınıflandırıcıya uygulanmıştır. Tablo 7.14'de sonuçların ne çıkması gerektiği gözükmektedir. Şekil 7.14'deki yapının simülasyon sonucu da Şekil 7.15'de verilmiştir: çıkış değerinin 10 μ A olması durumunda sonuç c_1 , 0 μ A olması durumundaysa sonuç c_2 olarak değerlendirilmiştir.



Şekil 7.15 : Şekil 7.14'daki Haberman verisi sınıflandırıcısı simülasyon sonucu.

İris verisinin perseptron öğrenme algoritması kullanılarak DU-TCC 1209 ile testi için Şekil 7.14'deki yapının deney düzeneği kurulmuştur. Bu düzenek kurulurken simulasyonlarda kullanılan ÇAD yapısından farklı olarak (test düzeneğini kurulum kolaylığı için) tek bir CCII elemanı kullanılmış ve kullanılan direnç değerleri aynen alınmıştır, Ancak $\frac{V_y}{R_1} \frac{R_2}{R_3}$ ifadesinde dirençler yerine $\frac{1}{R_1} \frac{R_2}{R_3}$ nin eşdeğer direnç değeri kullanılmıştır. Eşdeğer direnç

değerleri Tablo 7.16'da verilmiştir.

Tablo 7.16: Sınıflandırıcı yapısında kullanılan ÇAD eşdeğer direnç değerleri.

| Direnç | ÇAD-1 | ÇAD-2 | ÇAD-3 | ÇAD-4 | ÇAD-5 | ÇAD-6 |
|--------|--------|--------|---------|--------|--------|-------|
| R | 7.2 kΩ | 4.5 kΩ | 0.63 kΩ | 7.7 kΩ | 4.5 kΩ | 1 kΩ |

Değişken işaretler üretebilen bir üreteç kullanılarak simülasyondakine benzer şekilde 1ms aralıklarla uygun giriş gerilimleri direnç üzerinden uygulanmıştır. Dolayısıyla sınıflandırıcı yapısına uygun akım değerlerinin girilmesi sağlanmıştır. Simülasyonlarda belirtilen kontrol akım değerleri de uygun dirençler sayesinde ayarlanmıştır. Çıkış karakteristiğinde c₁ sınıfına ait verilerin çıkış akımı 10 μ A, c₂ sınıfına ait olan veriler içinse 0 μ A çıkış olacak şekilde test düzeneği kurulmuştur. Sınıflandırıcı yapısının çıkışına 10k Ω 'luk direnç bağlanılarak çıkış akımı ölçülmüş ve çıkış geriliminin değişim karakteristiği Şekil 7.16'da verilmiştir. Osiloskopda X ekseni 2ms/Div ve Y ekseni 0.05V/Div olarak ayarlanmıştır.



Şekil 7.16 : DU-TCC 1209 ile Haberman verisi test sonucu osiloskop çıktısı.

Şekil 7.15'deki simülasyon sonucu ile Şekil 7.16'da verilen test sonucu uyum içindedir. Sonuçlardan da görüldüğü 20 adet Haberman test verisi doğru olarak sınıflandırılmıştır.

İris ve Haberman verilerine önerilen iki yöntem uygulanmış ve sınıflandırıcı başarımları Tablo 7.17'de verilmiştir. Önerilen algoritmalar ile sınıflandırıcı parametreleri Pentium 3 GHz işlemcili ve 1 GB belleği olan bir bilgisayar yardımı ile gerçekleştirilmiştir.

| Algoritma | Sınıflandırıcı | hata başarımı [%] | Algoritmanın sınıflandırıcı parametrelerini bulma süresi [sn] | | | |
|-----------------------|----------------|-------------------|--|----------|--|--|
| | İris | Haberman | İris | Haberman | | |
| Fisher tabanlı | 100 | 100 | 0.4 | 0.3 | | |
| Perseptron öğrenme | 100 | 100 | 220 | 190 | | |

Tablo 7.17: Sınıflandırıcı başarımları karşılaştırması.

Tablo 7.17'den görüldüğü gibi her iki yöntem ile de veri sınıflandırılabilmektedir. Sınıflandırıcı algoritmasının kontrol parametrelerini bulmasında, Fisher tabanlı algoritmanın daha iyi olduğu gözlenmektedir.

Dolayısıyla gerçekleştirilmiş olan sınıflandırıcı devreleri ile literatürdeki veri kümelerinin DU-TCC 1209 ile başarılı bir şekilde sınıflandırıldığı, ayrıca sınıflandırma işlemi dışındaki diğer çeşitli uygulamalar için de kullanılabilecekleri ve yeni olanaklar ortaya çıkardıkları gösterilmiştir.

8. SONUÇLAR VE ÖNERİLER

Bu projede, kuantalama, görüntü işleme, istatistik, tıbbi tanı, otomatik hedef belirleme, yapay zekâ, YSA, analog-sayısal dönüştürücüler gibi uygulama alanı bulan sınıflandırıcı devreler ele alınmıştır. Literatürde rastlanan devrelerin dışarıdan ayarlanabilme yeteneğine sahip olma gibi bazı sınıflandırma konularında istenildiği biçimde gereksinimleri karşılayamadığı saptanmıştır. Çalışmada, bu eksikliklerin bir kısmı giredirmiş ve mevcut literatüre katkı sağlayacak yeni devreler geliştirilmiş, bu devrelerin bazı gerçek uygulama yapıları gerek simülasyonlar ile gerekse testleri yapılarak sonuçları verilmiştir.

Bu çalışma kapsamında yapılanları şöyle sıralayabiliriz:

Gerek dışarıdan ayarlanabilme yeteneği, gerekse uygulama alanlarının esnek bir şekilde kullanılmalarını sağlamak amacıyla önce temel bir sınıflandırıcı devre bloğuna işlevsel olarak karar verilmiş ve bu işlevi gerçekleyen iki adet CMOS devre önerilmiştir. Temel bir yapı niteliğinde olan bu sınıflandırıcı işlevsel bloğa çekirdek devre ismi verilmiştir. Bu çekirdek devrenin önemi, temel bir yapı taşı gibi çeşitli topolojiler de kullanılabilir olmasıdır. Ayrıca, bu çekirdek devre literatürdeki mevcut yapılara göre çok daha esnek şekilde tasarlanmıştır (YILDIZ, 2007a). Çekirdek devre blokları, akım modlu tasarlanmış olmalarından dolayı, bloklar uygun şekilde paralel ya da ard arda bağlanabilmektedir. Dolayısıyla tek boyutlu sınıflandırıcı yapıların kolayca oluşturulmasına olanak tanınmıştır. Bu çok boyutlu sınıflandırıcı devreler ile dikdörtgen ızgara bölgelerle ayrılmış veri kümelerinin sınıflandırılacağı da gösterilmiştir ki bu tür veriler en yoğun kullanılanların başında gelmektedir. Özellikle, bazı veri kümeleri için çekirdek devrenin esnek yapıda tasarlanmasının getirdiği kullanım kolaylıkları da üstünlükleri arasındadır. Ayrıca bahsedilen temel çekirdek yapısının, literatürdeki mevcut sınıflandırına yapılarıyla karşılaştırılarak, oldukça daha az güç harcadığı gösterilmiştir (YILDIZ, 2007b).

Diğer taraftan geliştirilen çok boyutlu sınıflandırıcı, bu çalışmada ele alınan karakter tanıma ve kuantalayıcı uygulamalarında olduğu gibi sadece dikdörtgen ızgara dağılımına sahip verilere uygulanabildiğinden, uygulanma alanlarını genişletebilmek amacıyla çekirdek devrelerin önüne girişin lineer kombinasyonunu alan ve çarpan devre denilen yeni bir kat

eklenmiştir. Söz konusu ön kat, çok yoğun kullanılan bazı veri dağılımlarının (örneğin, iris, Haberman, CMOS tranzistorların standart bölgeleri) sınıflandırılmasını da sağlamıştır (YILDIZ, 2009). Çarpan devrenin sağladığı giriş ağırlık katsayılarını da kapsayacak şekilde sınıflandırma işlemi için gerekli parametrelerinin bulunmasını sağlayan iki faklı yöntem geliştirilmiş, üstünlükleri ve sakıncaları incelenmiştir.

Ayrıca, yöntemlerin dayandığı iki öğrenme algoritması kullanılarak literatürde çok kullanılan iki farklı veri kümesini sınıflandıran devrelerin parametreleri bulunmuş; simülasyonlar ve laboratuar testleriye başarımları gösterilmiştir. Sonuçlar devrelerin ve algoritmaların etkinliğini ve başarımını göstermesi için karşılaştırmalı olarak verilmiştir.

Yoğun emek gerektiren bir katkı da sınıflandırma işlemini yapacak DU-TCC 1209 integre devresinin, çeşitli aşamalardan geçen seriminin tasarımı, imalatı ve sınanması olmuştur. 0.35 µm AMS CMOS teknoloji parametreleriyle tasarlanan DU-TCC1209'un prototipi Fransa'daki Multi-Project Circuit firmasında imal edilmiş, tasarlanan bir baskı devre ile kullanılarak yazılımlarla elde edilen sonuçları başarıyla verdiği deneylerle gösterilmiştir.

Proje çalışmalarına, sınıflandırma ve sınıflandırıcı uygulamaları amacıyla yola çıkıldığından kontrol akımları sabittir; dolaysıyla bütün düşünce ve uygulamalarda devrenin statik (zamanla değişmeyen) çalışması göz önüne alınmıştır. Kontrol akımlarının zamanla değiştirilmesi ile elekronik ve haberleşmede kullanılacak bir çok yeni devre ve uygulama geliştirilebileceği fark edilmiştir. DU-TCC1209'un dinamik karakteristiklerinin incelenmesinden sonra sözü edilen devre ve uygulamaların geliştirilmesi ayrı bir araştırma ve proje konusu olacaktır.

Proje çalışmaları sırasında üç adet dergi (ikisi SCI kapsamında) üç adet te bildiri yayımlanmış olup, bir dergi (SCI) ve bir bildiri de hazırlanmaktadır.

REFERANSLAR

ABDEL-ATY-ZOHDY, H. S., and Al-Nsour, M., Reinforcement learning neural network circuits for electronic nose, IEEE International Symposium on Circuits and Systems, Florida, USA, May 30- July 2, (1999) pp: 379- 382.

AKSIN, D. Y., Aras, S., and Göknar, İ. C., CMOS Realization of User Programmable, Single-Level, Double-Threshold Generalized Perceptron, Proceedings of Turkish Artificial Intelligence and Neural Networks Conference, İzmir, Turkey, July 21-23, (2000) pp: 117-125.

AKSIN, D. Y., and Aras, S., A compact Distance Cell for Analog Classifiers, Proceedings of the IEEE International Symposium on Circuits and Systems, Kobe, Japan, May 23-26, (2005) pp: 3627-3630.

ALMEIDA, A. P., and Franca, J. E., A mixed-mode architecture for implementation of analog neural networks with digital programmability, Proceedings of the International Joint Conference on Neural Networks, Nagoya, Japan, October 25-29, (1993) pp: 887-890.

AVCI, M., *Hibrit Bir Donanımla Yapay Sinir Ağı Gerçekleme AY Tümdevresi*, (Doktora Tezi), Yıldız Teknik Üniversitesi, İstanbul (2005).

AYBAY, I., Çetinkaya, S., and Halıcı, U., Classification of Neural Networks Hardware, *Neural Networks World*, Vol. 6, 11-29, (1996).

BEIU, V., *Neural Networks and Their Application*, John Wiley & Sons Ltd., Baffins Lane, Chichester, England, pp. 1-98, (1996).

BERMAK, A., and Martinez, D., 2003: A Compact 3-D VLSI Classifier Using Bagging Threshold Network Ensembles, *IEEE Transaction on Neural Networks*, Vol. 14, 1097-1109, (2003).

BRIDGES, S., Holleman, J., and Figueroa. M., 2006: A Random Projection Imager for Visual Pattern Classification in Analog VLSI, Proceedings of the 32nd European Solid-State Circuits Conference, Montreux, Switzerland, September 18-22, (2006) pp: 428-431.

BİLGİLİ, E., Göknar, İ. C., and Uçan, O. N., Cellular Neural Networks with Trapezoidal Activation Function, *Int. Journal of Circuit Theory and Applications*, Vol. 33, 393-417, (2005).

BİLGİLİ, E., Göknar, C., Uçan, O. N., and Albora, M., Stability of CNN with Trapezoidal Activation Function, *Complex Computing-Networks: Brain-like and Wave-oriented Electrodynamic Algorithms Springer Proceedings in Physics*, Vol. 104, (2006) pp: 267-274.

BISHOP, C. M., 1996: Neural Networks for Pattern Recognition, Oxford University Press, Inc., Newyork, USA (1996). pp: 116-191.

CAMBONI, F., and Valle, M., A Mixed Mode Perceptron Cell for VLSI Neural Networks, International Conference on Electronics, Circuits and Systems, Malta, September 2-5, (2001) pp: 377-380. CHAKBARRTTY, S., and Cauwenberghs, G., Sub-Microwatt Analog VLSI Trainable Pattern Classifier, *IEEE Journal of Solid-State Circuits*, Vol. 42, 1169-1178, (2007).

CHENYZ, Y., Hungyz, Y., and Fuhz, C., Fast Algorithm for Nearest Neighbor Search Based on a Lower Bound Tree, Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada, July 7-14, (2001), pp: 446-453.

CHOU, C. H., and Chen, Y. C., A VLSI Architecture for real-time and flexible image template matching, *IEEE Trans. on Circuit and Systems*, Vol. 36, 1336-1342, (1989).

CİLİNGİROGLU, U., A charge based neural Hamming Classifier, *IEEE Journal of Solid-State Circuits*, Vol. 28, 59-67, (1993).

CİLİNGİROGLU, U., and Aksın, D. Y., A 4 transistor Euclidean Distance Cell for Analog Classifiers, Proceedings of the IEEE International Symposium on Circuits and Systems, Monterey, USA, May 31- June 3, (1998) pp: 84-87.

COVER, T. M., and Hart, P. E., Nearest neighbor pattern classification, *IEEE Trans. on Inform. Theory*, Vol. 13, 21–27, (1967).

ÇEVİKALP, H., Theoretical Analysis of Linear Discriminant Analysis Criteria, IEEE 14th Signal Processing and Communications Applications, Antalya, Turkey, April 17-19, (2006) pp: 1-4.

DJAHANSHAHİ, H., Jullien, G. A., Miller, W.C., and Ahmadi, M., Neural-Based Smart CMOS Sensors for On-Line Pattern Classification Application, Proceedings of the IEEE International Symposium on Circuits and Systems, Atlanta, USA, May 12-15, (1996) pp: 384-387.

DONCKERS, N., Dualibe, C., and Verleysen, M., Design of Complementary Low-Power CMOS Architectures for Looser-take all and Winner-take all, Proceedings of the Seventh International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems, Granada, Spain, April 7-9, (1999) pp: 360-365.

DRAGHICI, S., and Miller, D. A., A VLSI Neural Network Classifier Based on Integer-Valued Weights, International Joint Conference on Neural Networks, Washington, USA, July 10-16, (1999) pp: 2419-2424.

DUDA, R. O., Stork, D. G., and Hart, P. E., 2000: *Pattern Classification and Scene Analysis*, John Wiley & Sons Ltd., Inc., 2nd ed., England, (2000). pp. 1-57.

ERKMEN, B., *Genel Amaçlı Bir Yapay Sinir Ağının Karma Bir Donanım ile Gerçeklenmesi*, (Doktora tezi), Yıldız Teknik Üniversitesi, İstanbul, (2007).

FERRİ, G., and Guerrini, N., High valued passive element simulation using low-voltage low-power current conveyor for fully integrated applications, *IEEE Transactions on Circuit and Systems II*, Vol. 48, 405-409, (2001).

FISHER, R.A., The use of multiple measurements in taxonomic problems, *Annual Eugenics*, Vol. 7, 179-188, (1936).

GALUSHKIN, A.I., and Sudarikov, V.A., Adaptive neural network algorithm for solving linear algebra problems, IEEE Symposium on Neuroinformatics and Neurocomputers, Rostov, Russia, October 7-10, (1992) pp: 128 – 138.

GATT, E., and Micallef, J., Analogue Radial Basis Function Networks for Phoneme Recognition, International Conference on Electronics, Circuits, and Systems, Dubrovnik, Croatia, September 15-18, (2002) pp: 583-586.

GENÇ, İ., and Güzeliş, C., Threshold Class CNNs with Input-Dependent Initial State, *IEEE International Workshop on Cellular Neural Networks and their Applications*, London, England, April 14-17, (1998) pp: 130-135

GENOV, R., and Cauwenberghs, G., Kerneltron: Support Vector Machine in Silicon, *IEEE Transactions on Neural Networks*, Vol. 14, 1426-1434, (2003).

GRANT, D., Taylor, J., and Houselander, P., A High-Speed Integrated Hamming Neural Classifier, Proceedings of the IEEE International Symposium on Circuits and Systems, London, England, May 30 –June 2, (1994) pp: 479-482.

GRAY, R. M., and Neuhoff, D. L., Quantization, *IEEE Trans. on Information Theory*, Vol. 44, 2325-2383, (1998).

HABERMAN, S.J., Generalized Residuals for Log-Linear Models, Proceedings of the 9th International Biometrics Conference, Boston, USA, (1976) pp: 104-122.

HODGES D., Jackson H., and Saleh R., *Analysis and Design of Digital Integrated Circuits*, Third edition, McGraw Hill, Newyork, USA (2004) pp: 66-69.

HUNG, Y. C., and Liu B. D., A CMOS Analog Vector Quantizer For Pattern Recognition, The First IEEE Asia Pacific Conference on ASICs, Korea, August 23-25, (1999) pp: 112-115.

JAIN, A. K., Duin, R. P. W., and Mao, J., Statistical Pattern Recognition: A Review, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 22, 4-37, (2000).

JIANG, G., Xiaoqing D., and Youshou W., On improvement of Multiple Discriminant Analysis Method for Discriminative Feature Extraction, Systems, Man, and Cybernetics, Conference Proceedings, October 12-15, (1999) pp: 915 – 920.

JING, Y., An Improved Cascade SVM Training Algorithm with Crossed Feedbacks, First International Multi-Symposiums Computer and Computational Sciences, Istanbul, Turkey, June 20-24, (2006) pp: 735-738.

KACHARE, M., Carvajel, J. R. A., and Lopez, M. A. J., 2005: New Low-Voltage Fully Programmable CMOS Triangular/Trapezoidal Function Generator Circuits, *IEEE Trans. On Circuit and Systems-I*, Vol. 52, 2033-2040, (2005).

KUMAR, N., Pouliquen, P. O., and Andreou, A. G., Device Mismatch Limitations on Performance of a Hamming Distance Classifier, The IEEE International Workshop on Defect and Fault Tolerance in VLSI System, Venice, Italy, October 27-29, (1993) pp: 327-334.

LEONG, P. H. W., and Jabri, M. A., A Low Power VLSI Arrhytmia Classifier, *IEEE Transaction on Neural Networks*, Vol. 6, 1435-1445, (1995).

LEONG, P. H. W., Kakadu-A Low Power Analogue Neural Network Classifier, *International Journal of Neural Systems*, Vol. 4, 381-394, (1993).

LIU, B., Chen, C., and Tsao, J., A Modular Current-Mode Classifier Circuit for Template Matching Application, *IEEE Trans. On Circuit and Systems-II, Analog and Digital Sig. Process.*, Vol. 47, 145-151, (2000).

LU, C., Shi, B., and Chen, L., A General Purpose Neural Network with on Chip BP Learning, *International Symposium on Circuits and System*, Arizona, USA, May 16-29, (2002) pp: 520-523.

LUBKIN, J., and Cauwenberghs G., A Micropower Learning Vector Quantizer for Parallel Analog-to-Digital Data Compression, *Proceedings of the International Symposium on Circuits and Systems*, Vol 3, May 31-June 3, (1998) pp: 58-61.

MCCULLOCH, W. S., and Pitts, W., A logical calculus of the immanent in nervous activity, *Bulletion of Mathematical Biophysics*, Vol. 5, 115-133, (1943).

MOODY. J., and Darken C., Fast Learning in Networks of Locally-Tuned Processing Units, *Neural Computation*, Vol. 1, 281-294, (1989).

MORGÜL, A., and Temel, T., Current-mode level restoration circuit for multi-valued logic, *Electronics Letters*, Vol. 41, 230-231, (2005).

NARAYAN, R., Honbo, D., Memik, G., Choudhary, A., and Zambreno, J., An FPGA Implementation of Decision Tree Classification, Design, Automation & Test in Europe Conference & Exhibition, Yokohama, Japan, April 16-20, (2007) pp: 1-6.

ONAT, B. M., McNeil, J. A., and Çilingiroğlu, U., Implementation of a charge based Neural Euclidian Classifier for a 3 bit flash Analog to digital converter, *IEEE Transactions on Instrumentation and Measurement*, Vol. 46, 672-677, (1997).

OYANG, Y. J., Hwang S. C., Ou, Y.Y., Chen, C. Y., and Chen, Z. W., Data Classification with Radial Basis Function Networks based on a Novel Kernel Density Estimation Algorithm, *IEEE Transactions on Neural Networks*, Vol. 16, 225-236, (2005).

PENG, S. Y., Minch, B. A., and Hasler, P.,: Analog VLSI Implementation of Support Vector Machine Learning and Classification, Proceedings of the IEEE International Symposium on Circuits and System, Washington, USA, May 18-21, (2008) pp: 860-863.

PENG, S. Y., Hasler, P. E., and Anderson, D., An Analog Programmable Multi-Diemensional Radial Basis Function Based Classifier, International Conference on Very Large Scale Integration, Atlanta, USA, October 15-17, (2007) pp: 13-18.

QIAN, D., Modified Fisher's Linear Discriminant Analysis for Hyperspectral Imagery, *IEEE Geoscience and Remote Sensing Letters*, Vol. 4, 503-507, (2007).

QI, L., and Donald, W. T.,: Principal Feature Classification, *IEEE Transaction on Neural Networks*, Vol. 8, 155-160, (1997).

REAZ, M. B. I., Islam, S. Z., Ali, M. A. M., and Sulaiman, M. S., FPGA Realization of Backpropagation for Stock Market Prediction, Proceedings of the 9th International Conference on Neural Information, Sofia, Bulgaria, Vol. 2, November 18-22, (2002) pp: 960-964.

REYES, C., and Adjouadi, M.,: A Clustering Technique for Random Data Classification, IEEE International Conference on Systems, Man and Cybernetics, Vancouver, Canada, December 22-25, (1995) pp: 316-321.

RIPLEY, B. D., *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, UK., (1996), pp. 143-177,

SUN, P., and Makro K., A Neural Network Classifier For Conflicting Information Environments, International Conference on Neural Networks, Stocholm, Sweden, June 9-12, (1997) pp. 1617-1622.

TZANAKOU, E. M., *Supervised and unsupervised Pattern Recognition*, CRC Press, Inc., , Boca Raton, FL, USA., (2000). pp. 7-75.
WANG, Y., and Salam, F. M. A., Experiments Using CMOS Neural Networks Chips as Pattern/Character Recognizers, Proceedings of the IEEE International Symposium on Circuits and Systems, Singapore, June 11-14, (1991) pp: 1196-1199.

XIAOQIN, Z., Yajuan, H., and Yeung, D. S., Determining the relevance of input features for multilayer perceptrons, IEEE International Conference on Systems Man ans Cybernetics, October 5-8, (2003) pp: 874-879.

YAMASAKI, T., Yamamoto, K., and Shibata, T., Analog pattern classifier with flexible matching circuitry based on principal-axis-projection vector representation, Proceedings of the 27th European Solid-State Circuits Conference, Villach, Austria, September 18-20, (2001) pp: 197-200.

YAMASAKI, T., and Shibata, T., Analog soft-pattern-matching classifier using floating-gate MOS technology, *IEEE Trans. on Neural Networks*, Vol. 14, 1257-1265, (2003).

YILDIZ, M., Minaei, S., and Göknar, C., A CMOS Classifier Circuit using Neural Networks with Novel Architecture, *IEEE Transaction on Neural Networks*, Vol. 18, 1845-1849, (2007).

YILDIZ, M., Minaei, S., and Göknar, C., A Low-Power Multilevel-Output Classifier Circuit, European Conference on Circuit Theory and Design, Seville, Spain, August 26-30, (2007) pp: 747-750.

YILDIZ, M., Minaei, S., and Göknar, C., Realization and Template Matching Application of a CMOS Classifier Circuit, *IEEE Applied Electronics*, Pilsen, Czech Republic, September 10-11, (2008) pp: 231-234.

YILDIZ, M., Minaei, S., and Özoğuz, S., Linearly Weighted Classifier Circuit, Northeast Workshop on Circuits and Systems, 28 Haziran -1 Temmuz, (2009) pp: 99-102.

YILDIZ, M., Özoğuz, S., Minaei, S., Göknar, C., A Low-power Multilevel CMOS classifier circuit, *İTÜ Dergisi/d*, cilt. 9, sayı. 1, 57-64, (2010).

YILDIZ, M, Minaei, S, Göknar, C, A flexible current-mode classifier circuit and its applications, *International Journal of Circuit Theory and Applications*, DOI: 10.1002/cta.677, (2010)

ZENG, H., Ying, H., and Xingbo, S., A fuzzy central cluster neural classifier, Proceedings of the 3rd World Congress on Intelligent Control and Automation, Hefei, China, June 28–July 2, (2000) pp: 1747-1750.

ZHAO, Y., Deng, B., and Wang, Z., Analysis and Study of Perceptron to Solve XOR Problem, *Proceeding of the 2th International Workshop on Autonomous Decentralized System*, China, November 6-7, (2002) pp: 168-173.

ZHOU, Z., Chen, S., and Chen, Z., FANNC: A Fast Adaptive Neural Network Classifier, *Knowledge and Informaton Systems*, Vol. 2, pp. 115-129, (2000).

EKLER

 $\mathbf{EK}\,\mathbf{A}$: AMS 0.35 μm MOSIS SPICE teknoloji parametreleri

EK B : Maksimum rank(S_B)=c-1'in ispatı

EK C.1 : İris veri kümesi

EK C.2 : Haberman veri kümesi

EK A

Bu ekte simülasyonlarda kullanılan NMOS ve PMOS tranzistorların 0.35 µm AMS CMOS SPICE teknoloji parametreleri verilmiştir.

NMOS:

(LEVEL=7 MOBMOD =1.000e+00 CAPMOD =2.000e+00 NLEV=0 K1=6.044e-01 K2=2.945e-03 K3=-1.72e K3B=6.325e-01 NCH=2.310e+17 VTH0=4.655e-01 VOFF=-5.72e-02 DVT0=2.227e+01 DVT1=1.051e DVT2=3.393e-03 KETA=-6.21e-04 PSCBE1=2.756e+08 PSCBE2=9.645e-06 DVT0W=0.000 DVT1W=0.000 DVT2W=0.000 UA=1.000e-12 UB=1.723e-18 UC=5.756e-11 U0=4.035e+02 DSUB=5.000e-01 ETA0=3.085e-02 ETAB=-3.95e-02 NFACTOR=1.119e-01 EM=4.100e+07 PCLM=6.831e-01 PDIBLC1=1.076e-01 PDIBLC2=1.453e-03 DROUT=5.000e-01 A0=2.208e+00 A1=0.000 A2=1.000 PVAG=0.000 VSAT=1.178e+05 AGS=2.490e-01 B0=-1.76e-08 B1=0.000 DELTA=1.000e-02 PDIBLCB=2.583e-01 W0=1.184e-07 DLC=8.285e-09 DWC=2.676e-08 DWB=0.000 DWG=0.000 LL=0.000 LW=0.000 LWL=0.000 LLN=1.000 LWN=1.000e WL=0.000 WW=0.000 WWL=0.000 WLN=1.000e WWN=1.000 AT=3.300e+04 UTE=-1.80e KT1=-3.30e-01 KT2=2.200e-02 KT1L=0.000 UA1=0.000 UB1=0.000 UC1=0.000 PRT=0.000 CGDO=2.100e-10 CGSO=2.100e-10 CGBO=1.100e-10 CGDL=0.000 CGSL=0.000 CKAPPA =6.000e-01 CF=0.000 ELM=5.000e XPART=1.000e CLC=1.000e-15 CLE=6.000e-01 RDSW=6.043e+02 CDSC=0.000 CDSCB=0.000 CDSCD=8.448e-05 PRWB=0.000 PRWG=0.000 CIT=1.000e-03 TOX=7.700e-09 NGATE=0.000e NLX=1.918e-07 ALPHA0=0.000 BETA0=3.000e+01 AF=1.400 KF=2.810e-27 EF=1.000 NOIA=1.000e+20 NOIB=5.000e+04 NOIC=-1.40e-12 LINT=-1.67e-08 WINT=2.676e-08 XJ=3.000e-07 RSH=8.200e+01 JS=2.000e-05 CJ=9.300e-04 CJSW=2.800e-10 MJ=3.100e-01 MJSW=1.900e-01 PB=6.900e-01 TT=0.000e+00 PBSW=9.400e-01)

PMOS:

(LEVEL=7 MOBMOD =1.000e CAPMOD=2.000e NLEV=0 K1=5.675e-01 K2=-4.39e-02 K3B=-8.52e-01 NCH=1.032e+17 VTH0=-6.17e-01 K3=4.540e VOFF=-1.13e-01 DVT0=1.482e+00 DVT1=3.884e-01 DVT2=-1.15e-02 KETA=-2.56e-02 PSCBE1=1.000e+09 PSCBE2=1.000e-08 DVT0W=0.000 DVT1W=0.000 DVT2W=0.000 UA=2.120e-10 UB=8.290e-19 UC=-5.28e-11 U0=1.296e+02 DSUB=5.000e-01 ETA0=2.293e-01 ETAB=-3.92e-03 NFACTOR=8.237e-01 EM=4.100e+07 PCLM=2.979e+00 PDIBLC1=3.310e-02 PDIBLC2=1.000e-09 DROUT =5.000e-01 A0=1.423e A1=0.000 A2=1.000 PVAG=0.000 VSAT=2.000e+05 AGS=3.482e-01 B0=2,719e-07 B1=0.000 DELTA=1.000e-02 PDIBLCB=-1.78e-02 W0=4.894e-08 DLC=-5.64e-08 DWC=3.845e-08 DWB=0.000 DWG=0.000 LL=0.000 LW=0.000 LWL=0.000 LLN=1.000 LWN=1.000 WL=0.000 WW=0.000 WWL=0.000 WLN=1.000 WWN=1.000 AT=3.300e+04 UTE=-1.35e+00 KT1=-5.70e-01 KT2=2.200e-02 KT1L=0.000 UA1=0.000 UB1=0.000 UC1=0.000 PRT=0.000 CGDO=2.100e-10 CGSO=2.100e-10 CGBO=1.100e-10 CGDL=0.000 CGSL=0.000 CKAPPA =6.000e-01 CF=0.000 ELM=5.000 XPART=1.000 CLC=1.000e-15 CLE=6.000e-01 RDSW=1.853e+03 CDSC=6.994e-04 CDSCB=2.943e-04 CIT=1.173e-04 TOX=7.700e-09 CDSCD=1.970e-04 PRWB=0.000 PRWG=0.000 NGATE=0 000 NLX=1.770e-07 ALPHA0=0.000 BETA0=3.000e+01 AF=1 290e KF=1.090e-27 EF=1.000e NOIA=1.000e+20 NOIB=5.000e+04 NOIC=-1.40e-12 LINT=-8.14e-08 WINT=3.845e-08 XJ=3.000e-07 RSH=1.560e+02 JS=2.000e-05 CJ=1.420e-03 CJSW=3.800e-10 MJ=5.500e-01 MJSW=3.900e-01 PB=1.020e+00 TT=0.000e PBSW=9.400e-01)

EK B

Bu ekte, maksimum(rank(S_B)=c-1' olduğu ispatlanmıştır.

$$S_{B} = n_{1} \left(\vec{\mu}_{1} - \frac{1}{c} (\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c}) \right) \left(\vec{\mu}_{1} - \frac{1}{c} (\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c}) \right)^{T} +$$

$$n_{2} \left(\vec{\mu}_{2} - \frac{1}{c} (\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c}) \right) \left(\vec{\mu}_{2} - \frac{1}{c} (\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c}) \right)^{T} + ... +$$

$$n_{c} \left(\vec{\mu}_{c} - \frac{1}{c} (\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c}) \right) \left(\vec{\mu}_{c} - \frac{1}{c} (\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c}) \right)^{T}$$

$$S_{B} = \frac{n_{1}}{c^{2}} ((c - 1)\vec{\mu}_{1} - \vec{\mu}_{2} - ... - \vec{\mu}_{c}) ((c - 1)\vec{\mu}_{1} - \vec{\mu}_{2} - ... - \vec{\mu}_{c})^{T} +$$

$$\frac{n_{2}}{c^{2}} \left(-\vec{\mu}_{1} + (c - 1)\vec{\mu}_{2} - ... - \vec{\mu}_{c} \right) \left(-\vec{\mu}_{1} + (c - 1)\vec{\mu}_{2} - ... - \vec{\mu}_{c} \right)^{T} + ... +$$

$$(B.2)$$

 S_B matrisinin ifadesinde $\Im = [\vec{\mu}_1 \ \vec{\mu}_2 \dots \vec{\mu}_c]$ tanımı kullanılırsa, toplam üç matrisin çarpımı biçiminde aşağıda gösterildiği şekilde ifade edilebilir.

$$S_{B} = \Im \begin{cases} \frac{n_{1}}{c^{2}} \begin{bmatrix} c-1\\-1\\.\\.\\-1 \end{bmatrix} \begin{bmatrix} c-1 & -1 & . & -1 \end{bmatrix} + \frac{n_{2}}{c^{2}} \begin{bmatrix} -1\\c-1\\-1\\.\\.\\-1 \end{bmatrix} \begin{bmatrix} -1 & c-1 & -1 & . & -1 \end{bmatrix} + \dots + \\ \frac{n_{c}}{c^{2}} \begin{bmatrix} -1\\.\\.\\.\\-1\\c-1 \end{bmatrix} \begin{bmatrix} -1 & . & -1 & -1 \end{bmatrix} + \dots + \\ \Im^{T}$$
(B.3)

(B.3) ifadesinde ortadaki toplamda yer alan matrislerin hepsinin satırları aynı lineer bağımlılığı (satırların toplamları sıfır ediyor) sağladığından toplam matrisin dolayısıyla da S_B matrisinin rankı *c*-*1* sayısından büyük olamaz.

EK C.1

| Canak Yaprağı | Canak Yaprağı | Tac Yaprağı | Tac Yaprağı | ~ * |
|---------------|----------------|---------------|----------------|--------|
| Uzunluğu [cm] | Genişliği [cm] | Uzunluğu [cm] | Genişliği [cm] | Sinif |
| 4.30 | 2.30 | 1.40 | 0.20 | Setosa |
| 4.40 | 2.90 | 1.40 | 0.20 | Setosa |
| 4.40 | 3.00 | 1.30 | 0.20 | Setosa |
| 4.40 | 3.00 | 1.50 | 0.20 | Setosa |
| 4.50 | 3.00 | 1.40 | 0.20 | Setosa |
| 4.60 | 3.00 | 1.70 | 0.40 | Setosa |
| 4.60 | 3.00 | 1.40 | 0.30 | Setosa |
| 4.60 | 3.00 | 1.50 | 0.20 | Setosa |
| 4.60 | 3.10 | 1.40 | 0.20 | Setosa |
| 4.70 | 3.10 | 1.50 | 0.10 | Setosa |
| 4.70 | 3.10 | 1.50 | 0.20 | Setosa |
| 4.80 | 3.10 | 1.60 | 0.20 | Setosa |
| 4.80 | 3.10 | 1.40 | 0.10 | Setosa |
| 4.80 | 3.20 | 1.10 | 0.10 | Setosa |
| 4.80 | 3.20 | 1.20 | 0.20 | Setosa |
| 4.80 | 3.20 | 1.50 | 0.40 | Setosa |
| 4.90 | 3.20 | 1.30 | 0.40 | Setosa |
| 4.90 | 3.20 | 1.40 | 0.30 | Setosa |
| 4.90 | 3.30 | 1.70 | 0.30 | Setosa |
| 4 90 | 3 30 | 1 50 | 0.30 | Setosa |
| 5.00 | 3 40 | 1 70 | 0.20 | Setosa |
| 5.00 | 3 40 | 1 50 | 0.40 | Setosa |
| 5.00 | 3 40 | 1.00 | 0.20 | Setosa |
| 5.00 | 3 40 | 1 70 | 0.50 | Setosa |
| 5.00 | 3 40 | 1.90 | 0.20 | Setosa |
| 5.00 | 3 40 | 1.50 | 0.20 | Setosa |
| 5.00 | 3 40 | 1.60 | 0.20 | Setosa |
| 5.00 | 3 40 | 1.50 | 0.10 | Setosa |
| 5.00 | 3 40 | 1.50 | 0.20 | Setosa |
| 5.10 | 3 50 | 1.40 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.60 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.50 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.50 | 0.40 | Setosa |
| 5.10 | 3.50 | 1.50 | 0.10 | Setosa |
| 5.10 | 3.50 | 1.40 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.30 | 0.10 | Setosa |
| 5.10 | 3.00 | 1.20 | 0.20 | Setosa |
| 5.20 | 3.00 | 1.50 | 0.20 | Setosa |
| 5.20 | 3.70 | 1.30 | 0.10 | Setosa |
| 5.20 | 3.70 | 1.50 | 0.20 | Setora |
| 5.50 | 3.70 | 1.30 | 0.20 | Setora |
| 5.40 | 3.00 | 1.30 | 0.50 | Setora |
| 5.40 | 3.00 | 1.30 | 0.30 | Setora |
| 5.40 | 3.00 | 1.30 | 0.20 | Setora |
| 5.40 | 2.00 | 1.00 | 0.00 | Setora |
| 5.40 5.50 | 3.90 | 1.90 | 0.40 | Setosa |
| 5.50 | 3.90 4.00 | 1.40 | 0.50 | Setosa |
| 5.5U 5.70 | 4.00 | 1.00 | 0.20 | Selosa |
| 5.70 | 4.10 | 1.40 | 0.20 | Setosa |
| 5.70 | 4.20 | 1.50 | 0.20 | Setosa |

Tablo C.1 : İris veri kümesi.

| Canak Yaprağı | Canak Yaprağı | Tac Yaprağı | Tac Yaprağı | G 6 |
|---------------|----------------|---------------|----------------|-------------|
| Úzunluğu [cm] | Genişliği [cm] | Uzunluğu [cm] | Genişliği [cm] | Sinii |
| 5.80 | 4.40 | 1.40 | 0.20 | Setosa |
| 4.90 | 2.00 | 4.70 | 1.40 | Versicolour |
| 5.00 | 2.20 | 4.50 | 1.50 | Versicolour |
| 5.00 | 2.20 | 4.90 | 1.50 | Versicolour |
| 5.10 | 2.30 | 4.00 | 1.30 | Versicolour |
| 5.20 | 2.30 | 4.60 | 1.50 | Versicolour |
| 5.40 | 2.30 | 4.50 | 1.30 | Versicolour |
| 5.50 | 2.40 | 4.70 | 1.60 | Versicolour |
| 5.50 | 2.40 | 3.30 | 1.00 | Versicolour |
| 5.50 | 2.40 | 4.60 | 1.30 | Versicolour |
| 5.50 | 2.50 | 3.90 | 1.40 | Versicolour |
| 5.50 | 2.50 | 3.50 | 1.00 | Versicolour |
| 5.60 | 2.50 | 4.20 | 1.50 | Versicolour |
| 5.60 | 2.50 | 4.00 | 1.00 | Versicolour |
| 5.60 | 2.60 | 4.70 | 1.40 | Versicolour |
| 5.60 | 2.60 | 3.60 | 1.30 | Versicolour |
| 5.60 | 2.60 | 4.40 | 1.40 | Versicolour |
| 5.70 | 2.70 | 4.50 | 1.50 | Versicolour |
| 5.70 | 2.70 | 4.10 | 1.00 | Versicolour |
| 5.70 | 2.70 | 4.50 | 1.50 | Versicolour |
| 5 70 | 2 70 | 3 90 | 1 10 | Versicolour |
| 5 70 | 2 70 | 4 80 | 1.80 | Versicolour |
| 5 80 | 2 80 | 4 00 | 1 30 | Versicolour |
| 5 80 | 2.80 | 4 90 | 1.50 | Versicolour |
| 5 80 | 2.80 | 4 70 | 1.20 | Versicolour |
| 5 90 | 2.80 | 4 30 | 1 30 | Versicolour |
| 5 90 | 2.80 | 4 40 | 1.50 | Versicolour |
| 6.00 | 2.80 | 4 80 | 1 40 | Versicolour |
| 6.00 | 2.90 | 5.00 | 1.10 | Versicolour |
| 6.00 | 2.90 | 4 50 | 1.50 | Versicolour |
| 6.00 | 2.90 | 3 50 | 1.00 | Versicolour |
| 6.10 | 2.90 | 3.80 | 1.00 | Versicolour |
| 6.10 | 2.90 | 3.70 | 1.00 | Versicolour |
| 6.10 | 2.90 | 3 90 | 1 20 | Versicolour |
| 6.10 | 2.90 | 5.10 | 1.60 | Versicolour |
| 6.20 | 3.00 | 4 50 | 1.50 | Versicolour |
| 6.20 | 3.00 | 4 50 | 1.60 | Versicolour |
| 6.30 | 3.00 | 4 70 | 1.50 | Versicolour |
| 6.30 | 3.00 | 4.70 | 1.30 | Versicolour |
| 6.30 | 3.00 | 4.10 | 1.30 | Versicolour |
| 6.40 | 3.00 | 4.10 | 1.30 | Versicolour |
| 6.40 | 3.00 | 4.00 | 1.30 | Versicolour |
| 6 50 | 3.00 | 4.60 | 1.20 | Versicolour |
| 6.60 | 3.10 | 4.00 | 1.40 | Versicolour |
| 6.60 | 3 10 | 3 30 | 1.20 | Versicolour |
| 6 70 | 3.10 | 5.50 A 20 | 1 20 | Versicolour |
| 6 70 | 3.10 | 4.20 | 1.30 | Versicolour |
| 6 70 | 3.20 | 4.20 | 1.20 | Versicolour |
| 6.70 | 3.20 | 4.20 | 1.30 | Versicolour |
| 0.00 | 3.20 | 4.30 | 1.30 | Versiceleur |
| 0.90 | 3.3U 2.40 | 5.00 4.10 | 1.10 | Versicolour |
| /.00 | 3.40 2.20 | 4.10 | 1.30 | Virginiag |
| 4.90 | 2.20 | 0.00 | 2.30 | virginica |

| Çanak Yaprağı | Çanak Yaprağı | Taç Yaprağı | Taç Yaprağı | Same |
|---------------|----------------|---------------|----------------|------------|
| Uzunluğu [cm] | Genişliği [cm] | Uzunluğu [cm] | Genişliği [cm] | SIIII |
| 5.60 | 2.50 | 5.10 | 1.90 | Virginica |
| 5.70 | 2.50 | 5.90 | 2.10 | Virginica |
| 5.80 | 2.50 | 5.60 | 1.80 | Virginica |
| 5.80 | 2.50 | 5.80 | 2.20 | Virginica |
| 5.80 | 2.60 | 6.60 | 2.10 | Virginica |
| 5.90 | 2.60 | 4.50 | 1.70 | Virginica |
| 6.00 | 2.70 | 6.30 | 1.80 | Virginica |
| 6.00 | 2.70 | 5.80 | 1.80 | Virginica |
| 6.10 | 2.70 | 6.10 | 2.50 | Virginica |
| 6.10 | 2.70 | 5.10 | 2.00 | Virginica |
| 6.20 | 2.80 | 5.30 | 1.90 | Virginica |
| 6.20 | 2.80 | 5.50 | 2.10 | Virginica |
| 6.30 | 2.80 | 5.00 | 2.00 | Virginica |
| 6.30 | 2.80 | 5.10 | 2.40 | Virginica |
| 6.30 | 2.80 | 5.30 | 2.30 | Virginica |
| 6 30 | 2 80 | 5 50 | 1.80 | Virginica |
| 6 30 | 2.80 | 6 70 | 2 20 | Virginica |
| 6 30 | 2.80 | 6 90 | 2.30 | Virginica |
| 6 40 | 2.90 | 5.00 | 1.50 | Virginica |
| 6 40 | 2.90 | 5 70 | 2.30 | Virginica |
| 6 40 | 3.00 | 4 90 | 2.00 | Virginica |
| 6 40 | 3.00 | 6 70 | 2.00 | Virginica |
| 6 40 | 3.00 | 4 90 | 1.80 | Virginica |
| 6 50 | 3.00 | 5 70 | 2 10 | Virginica |
| 6 50 | 3.00 | 6.00 | 1.80 | Virginica |
| 6 50 | 3.00 | 4 80 | 1.80 | Virginica |
| 6.50 | 3.00 | 4.00 | 1.80 | Virginica |
| 6.70 | 3.00 | 5.60 | 2 10 | Virginica |
| 6.70 | 3.00 | 5.80 | 1.60 | Virginica |
| 6.70 | 3.00 | 6.10 | 1.00 | Virginica |
| 6.70 | 3.00 | 6.40 | 2.00 | Virginica |
| 6.70 | 3.00 | 5.60 | 2.00 | Virginica |
| 6.80 | 3.10 | 5.10 | 1.50 | Virginica |
| 6.80 | 3.10 | 5.60 | 1.30 | Virginica |
| 6.90 | 3.10 | 6.10 | 2 30 | Virginica |
| 6.90 | 3.10 | 5.60 | 2.30 | Virginica |
| 6.90 | 3 20 | 5.50 | 2.40 | Virginica |
| 7.10 | 3.20 | 5.50 4.80 | 1.80 | Virginica |
| 7.10 | 3.20 | 4.80 5.40 | 2.10 | Virginica |
| 7.20 | 3.20 | 5.40 | 2.10 | Virginica |
| 7.20 | 3.20 | 5.00 | 2.40 | Virginica |
| 7.20 | 3.20 | 5.10 | 2.30 | Virginica |
| 7.30 | 2 20 | 5.10 | 1.90 | Virginica |
| 7.40 7.60 | 3.30 | 5.90 5.70 | 2.30 2.50 | Virginica |
| /.00 | 5.50 | 5.70 | 2.30 | Virginica |
| /./0 | 3.40 | 5.20 | 2.30 | v irginica |
| /./0 | 3.40 | 5.00 | 1.90 | v irginica |
| /./0 | 3.00 | 5.20 | 2.00 | v irginica |
| /./0 | 3.80 | 5.40 | 2.30 | v irginica |
| 7.90 | 3.80 | 5.10 | 1.80 | v irginica |

Tablo C.1 : İris veri kümesi (devam).

EK C.2

Bu ekte veri sınıflandırıcılarının uygulamalarında eğitim ve test amacı ile kullanılan Haberman verisi verilmiştir. x_1 hastanın yaşını, x_2 ameliyat tarihini ve x_3 pozitif netice veren göğüs kanserli hasta sayısını göstermektedir. c_1 beş yıldan daha uzun yaşayan, c_2 beş yıldan daha az yaşayan hasta sınıfını göstermektedir.

| x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf |
|-------|-------|-------|-------|-------|-------|-------|-------|----|-------|-------|-------|-------|-------|-------|-------|
| 30 | 64 | 1 | c_1 | 37 | 63 | 0 | c_1 | 39 |) 59 | 2 | c_1 | 42 | 62 | 20 | c_1 |
| 30 | 62 | 3 | c_1 | 37 | 58 | 0 | c_1 | 39 | 63 | 4 | c_1 | 42 | 65 | 0 | c_1 |
| 30 | 65 | 0 | c_1 | 37 | 59 | 6 | c_1 | 40 |) 58 | 2 | c_1 | 42 | 63 | 1 | c_1 |
| 31 | 59 | 2 | c_1 | 37 | 60 | 15 | c_1 | 40 |) 58 | 0 | c_1 | 43 | 63 | 14 | c_1 |
| 31 | 65 | 4 | c_1 | 37 | 63 | 0 | c_1 | 40 |) 65 | 0 | c_1 | 43 | 64 | 2 | c_1 |
| 33 | 58 | 10 | c_1 | 38 | 59 | 2 | c_1 | 4 | 58 | 0 | c_1 | 43 | 64 | 3 | c_1 |
| 33 | 60 | 0 | c_1 | 38 | 60 | 0 | c_1 | 4 | 59 | 8 | c_1 | 43 | 60 | 0 | c_1 |
| 34 | 60 | 1 | c_1 | 38 | 62 | 3 | c_1 | 4 | 59 | 0 | c_1 | 43 | 65 | 0 | c_1 |
| 34 | 61 | 10 | c_1 | 38 | 64 | 1 | c_1 | 4 | 64 | 0 | c_1 | 43 | 66 | 4 | c_1 |
| 34 | 67 | 7 | c_1 | 38 | 66 | 0 | c_1 | 4 | 69 | 8 | c_1 | 44 | 61 | 0 | c_1 |
| 34 | 60 | 0 | c_1 | 38 | 66 | 11 | c_1 | 4 | 65 | 0 | c_1 | 44 | 63 | 1 | c_1 |
| 35 | 64 | 13 | c_1 | 38 | 60 | 1 | c_1 | 4 | 65 | 0 | c_1 | 44 | 61 | 0 | c_1 |
| 35 | 63 | 0 | c_1 | 38 | 67 | 5 | c_1 | 42 | 2 58 | 0 | c_1 | 44 | 67 | 16 | c_1 |
| 36 | 60 | 1 | c_1 | 39 | 63 | 0 | c_1 | 42 | 2 60 | 1 | c_1 | 45 | 60 | 0 | c_1 |
| 36 | 69 | 0 | c_1 | 39 | 67 | 0 | c_1 | 42 | 2 59 | 2 | c_1 | 45 | 67 | 0 | c_1 |
| 45 | 59 | 14 | c_1 | 49 | 62 | 0 | c_1 | 52 | 2 61 | 0 | c_1 | 54 | 69 | 7 | c_1 |
| 45 | 64 | 0 | c_1 | 49 | 66 | 0 | c_1 | 52 | 2 63 | 4 | c_1 | 54 | 63 | 19 | c_1 |
| 45 | 68 | 0 | c_1 | 49 | 60 | 1 | c_1 | 52 | 2 69 | 0 | c_1 | 54 | 58 | 1 | c_1 |
| 45 | 67 | 1 | c_1 | 49 | 62 | 1 | c_1 | 52 | 2 60 | 4 | c_1 | 54 | 62 | 0 | c_1 |
| 46 | 62 | 0 | c_1 | 49 | 63 | 3 | c_1 | 52 | 2 60 | 5 | c_1 | 55 | 58 | 1 | c_1 |
| 46 | 58 | 3 | c_1 | 49 | 61 | 0 | c_1 | 52 | 2 62 | 0 | c_1 | 55 | 58 | 0 | c_1 |
| 46 | 63 | 0 | c_1 | 49 | 67 | 1 | c_1 | 52 | 2 62 | 1 | c_1 | 55 | 58 | 1 | c_1 |
| 47 | 61 | 0 | c_1 | 50 | 59 | 0 | c_1 | 52 | 2 64 | 0 | c_1 | 55 | 66 | 18 | c_1 |
| 47 | 63 | 6 | c_1 | 50 | 61 | 6 | c_1 | 52 | 2 65 | 0 | c_1 | 55 | 66 | 0 | c_1 |
| 47 | 66 | 0 | c_1 | 50 | 61 | 0 | c_1 | 52 | 2 68 | 0 | c_1 | 55 | 69 | 3 | c_1 |
| 47 | 67 | 0 | c_1 | 50 | 63 | 1 | c_1 | 53 | 58 | 1 | c_1 | 55 | 69 | 22 | c_1 |
| 47 | 58 | 3 | c_1 | 50 | 58 | 1 | c_1 | 53 | 60 | 1 | c_1 | 55 | 67 | 1 | c_1 |
| 47 | 60 | 4 | c_1 | 50 | 59 | 2 | c_1 | 53 | 60 | 2 | c_1 | 56 | 60 | 0 | c_1 |
| 47 | 68 | 4 | c_1 | 50 | 61 | 0 | c_1 | 53 | 61 | 1 | c_1 | 56 | 66 | 2 | c_1 |
| 47 | 66 | 12 | c_1 | 50 | 64 | 0 | c_1 | 53 | 63 | 0 | c_1 | 56 | 66 | 1 | c_1 |
| 48 | 61 | 8 | c_1 | 50 | 65 | 4 | c_1 | 54 | 59 | 7 | c_1 | 56 | 67 | 0 | c_1 |

Tablo C.2 : Haberman veri kümesi.

| x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf |
|-------|-------|-------|-----------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 58 | 60 | 3 | c_1 | 59 | 67 | 3 | c_1 | 61 | 64 | 0 | c_1 | 76 | 67 | 0 | c_1 |
| 58 | 61 | 1 | c_1 | 60 | 61 | 1 | c_1 | 61 | 65 | 8 | c_1 | 77 | 65 | 3 | c_1 |
| 58 | 67 | 0 | c_1 | 59 | 64 | 7 | c_1 | 61 | 68 | 0 | c_1 | 34 | 59 | 0 | c_2 |
| 58 | 58 | 0 | c_1 | 63 | 63 | 0 | c_1 | 61 | 59 | 0 | c_1 | 34 | 66 | 9 | c_2 |
| 58 | 58 | 3 | c_1 | 63 | 66 | 0 | c_1 | 62 | 62 | 6 | c_1 | 38 | 69 | 21 | c_2 |
| 58 | 61 | 2 | c_1 | 63 | 61 | 9 | c_1 | 62 | 66 | 0 | c_1 | 39 | 66 | 0 | c_2 |
| 59 | 60 | 0 | c_1 | 63 | 61 | 28 | c_1 | 62 | 66 | 0 | c_1 | 41 | 60 | 23 | c_2 |
| 65 | 58 | 0 | c_1 | 64 | 58 | 0 | c_1 | 62 | 58 | 0 | c_1 | 41 | 64 | 0 | c_2 |
| 65 | 64 | 0 | c_1 | 64 | 65 | 22 | c_1 | 63 | 61 | 0 | c_1 | 41 | 67 | 0 | c_2 |
| 65 | 67 | 0 | c_1 | 64 | 66 | 0 | c_1 | 63 | 62 | 0 | c_1 | 42 | 69 | 1 | c_2 |
| 65 | 59 | 2 | c_1 | 64 | 61 | 0 | c_1 | 63 | 63 | 0 | c_1 | 42 | 59 | 0 | c_2 |
| 65 | 64 | 0 | c_1 | 64 | 68 | 0 | c_1 | 67 | 66 | 0 | c_1 | 43 | 58 | 52 | c_2 |
| 71 | 68 | 2 | c_1 | 69 | 65 | 0 | c_1 | 67 | 61 | 0 | c_1 | 43 | 59 | 2 | c_2 |
| 70 | 63 | 0 | c_1 | 70 | 66 | 14 | c_1 | 68 | 67 | 0 | c_1 | 43 | 64 | 0 | c_2 |
| 72 | 58 | 0 | c_1 | 70 | 67 | 0 | c_1 | 68 | 68 | 0 | c_1 | 43 | 64 | 0 | c_2 |
| 72 | 64 | 0 | c_1 | 70 | 68 | 0 | c_1 | 69 | 60 | 0 | c_1 | 44 | 64 | 6 | c_2 |
| 72 | 67 | 3 | c_1 | 65 | 67 | 1 | c_1 | 74 | 63 | 0 | c_1 | 44 | 58 | 9 | c_2 |
| 73 | 62 | 0 | c_1 | 73 | 68 | 0 | c_1 | 75 | 62 | 1 | c_1 | 44 | 63 | 19 | c_2 |
| 46 | 62 | 5 | <i>c</i> ₂ | 48 | 67 | 7 | c_2 | 52 | 59 | 2 | c_2 | 45 | 65 | 6 | c_2 |
| 46 | 65 | 20 | <i>c</i> ₂ | 49 | 63 | 0 | c_2 | 52 | 62 | 3 | c_2 | 45 | 66 | 0 | c_2 |
| 47 | 63 | 23 | <i>c</i> ₂ | 49 | 64 | 10 | c_2 | 52 | 66 | 4 | c_2 | 45 | 67 | 1 | c_2 |
| 47 | 62 | 0 | <i>c</i> ₂ | 50 | 63 | 13 | c_2 | 53 | 58 | 4 | c_2 | 46 | 58 | 2 | c_2 |
| 47 | 65 | 0 | <i>c</i> ₂ | 50 | 64 | 0 | c_2 | 53 | 65 | 1 | c_2 | 46 | 69 | 3 | c_2 |
| 48 | 58 | 11 | <i>c</i> ₂ | 51 | 59 | 13 | c_2 | 53 | 59 | 3 | c_2 | 56 | 65 | 9 | c_2 |
| 48 | 58 | 11 | <i>c</i> ₂ | 51 | 59 | 3 | c_2 | 53 | 60 | 9 | c_2 | 56 | 66 | 3 | c_2 |
| 54 | 65 | 5 | <i>c</i> ₂ | 52 | 69 | 3 | c_2 | 53 | 63 | 24 | c_2 | 57 | 61 | 5 | c_2 |
| 54 | 68 | 7 | <i>c</i> ₂ | 56 | 65 | 9 | c_2 | 53 | 65 | 12 | c_2 | 57 | 62 | 14 | c_2 |
| 55 | 63 | 6 | <i>c</i> ₂ | 56 | 66 | 3 | c_2 | 54 | 60 | 11 | c_2 | 57 | 64 | 1 | c_2 |
| 55 | 68 | 15 | <i>c</i> ₂ | 57 | 61 | 5 | c_2 | 54 | 65 | 23 | c_2 | 59 | 62 | 35 | c_2 |
| 62 | 59 | 13 | <i>c</i> ₂ | 57 | 64 | 1 | c_2 | 59 | 62 | 35 | c_2 | 60 | 65 | 0 | c_2 |
| 62 | 58 | 0 | <i>c</i> ₂ | 61 | 62 | 5 | c_2 | 60 | 59 | 17 | c_2 | 61 | 62 | 5 | c_2 |
| 62 | 65 | 19 | <i>c</i> ₂ | 61 | 65 | 0 | c_2 | 60 | 65 | 0 | c_2 | 61 | 65 | 0 | c_2 |
| 63 | 60 | 1 | <i>c</i> ₂ | 65 | 66 | 15 | c_2 | 67 | 63 | 1 | c_2 | 70 | 58 | 0 | c_2 |
| 65 | 58 | 0 | <i>c</i> ₂ | 66 | 58 | 0 | c_2 | 69 | 67 | 8 | c_2 | 70 | 58 | 4 | c_2 |
| 65 | 61 | 2 | <i>c</i> ₂ | 66 | 61 | 13 | c_2 | 74 | 65 | 3 | c_2 | 72 | 63 | 0 | c_2 |
| 65 | 62 | 22 | <i>c</i> ₂ | 67 | 64 | 8 | c_2 | 78 | 65 | 1 | c_2 | 83 | 58 | 2 | c_2 |
| 57 | 63 | 0 | c_1 | 59 | 63 | 0 | c_1 | 60 | 67 | 2 | c_1 | 66 | 58 | 0 | c_1 |
| 57 | 64 | 0 | c_1 | 59 | 64 | 1 | c_1 | 60 | 61 | 25 | c_1 | 66 | 58 | 1 | c_1 |
| 57 | 67 | 0 | c_1 | 59 | 64 | 0 | c_1 | 61 | 59 | 0 | c_1 | 67 | 66 | 0 | c_1 |

Tablo C.2 : Haberman veri kümesi (devam).

- [9] T.-L. Liao and S.-H. Tsai, "Adaptive synchronization of chaotic systems, and its application to secure communications," *Chaos Solitons Fractals*, vol. 11, no. 9, pp. 1387–1396, Jul. 2000.
- [10] Q. Liu, J. Cao, and Y. Xia, "A delayed neural network for solving linear projection equations and its analysis," *IEEE Trans. Neural Netw.*, vol. 16, no. 4, pp. 834–843, Jul. 2005.
- [11] H. Lu, "Chaotic attractors in delayed neural networks," *Phys. Lett. A*, vol. 298, no. 2–3, pp. 109–116, Jun. 2002.
- [12] V. Milanović and M. E. Zaghloul, "Synchronization of chaotic neural networks and applications to communications," *Int. J. Bifurcation Chaos*, vol. 6, no. 12B, pp. 2571–2585, Nov. 1996.
- [13] L. M. Pecora and T. L. Carroll, "Synchronization in chaotic systems," *Phys. Rev. Lett.*, vol. 64, no. 8, pp. 821–824, Feb. 1990.
- [14] H. Qi and L. Qi, "Deriving sufficient conditions for global asymptotic stability of delayed neural networks via nonsmooth analysis," *IEEE Trans. Neural Netw.*, vol. 15, no. 1, pp. 99–109, Jan. 2004.
- [15] Z. Tan and M. K. Ali, "Associative memory using synchronization in a chaotic neural network," *Int. J. Modern Phys. C*, vol. 12, no. 1, pp. 19–29, Jan. 2001.
- [16] L. Wang, S. Li, F. Tian, and X. Fu, "A noisy chaotic neural network for solving combinatorial optimization problems: Stochastic chaotic simulated annealing," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 34, no. 5, pp. 2119–2125, Oct. 2004.
- [17] C. Wohler and J. K. Anlauf, "An adaptable time-delay neural-network algorithm for image sequence analysis," *IEEE Trans. Neural Netw.*, vol. 10, no. 6, pp. 1531–1536, Nov. 1999.
- [18] H. Zhang, W. Huang, Z. Wang, and T. Chai, "Adaptive synchronization between two different chaotic systems with unknown parameters," *Phys. Lett. A*, vol. 350, no. 5–6, pp. 363–366, Feb. 2006.

A CMOS Classifier Circuit Using Neural Networks With Novel Architecture

Merih Yıldız, Shahram Minaei, and İzzet Cem Göknar

Abstract—In this letter, complementary metal–oxide–semiconductor (CMOS) implementation of a neural network (NN) classifier with several output levels and a different architecture is given. The proposed circuit operates in current mode and can classify several types of data. The classifier circuit is designed using a current-voltage converter, an inverter followed by a NOR gate and a voltage-current output stage. Using a 0.35- μ m TSMC technology parameters, SPICE simulation results for a classifier with two inputs are included to verify the expected results.

Index Terms—Classifier, current mode, neural network (NN).

I. INTRODUCTION

The aim of classification is to assign an unknown object to a class containing similar objects. Classifiers find applications in various fields

Manuscript received November 23, 2006; revised March 9, 2007 and April 18, 2007; accepted May 1, 2007. This work was supported by the Scientific & Technological Research Council of Turkey (TÜBİTAK) under a Project 106E139.

The authors are with the Department of Electronics and Communication Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul, Turkey (e-mail: sminaei@dogus.edu.tr; cgoknar@dogus.edu.tr).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2007.902961

of applied science such as medical diagnosis, pattern recognition, artificial intelligence, real-time object recognition, neural networks (NNs), and statistics [1]–[5]. Whereas the literature abounds with soft algorithms, hard classifiers are seldom encountered.

The basic building block of a feedforward artificial neural network (FANN) is a simple processing element called neuron. The response of each neuron in an FANN is the output of a block with nonlinear transfer characteristic, called activation function, generally in the shape of a sigmoid which is monotonically nondecreasing and is bounded by finite upper and lower limits [6]; typically, they are the hard limiter characterized by the signum function, its piecewise linear version, or its continuously differentiable version expressed with the exponential function. Also, neuron activation functions can be used in adaptive processor architectures to extend the linear region of the input signal and improve system performance [7]. The classical neuron architecture (CNA) with trapezoidal activation function (TAF) is shown in Fig. 1(a). Due to TAF used in this classical architecture, only binary outputs can be obtained. The decision regions obtained from a two-input (2-D) CNA are shown in Fig. 1(b).

In this letter, a 1-D classifier (core cell) shown in Fig. 2(a) with transfer characteristic as in Fig. 2(b) is used to realize an n-dimensional classifier as shown in Fig. 3. It is well known that shrinking bias voltages make it difficult to process data in voltage mode. Therefore, current-mode processing is preferred in which output currents can be easily added by connecting the output terminals of the blocks without requiring the use of extra active blocks.

Although the classifier architecture deployed in Fig. 3 looks somewhat different than the classical FANN, functionally the configuration behaves exactly like an FANN as far as its input–output (I–O) behavior is concerned with the weight of each synapse being the height of the transfer characteristic. The input current I_{in} in Fig. 2(a) is the 1-D data and I_{out} is the output of the classifier. The transfer characteristic of the block in Fig. 2(b) is similar to a TAF [8] which is used in piecewise linear (PWL) approximation, high-speed folding analog-to-digital converters, fuzzy controllers, etc.

The stability properties of NNs with TAF have been investigated in [9]. Neurons with double threshold activation function have been investigated and used to classify data separable by two parallel hyperplanes [10].

The primary aim of this letter is to develop a classifier circuit with n inputs and externally tunable mesh-grid decision regions. Toward that end, an analog complementary metal–oxide–semiconductor (CMOS) current-mode realization of a 1-D "classifier" circuit (core neuron) is realized. Also, a classifier circuit with two inputs is simulated to show the output behavior of a 2-D classifier circuit with a mesh-grid partitioned domain. The core of the proposed circuit is the block shown in Fig. 2(a) with the direct current (dc) transfer characteristic of Fig. 2(b). The horizontal position, width, and height of the transfer characteristic can be adjusted independently by means of external currents I_1 , I_2 , I_{H1} and I_{H2} . Using several of these proposed core circuits, a multidimensional, multilevel-output (each level corresponding to the coding of a data class) classifier can be obtained.

The organization of this letter is as follows. In Section II, the block diagram description and the CMOS design methodology of the core circuit are given and, in Section III, the classifier consisting of several core circuits and its basic operating principles are explained. In Section IV, SPICE simulation results of the classifier are shown to be in conformity with expectations. Finally, in Section V, the mesh-grid classification regions together with the output encoding of the input data are presented; some conclusions are given in Section VI.

1045-9227/\$25.00 © 2007 IEEE

X₂

0.8

0.6 П 4

0.2

08

X₁

2

ο ο

(b)



Fig. 1. (a) Block diagram of CNA with TAF. (b) CNA decision regions for two inputs.



Fig. 2. (a) Core circuit block diagram. (b) Core circuit transfer characteristic.

II. CMOS REALIZATION OF THE CORE CIRCUIT

The I-O transfer characteristic of the core circuit in Fig. 1(b) can be expressed as

$$I_{\rm out} = \begin{cases} I_{H1}, & \text{for } I_1 < I_{\rm in} < I_2 \\ -I_{H2}, & \text{otherwise} \end{cases}$$
(1)

The currents I_{H1} and I_{H2} , as shown in Fig. 2(b), are the positive and negative heights of the output current. The currents I_1 and I_2 are used to shift the horizontal position and adjust the width of the output current. The proposed block diagram for realizing the core circuit with transfer characteristic of Fig. 2(b) is shown in Fig. 4. It consists of a current-to-voltage input stage with two different outputs V_1 and V_2 , an inverter, a NOR gate and a voltage-to-current output stage.

The input stage and the inverter of the core are shown in Fig. 5. The diode-connected transistor M_1 and the bias current I_{BIAS} form a current-to-voltage converter. The transistors M_2 and M_3 are biased with different currents $I_{\text{BIAS}} + I_1$ and $I_{\text{BIAS}} + I_2$, respectively, to form two different threshold values of the transfer characteristic. Note that



Fig. 3. Block diagram of an n-dimensional classifier.

TABLE I DIMENSIONS OF THE MOS TRANSISTORS

| MOSFET | W [µm] | L [µm] |
|--|---------------|--------|
| | 10.5 | 0.7 |
| M ₅ , M ₁₁ , M ₁₃ , | 5.25 | 0.7 |

all of the bias currents I_{BIAS} , $I_{\text{BIAS}} + I_1$, and $I_{\text{BIAS}} + I_2$ are obtained with bias circuits consisting of simple current mirrors.

The transistors M_4 and M_5 constitute an inverter which is used to obtain an I–O characteristic with positive jump at threshold current I_2 , as shown in Fig. 6. Considering $I_{D(M2)} = I_{\text{BIAS}} + I_1$ and $I_{D(M3)} = I_{\text{BIAS}} + I_2$, the following expressions can be written:

$$V_{1} = \begin{cases} V_{SS}, & I_{\rm in} > I_{1} \\ V_{DD}, & I_{\rm in} < I_{1} \end{cases}$$
(2)

$$V_{2} = \begin{cases} V_{SS}, & I_{\rm in} > I_{2} \\ V_{DD}, & I_{\rm in} < I_{2} \end{cases}$$
(3)



Fig. 4. Block diagram of the proposed core circuit.



Fig. 5. Input stage of the core circuit with the inverter.



Fig. 6. $V_1 - I_{in}$ and $V_2' - I_{in}$ transfer characteristics for the circuit of Fig. 5.



Fig. 7. The NOR gate and the output stage of the core circuit.

$$V_2' = \begin{cases} V_{DD}, & I_{\rm in} > I_2 \\ V_{SS}, & I_{\rm in} < I_2 \end{cases}.$$
 (4)

The NOR gate and the output stage of the proposed core circuit are given in Fig. 7. The desired part of the transfer characteristics shown in Fig. 6 is obtained by the NOR gate (transistors M_6 , M_7 , M_8 , and M_9)







Fig. 9. $I_{\rm out} - I_{\rm in}$ characteristic of the core circuit.



Fig. 10. $I_{out} - I_{in}$ characteristic of the 1-D multilevel classifier circuit.

as follows: if both inputs of the NOR gate are low level, then the output of the gate is high; otherwise, the output is low. In fact, $V_{\rm out}$ can be written as

$$V_{\rm out} = \begin{cases} V_{DD}, & \text{for } I_1 < I_{\rm in} < I_2 \\ Vss, & \text{otherwise} \end{cases}$$
(5)

The location of the high output portion is shifted with the currents I_1 and I_2 . This shift of the intercept points by adjusting the starting

Authorized licensed use limited to: ULAKBIM UASL - DOGUS UNIV. Downloaded on August 13,2010 at 13:52:17 UTC from IEEE Xplore. Restrictions apply.



Fig. 11. Block diagram of a 2-D multilevel classifier circuit.



 $\mathbf{I}_{\mathbf{H}}(\mathbf{H}\mathbf{A})$

Fig. 12. $I_{out} - (I_{in1} - I_{in2})$ characteristics of the proposed 2-D multilevel classifier.

| CORE CIRCUIT CURRENTS | | | | | |
|-----------------------|---------------------|--------------|--|--|--|
| uit (CC) | I ₁ (μA) | $I_2(\mu A)$ | | | |
| | | | | | |

Core Ciro

TABLE II

| | -1 (F) | -2 () | -11 ()) |
|-----|-----------------|-------|---------|
| CC1 | 20 | 80 | 130 |
| CC2 | 130 | 180 | 90 |
| CC3 | 240 | 320 | 100 |
| CC4 | 380 | 490 | 150 |
| | | | |

and ending boundary points of the specified decision region makes the device custom tunable.

The current output stage of the circuit is obtained by connecting two complementary source-coupled pairs [11]; currents I_{H1} and I_{H2} determine the peak values of the function.

The current relations for the input stage in Fig. 5 are $I_{DM11} + I_o^- =$ $I_{DM10}, I_{DM13} + I_o^+ = I_{DM12}, I_{H1} = I_{DM10} + I_{DM12}$, and $I_{H2} =$ $I_{DM11} + I_{DM13}.$

The output currents I_o^+ and I_o^- can be given in terms of V_{out} as follows:

$$I_{o}^{+} = \begin{cases} I_{H1}, & \text{for } V_{\text{out}} = V_{DD} \\ -I_{H2}, & \text{for } V_{\text{out}} = V_{SS} \end{cases}$$
(6)

$$I_{o}^{-} = \begin{cases} I_{H1}, & \text{for } V_{\text{out}} = V_{SS} \\ -I_{H2}, & \text{for } V_{\text{out}} = V_{DD} \end{cases}.$$
 (7)

It should be noted that, in Section III, the current I_o^+ is used as the output current I_{out} , the current I_{H2} is chosen as 0, and the current I_{H1} denoted by I_H is used to change the height of the transfer characteristic. In this case, the low level of the output is 0 and the high level of the output is an adjustable current equal to I_H .

III. PROPOSED MULTILEVEL CLASSIFIER CIRCUIT

The transfer characteristic of a multilevel classifier circuit is shown in Fig. 8. To classify different types of data, the proposed core circuits' outputs can be connected "in parallel."

In order to achieve the I-O characteristic shown in Fig. 8, the following constraints must be satisfied:

$$I_{\text{in}\,1} = I_{\text{in}\,2} = \ldots = I_{\text{in}\,N} = I_{\text{in}\,}, \ I_1 < I_2 < I_3 < \ldots < I_{(2N-1)} < I_{2N}.$$
(8)

Note that if these input currents are not chosen equal, then more complex decision regions as discussed in Section V can be obtained.

IV. SIMULATION OF THE CORE CIRCUIT

In the following applications, the proposed core circuit is simulated using SPICE with 0.35- μ m TSMC CMOS technology parameters. The voltage supplies and the bias current (I_{BIAS}) in this circuit are selected as ± 1.25 V and 10 μ A, respectively. The dimensions of the transistors are given in Table I. Taking $I_1 = 110 \ \mu\text{A}$, $I_2 = 310 \ \mu\text{A}$, and $I_H = 80$ μA results in the $I_{\rm out} - I_{\rm in}$ characteristic for the circuit as shown in Fig. 9. To obtain a "current-input–voltage-output" characteristic, $V_{\rm out}$ can also be used as output.

To construct a single-input (1-D) multilevel classifier, we use four core blocks connected in parallel with the same input current (I_{in}) and four different sets of control currents. The selected classification currents I_1 , I_2 , and I_H for each core circuit are shown in Table II. SPICE

| | | | | | | | - |
|----------------|----------------|-----------------|----------------------|----------------|-------|-----------------|----------------------|
| | Core Circuit-I | | | | Cor | e Circu | it-II |
| I ₁ | I ₂ | I _{H1} | Power Dissipation | I ₃ | I4 | I _{H2} | Power Dissipation |
| 70 | 140 | 60 | 0.35mW | 210 | 280 | 120 | 0.77mW |
| 10 | 20 | 60 | 0.12mW | 30 | 50 | 70 | 0.20mW |
| | Co | re Circ | uit-III | | Cor | e Circui | it-IV |
| I5 | I ₆ | I _{H3} | Power Dissipation | I_7 | I_8 | $I_{\rm H4}$ | Power Dissipation |
| 60 | 140 | 100 | 0.39mW | 280 | 360 | 80 | 0.91mW |
| 15 | 35 | 160 | 0.27mW | 100 | 120 | 65 | 0.37mW |

 TABLE III

 Core Circuit Currents (Currents in Microamperes)



Fig. 13. Different output levels in a 2-D mesh-grid classifier.

simulation results are shown in Fig. 10; 7-ns delay was observed at the output of the core cell for step input.

The block diagram for realizing a 2-D multilevel classifier with twoinput currents (I_{in1}, I_{in2}) is shown in Fig. 11. The control currents and the power dissipation for this multilevel classifier are given in Table III together with smaller control currents to compare the effect on power consumption in the same circuit; as expected, the consumption has been reduced almost by one third.

SPICE simulation results for two different inputs to the classifier circuit are shown in Fig. 12 which contains nine different output values as opposed to the binary output in Fig. 1(b) of the classical NN. This kind of configuration can classify 2-D data into eight different types of classes (nine counting the $I_{out} = 0$ plane), each type being encoded with a different output current value, whereas a classical NN can only classify into two groups.

V. CLASSIFICATION REGIONS AND ENCODING

The proposed block diagram of Fig. 11 can be generalized in the following three different ways.

- 1) By applying the same input current I_{in1} and different control currents to *m* core blocks, the same input currents I_{in2} and different control currents to *n* core blocks and connecting all outputs in parallel; in both cases, the control currents must satisfy the inequalities of expression (8). The resulting circuit will then allocate 2-D input data into m(n+1) + n + 1 different classes (counting the $I_{out} = 0$ plane). The different output code levels (height of the outputs) in Fig. 13 for each data class are shown inside each region where ΔI is the increment necessary for distinguishing outputs closest in value.
- 2) By applying different input currents to each of the n core blocks and connecting them all in parallel. This will produce an n-dimensional classifier with one decision interval for each data class.

 Using any combination of 1) and 2), thus creating a multidimensional classifier with more than one decision region for each type of data.

This multilevel output-encoding of each class of data can be viewed as the problem of assigning weights (I_{H1} and I_{H2} of each core circuit) to the synapses of each neuron in a classical NN in order to perform the desired classification.

VI. CONCLUSION

In this letter, a new architecture for an NN has been introduced and a current mode CMOS-only circuit consisting of an array of core neurons which mimics an FANN that acts as a classifier has been proposed; it classifies data separable by mesh-grid hyperplanes.

The core circuits have independently adjustable height and width of their transfer characteristics as well as horizontal position; these properties allow field-encoding of the proposed classifier output for (more than two) different data classes as opposed to the binary output in classical NN. This property makes the new architecture very suitable for analog-to-digital converter (ADC) applications and will be elaborated in a future work. Another possible extension will be to preprocess input data in order to transform the mesh-grid partitioning of the data domain into one separated by arbitrary hyperplanes. A further extension for multiple-input–multiple-output (MIMO) classifiers, in order to facilitate access and reduce I–O pins, would be to embed a serial-input parallel-output memory-like unit into the design for adjusting the control parameters.

Designed CMOS circuits have been verified with SPICE simulation results. In the literature, some voltage mode [12] or voltage-input/current-output mode FANN circuits using the trapezoidal activation function have been reported; these circuits are not suitable for low-voltage operation and are not designed in current mode. The circuit proposed in this letter operates at reasonably low supply voltages and consumes low power. Moreover, the parallel processing ability of the circuit makes it well suited for real-world applications.

REFERENCES

- B. Liu, C. Chen, and J. Tsao, "A modular current-mode classifier circuit for template matching application," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 47, no. 2, pp. 145–151, Feb. 2000.
- [2] E. Hunt, Artificial Intelligence. New York: Academic, 1975.
- [3] H. S. Abdel-Aty-Zohdy and M. Al-Nsour, "Reinforcement learning neural network circuits for electronic nose," in *Proc. IEEE Int. Symp. Circuits Syst.*, Orlando, FL, 1999, vol. 5, pp. 379–382.
- [4] G. Lin and B. Shi, "A current-mode sorting circuit for pattern recognition," in *Proc. 2nd Int. Conf. Intell. Process. Manuf. Mater.*, Honolulu, HI, Jul. 10–15, 1999, pp. 1003–1007.
- [5] G. Lin and B. Shi, "A multi-input current-mode fuzzy integrated circuit for pattern recognition," in *Proc. 2nd Int. Conf. Intell. Process. Manuf. Mater.*, Honolulu, HI, Jul. 10–15, 1999, pp. 687–693.
- [6] C. Fausto and V. Maurizio, "A mixed mode perceptron cell for VLSI neural networks," in *Proc. IEEE Int. Conf. Electron., Circuits, Syst.* (*ICECS*), 2001, pp. 377–380.
- [7] G. Zatorre-Navarro, N. Medrano-Marques, and S. Celma-Pueyo, "Analysis and simulation of a mixed-mode neuron architecture for sensor conditioning," *IEEE Trans. Neural Netw.*, vol. 17, no. 5, pp. 1332–1335, Sep. 2006.
- [8] E. Bilgili, İ. C. Göknar, and O. N. Uçan, "Cellular neural networks with trapezoidal activation function," *Int. J. Circuit Theory Appl.*, vol. 33, no. 5, pp. 393–417, 2005.
- [9] D. Y. Aksın, S. Aras, and İ. C. Göknar, "CMOS realization of user programmable, single-level, double-threshold generalized perceptron," in *Proc. Turkish Artif. Intell. Neural Netw. Conf.*, İzmir, Turkey, Jun. 21–23, 2000, pp. 117–125.
- [10] M. Yıldız, S. Minaei, and C. Göknar, "Current mode double threshold neuron activation function," in *Complex Computing-Networks: Brain-Like and Wave-Oriented Electrodynamic Algorithms*. New York: Springer-Verlag, 2006, pp. 267–274.

- [11] A. F. Arbel and L. Goldminz, "Output stage for current-mode feedback amplifier, theory and applications," *Analog Integrated Circuits Signal Process.*, vol. 2, no. 3, pp. 243–255, 1992.
- [12] M. Kachare, J. Ramirez-Angulo, R. Carvajel, and A. J. Lopez-Martin, "New low-voltage fully programmable CMOS triangular/trapezoidal function generator circuits," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 10, pp. 2033–2040, Oct. 2005.

Stability Analysis for Neural Networks With Time-Varying Interval Delay

Yong He, G. P. Liu, D. Rees, and Min Wu

Abstract—This letter is concerned with the stability analysis of neural networks (NNs) with time-varying interval delay. The relationship between the time-varying delay and its lower and upper bounds is taken into account when estimating the upper bound of the derivative of Lyapunov functional. As a result, some improved delay/interval-dependent stability criteria for NNs with time-varying interval delay are proposed. Numerical examples are given to demonstrate the effectiveness and the merits of the proposed method.

Index Terms—Delay/interval-dependent stability, neural networks (NNs), time-varying interval delay.

I. INTRODUCTION

In the past few decades, neural networks (NNs) have received increasing interest owing to their applications in a variety of areas, such as signal processing, pattern recognition, static image processing, associative memory, and combinatorial optimization [1]. Up to now, stability of NNs with a time delay has also received attention [2]–[26] since time delay is frequently encountered in NNs, and it is often a source of instability and oscillations in a system. Both delay-independent [2]–[16] and delay-dependent [17]–[26] stability criteria for NNs have been proposed in recent years. Since delay-independent criteria tend to be conservative, especially when the delay is small or it varies in an interval, much attention has been paid to the delay-dependent type.

Recently, the free-weighting matrix approach proposed in [27]–[29], which is very effective as the bounding techniques on some cross

Manuscript received December 7, 2006; revised March 16, 2007; accepted May 21, 2007. This work was supported in part by the Program for New Century Excellent Talents in University under Grant NCET-06-0679, the National Science Foundation of China under Grants 60425310, 60528002, and 60574014, the Doctor Subject Foundation of China under Grant 20050533015, and the Leverhulme Trust in the United Kingdom.

Y. He is with the School of Information Science and Engineering, Central South University, Changsha 410083, China and the Faculty of Advanced Technology, University of Glamorgan, Pontypridd CF37 1DL, U.K. (e-mail: heyong08@yahoo.com.cn).

G. P. Liu is with the Faculty of Advanced Technology, University of Glamorgan, Pontypridd, CF37 1DL, U.K. and with the Laboratory of Complex Systems and Intelligence Science, Chinese Academy of Sciences, Beijing 100080, China.

D. Rees is with the Faculty of Advanced Technology, University of Glamorgan, Pontypridd CF37 1DL, U.K.

M. Wu is with the School of Information Science and Engineering, Central South University, Changsha 410083, China (e-mail: min@mail.csu.edu.cn).

Digital Object Identifier 10.1109/TNN.2007.903147

product terms are not involved in the approach, was employed in [20] to establish the delay-dependent stability criteria for NNs with multiple time-varying delays. On the other hand, an alternative criterion was derived for NNs with single time-varying delay in [25] by introducing a new Lyapunov functional which is similar to [30]. In order to overcome the conservativeness, [26] introduced the new free-weighting matrices to estimate the upper bound of the derivative of Lyapunov functional without ignoring any negative quadratic terms and some improved delay-dependent stability criteria are established for NNs with time-varying delay. However, there is room for further investigation. For example, the delay term d(t) with $0 \le d(t) \le h$ was enlarged as h and another term h - d(t) was also regarded as h in [26], that is, h = d(t) + (h - d(t)) was enlarged as 2h; so, the aforementioned treatment may lead to a conservative result.

On the other hand, the range of time-varying delay for NNs considered in [20], [21], [25], and [26] is from 0 to an upper bound. In practice, a time-varying interval delay is often encountered, that is, the range of delay varies in an interval for which the lower bound is not restricted to 0. In this case, the stability criteria for NNs with time-varying delay in [20], [21], [25], and [26] are conservative because they do not take into account the information of the lower bound of delay. To the best of the authors' knowledge, few stability results have been reported in the literature for NNs with time-varying interval delay.

In this letter, the stability problem for NNs with time-varying interval delay is taken into account. A new method that considers the relationship between the time-varying delay and its lower and upper bounds is proposed when estimating the upper bound of the derivative of Lyapunov functional. Less conservative delay/interval-dependent stability criteria for NNs with time-varying interval delay are presented. Numerical examples are given to demonstrate the effectiveness and the benefits of the proposed method.

Notation

Throughout this letter, the superscripts "-1" and "T" stand for the inverse and transpose of a matrix, respectively; R^n denotes the *n*-dimensional Euclidean space; $R^{n \times m}$ is the set of all $n \times m$ real matrices; P > 0 means that the matrix P is positive definite; I is an appropriately dimensioned identity matrix; diag {...} denotes a block-diagonal matrix; and the symmetric terms in a symmetric matrix are denoted by \star , e.g.,

$$\begin{bmatrix} X & Y \\ \star & Z \end{bmatrix} = \begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix}$$

II. PROBLEM FORMULATION

Consider the following NN with a time-varying delay:

$$\dot{x}(t) = -Ax(t) + W_0 g(x(t)) + W_1 g(x(t - d(t))) + u \quad (1)$$

where $x(\cdot) = \begin{bmatrix} x_1(\cdot) & x_2(\cdot) & \cdots & x_n(\cdot) \end{bmatrix}^T \in \mathbb{R}^n$ is the neuron state vector, $g(x(\cdot)) = \begin{bmatrix} g_1(x_1(\cdot)) & g_2(x_2(\cdot)) & \cdots & g_n(x_n(\cdot)) \end{bmatrix}^T \in \mathbb{R}^n$ denotes the neuron activation function, and $u = \begin{bmatrix} u_1 & u_2 & \cdots & u_n \end{bmatrix}^T \in \mathbb{R}^n$ is a constant input vector. $A = \text{diag}\{a_1, a_2, \dots, a_n\}$ is a diagonal matrix with $a_i > 0$, $i = 1, 2, \dots, n$, and W_0 and W_1 are the connection weight matrix and the delayed connection weight matrix, respectively. The time delay d(t) is a time-varying differentiable function that satisfies

$$0 \le h_1 \le d(t) \le h_2 \tag{2}$$

$$\dot{d}(t) \le \mu \tag{3}$$

1045-9227/\$25.00 © 2007 IEEE

A flexible current-mode classifier circuit and its applications

Merih Yıldız, Shahram Minaei and İzzet Cem Göknar*, †

Department of Electronics and Communications Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul, Turkey

ABSTRACT

In this paper a new CMOS classifier circuit is presented, simulated, and compared with other recently introduced circuits. The proposed CMOS circuit operates in current-mode and can classify several types of data. The architecture is designed using two threshold circuits and a subtraction circuit. Among many possible applications of the classifier circuit, template-based pattern classification, namely template matching and character recognition with corruption, and in another direction its use as a quantizer are given. Using $0.35-\mu$ m AMS technology parameters, SPICE simulations as well as hard realization results for the classifier and application circuits are included; detailed Monte Carlo analyses to assess parameter mismatch effects are also performed. Copyright © 2010 John Wiley & Sons, Ltd.

KEYWORDS

classifier; character recognition; template matching; quantizer; CMOS

Correspondence

* izzet Cem Göknar, Department of Electronics and Communications Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul, Turkey.

[†]E-mail: cgoknar@dogus.edu.tr

Contract/grant sponsor: Scientific & Technological Research Council of Turkey (TÜBİTAK)

Received 25 April 2008; Revised 23 July 2009; Accepted 8 December 2009

1. INTRODUCTION

Classification is a very important topic in many applications, such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks, statistics, and template matching [1-6]. The aim of this classification is to assign an object under consideration into a class containing similar objects or to classify unknown patterns. To this purpose many classification algorithms based on concepts, such as Euclidean distance, cross correlation, K-nearest neighbor, crisp c-means, have been proposed. However, the processing is computationally very expensive, consuming a lot of CPU time when implemented as software running on general purpose computers, whereas the literature abounds with soft algorithms, hard classifiers are seldom encountered [5-7]. A number of soft techniques have been investigated with the intent of speeding up the process [8-12] providing little practical use for real-time applications [13]. In order to speed up these algorithms it is desirable to have them implemented in hardware; in the literature some researchers have designed Euclidean distance and K-nearest neighbor calculators used for classification [1, 2]

The primary aim of this paper is to develop a flexible and yet simple classifier circuit which can be used for hardimplementing several techniques, such as pattern classification, template matching, quantization, and to demonstrate its use through applications.

Character recognition and pattern classification using template matching techniques are powerful tools for many data processing systems. Template matching is a technique in image processing for finding small parts of an image which match a given template pattern [12]; it can be used in manufacturing as part of quality control, in robot vision as a way to navigate a mobile robot, as a way to detect edges in images, etc. The basic method in template matching is to loop through all the pixels in the search image and compare them to the pattern. Another approach to make the matching faster is to divide the image into smaller images, and then search the smaller subimages. After finding matches in the smaller images, the obtained information will be used in the larger image. Whereas template matching tools Quantization is the first step for converting an analog signal to a digital one and needs no further introduction. The classifier circuit developed in this paper can also be used as a quantizer.

The aim of this paper is threefold: (i) to design a simpler and better performing flexible classifier circuit, (ii) to illustrate the use of the circuit in character recognition, template matching and quantizer applications, and (iii) to compare the newly introduced circuit with the ones previously introduced.

The paper is organized as follows. In Section 2 a new current-mode CMOS Core Circuit (Core Cell) (CC) using the threshold gate introduced in [14] is proposed; as a byproduct a deficiency of this gate is exhibited and corrected. Moreover, how an $n \times m$ -dimensional classifier circuit with *n* inputs and *m* outputs using CCs connected in parallel can be realized is specified. Character recognition, template matching in the presence of corruption and faults, and quantizer applications of the proposed classifier circuit are presented in Section 3. SPICE simulations, hardware implementation results of the proposed CC and its applications are given in Section 4 along with performance comparison with previously proposed CCs. Section 5 concludes the paper.

2. CMOS CORE CIRCUIT

The block diagram of a CC and its transfer characteristic are shown in Figure 1(a),(b), respectively.

The input–output (I–O) transfer characteristic shown in Figure 1(b) can be expressed as:

$$I_{\text{out}} = \begin{cases} I_{\text{H}} & \text{if } I_1 < I_{\text{in}} < I_2 \\ 0 & \text{otherwise} \end{cases}$$
(1)

The horizontal position, width, and height of the transfer characteristic can be adjusted independently by means of external control currents I_1 , I_2 , and I_H . The current I_H determines the amplitude of the output waveform, currents I_1 and I_2 are used to shift the horizontal position of the output waveform and adjust its width. As shown in Figure 2, the



Figure 1. (a) CC block diagram and (b) transfer characteristic of the CC.

Int. J. Circ. Theor. Appl. (2010) © 2010 John Wiley & Sons, Ltd. DOI: 10.1002/cta

CC can be constructed using two threshold circuits and a current mirror which acts as a subtractor.

There are many techniques in the literature for CMOS realization of current comparator and threshold circuits [14, 15]. The I–O characteristic of the threshold circuit is shown in Figure 3.

The threshold circuit given in Figure 4 has been investigated and tested before being used in the classifier circuit. Constructed with CD4007 MOS array transistors for experimental testing, the threshold circuit has exhibited a hysteresis characteristic as shown in Figure 5, observed by a digital oscilloscope operating in X-Y mode. SPICE simulations of the threshold circuit executed first for increasing then decreasing values of the input produced the same hysteresis effect as shown in Figure 6. This hysteresis effect is undesirable for classifier applications as it may lead to misclassifications and should be eliminated. Removal of the transistor M₃ that forms a positive feedback path, thereby causing the hysteresis effect, produced the desired I–O characteristic in Figure 3; the modified circuit is shown in Figure 7.

The current sources I_{in} , I_1 , and I_H in Figure 7 are applied to the circuit through simple current mirrors. All the transistors M_1-M_6 are identical. The current I_1 is the threshold value of the circuit. If the input current I_{in} is smaller than the threshold current I_1 , the drain voltage of the transistor M_2 is approximately V_{DD} , and the current I_H flows through M_4 , so the output current is zero. When the current I_{in} exceeds the threshold current I_1 , the drain





Figure 3. Input-output characteristic of the threshold circuit.



Figure 4. The threshold circuit reported in [15].



Figure 5. Experimental characteristic of the threshold circuit in Figure 4.

voltage of the transistor M_3 becomes approximately V_{SS} , the current flowing through M_4 is zero, and the current I_H flows through M_5 , so the output current becomes equal to I_H .

The I-O characteristic of the circuit can be expressed as:

$$I_{\text{out}} = \begin{cases} I_{\text{H}} & \text{if } I_{\text{in1}} \ge I_{1} \\ 0 & \text{otherwise} \end{cases}$$
(2)

To realize a CC, two threshold circuits and a subtractor circuit are used as shown in Figure 8. Thus the classification region is obtained by taking the difference of the output currents of the threshold circuits yielding the transfer characteristic as shown in Figure 1(b). Note that the current $I_{\rm H}$ of the threshold circuits should be equal and the following constraint must be satisfied

$$I_2 > I_1 \tag{3}$$

In Figure 8 the transistors M_1-M_6 and M_9-M_{14} constitute two threshold circuits, respectively.

The basic current mirror constructed with the transistors M_7 and M_8 performs the desired operation of subtraction. The transistors M_{15} , M_{16} , M_{17} are used to provide currents equal to I_H (adjusting the output level) for the threshold circuits. Similarly, the same approach is used with M_{18} , M_{19} , and M_{20} to apply the input current I_{in} to both of the threshold circuits.

By interconnecting several CCs and adding their output currents, a multi-input classifier ($n \times 1$ -dimensional) can be obtained as shown in Figure 9 [16]. The current I_{in-k} (k=1, 2,...,n) is the input current of the k-th CC and the current I_{out} is the output of the classifier. For an $n \times m$ -dimensional classifier $n \times m$ CCs may be combined (as in Figure 12 of the next section) to form m groups and the outputs in each group added to form one of the classifier's outputs

3. APPLICATIONS OF THE CLASSIFIER CIRCUIT

3.1. Character recognition and template matching

Character recognition is a special case of pattern classification and template matching (which is a classical technique of classifying subimages inside a large image) is in turn a special case of character recognition; in fact a template is the subimage of a larger image. The subimage can be a character, a special picture, or a number. Here, the application is constructed for multiple subimages assuming that the templates are chosen as decimal numbers as shown in Figure 10(a). Each subimage in the template is divided into $4 \times 5 = 20$ cells as illustrated in Figure 11; the reason for the extra blank column and the special position of the numbers is to provide a special coding needed for fault diagnosis

The template matching classifier topology is designed to compare the sample template cells with other template cells as shown in Figure 12. The block diagram shown in Figure 12 is composed of 4×1 -**D**imensional **M**ultilevel Classifiers (FDMC). Each FDMC with same output function $f(x_1, x_2, x_3, x_4)$ given by (4) consists of four core cells connected in parallel in the form shown in Figure 9 with output currents connected to the same node and the classifier inside the dashed box in Figure 12 is 20×5 -dimensional. The output of each FDMC is applied to a single CC with function 'g(y)'. Values for $a_i, b_i, c_i, d_i(i=1, ..., 5)$ are selected as 10μ A for logic '1' and 1μ A for '0' depending on the applied input pattern for binary images, can take intermediate values for multilevel inputs. The output of the FDMC can be expressed as:

$$f(x_1, x_2, x_3, x_4) = x_1 2^0 + x_2 2^1 + x_3 2^2 + x_4 2^3$$
(4)

This choice in (4) for $f(x_1, x_2, x_3, x_4)$ is to ensure that every combination of a_i , b_i , c_i , d_i provides at the output, information about the row of the applied template. Each row of the sample template is applied to the input of the corresponding FDMC in the order given in Figure 12. Depending on which template from Figure 10 is applied, the output of each FDMC block (which changes according to (4)) is given in Table I and it can be observed that none of the five-dimensional outputs coincides with the other. So by checking the outputs $y_i(i=1, ..., 5)$ according to Table I, the input pattern can be determined, and hence character recognition can be achieved at this level; in case a single output is desired, these outputs can be added (a single node will suffice in case they are currents) as the sum Σ is different for each input template.

To further realize template matching, single CCs with transfer characteristic $g_i(y_i)$ (i=1,...,5) are introduced as shown in Figure 12; the control currents of the CCs in Figure 1(b) are chosen appropriately to indicate the selected template. If the input of the core cell block $g_i(y_i)$ is in the correct region the outputs z_i will be digitally '1' otherwise '0'. So if a template to be matched is applied to the inputs and the applied template is correct then the output z of the



Figure 6. Simulated characteristic of the threshold circuit in Figure 4.



Figure 7. The modified threshold circuit.



Figure 8. CMOS implementation of the CC.



Figure 9. Block diagram of an *n*-dimensional classifier.





| a ₁ | b ₁ | c ₁ | d ₁ |
|-----------------------|-----------------------|-----------------------|-----------------------|
| a ₂ | b ₂ | c_2 | d ₂ |
| a 3 | b ₃ | c ₃ | d ₃ |
| a4 | b_4 | c ₄ | d ₄ |
| a ₅ | b ₅ | c ₅ | d ₅ |

Figure 11. Cell arrangement of the template.



Figure 12. Template matching topology constructed with classifiers and core cells.

Table I. The outputs y_i (i = 1, ..., 5) of different FDMCs and their sum.

| Templates | $\gamma_1 \ (\mu A)$ | $\gamma_2~(\mu A)$ | $y_3~(\mu A)$ | $y_4~(\mu A)$ | $y_5~(\mu A)$ | $\sum (\mu A)$ |
|-----------|----------------------|--------------------|---------------|---------------|---------------|----------------|
| 0 | 7 | 5 | 5 | 5 | 7 | 29 |
| 1 | 8 | 8 | 8 | 8 | 8 | 40 |
| 2 | 14 | 8 | 14 | 2 | 14 | 52 |
| 3 | 14 | 8 | 12 | 8 | 14 | 56 |
| 4 | 10 | 10 | 14 | 8 | 8 | 50 |
| 5 | 7 | 1 | 7 | 4 | 7 | 26 |
| 6 | 1 | 1 | 7 | 5 | 7 | 21 |
| 7 | 7 | 4 | 4 | 4 | 4 | 23 |
| 8 | 14 | 10 | 14 | 10 | 14 | 62 |
| 9 | 7 | 5 | 7 | 4 | 4 | 27 |

| MOSFET | $M_1,M_2,M_4,M_5,M_6,M_9,M_{10}M_{12},M_{13},M_{14}$ | $M_{7},M_{8},M_{15},M_{16},M_{17},M_{18},M_{19},M_{20},M_{25},M_{26},M_{27},M_{28}$ |
|--------|--|---|
| | 10.5 | 35.5 |
| L (µm) | 1.05 | 1.05 |

Table II. Dimensions of the MOS transistors in the circuit of Figure 8.

classifier will be equal to '5' in magnitude; if it is not correct it will be less than '5'.

It should be observed from Table I that for any two different inputs the outputs y_i may match at most for three i = 1, 2, ..., 5 and not four, thus providing the basis for **fault diagnosis**. In fact if there is an error in one line of the templates in Figure 10 at one or more pixels, only four of the outputs z_i will be '1' and z will be '4' in magnitude. For error correction purposes, a final CC with transfer characteristic g(z) having properly chosen threshold values is added. This CC will produce an output '1' in magnitude if its input magnitude is 4 or 5 and a '0' output otherwise, correctly identifying the error-containing template to be matched.

4. SIMULATION AND EXPERIMENTAL RESULTS

4.1. Threshold and core circuits

The threshold and the core circuits in Figures 7 and 8 have been simulated with 0.35- μ m AMS CMOS technology parameters using the SPICE program. The supply voltages were selected as ± 1.65 V. The transistor dimensions are given in Table II.

Simulation result of the threshold circuit shown in Figure 7 is given in Figure 13 for $I_1 = 10 \ \mu$ A and $I_H = 2 \ \mu$ A. The simulation result for the CC of Figure 8 is shown in Figure 14, where $I_1 = 10 \ \mu$ A, $I_2 = 20 \ \mu$ A, and $I_H = 2 \ \mu$ A. The power consumption and the delay of the CC are, respectively, $61 \ \mu$ W and 36 ns which change with the choice for control currents.

Moreover, the proposed CC of Figure 8 has been constructed with CD4007 CMOS array transistors. Supply voltages were chosen as $V_{\text{DD}} = 5 \text{ V}$ and $V_{\text{SS}} = -5 \text{ V}$. The control currents were selected as $I_1 = 85\mu\text{ A}$, $I_2 = 120\mu\text{ A}$, and $I_{\text{H}} = 38\mu\text{ A}$. Using a digital oscilloscope, the I–O characteristic of the hard-realized CC is given in Figure 15.

Several Monte Carlo analyses of the CC for parameter mismatch evaluation was performed for hundred runs by taking L, W, μ_n , μ_p , V_{Tn} , V_{Tp} deviations (as specified by the AMS CMOS technology used) for the transistors. These analyses have been repeatedly performed by incrementing the control currents I_1 , I_2 , and I_H from 10 to 200μ A. The deviations of the control current I_2 from its nominal values which we call Monte Carlo Relative Tracking Error (MC-RTE), are given in Figure 16(a). From Figure 16(a) it can be seen that the deviations of I_2 have a maximum value of $\% \pm 1.1 (\frac{\Delta I_H}{I_H} < \% \pm 0.85$ for I_H). An example of the Monte Carlo analyses for the I–O characteristic of the



Figure 13. The input–output characteristic of the threshold circuit in Figure 7.



Figure 14. Simulated input-output characteristic of the CC.

CC is shown in Figure 16(b) for the choice $I_1 = 30\mu$ A, $I_2 = 150\mu$ A, and $I_H = 20\mu$ A of the control currents. From these analyses it can be observed that parameter mismatch is not of real concern for classification usage of the proposed circuit as far as false classification of data is concerned since choice of amplitudes identifying different data regions is left to the discretion of the user.



Figure 15. Experimental input–output characteristic of the CC (X 50 µA/div, Y 20 µA/div).









Int. J. Circ. Theor. Appl. (2010) © 2010 John Wiley & Sons, Ltd. DOI: 10.1002/cta



Figure 18. (a) Binary 1-line faulty decimal numbers and (b) corrupted and 1-line faulty test number examples



Figure 19. Simulation results of the erroneous test templates applied to classifier circuit.

4.2. Classifier circuit

The classifier circuit has been tested for the template matching application of Figure 10(a); in the simulations, magnitude corresponding to '1' is chosen as 10μ A. To simulate the circuit, from 0 to 1 ms template '0', 1 to 2 ms template '1', and consecutively other templates were applied one by one until 10ms. The simulation results of the template matching application are shown in Figure 17 where F(i) means the control currents are set to classify the template *i*. From Figure 17 it can be seen that from 0 to 10 ms only one section (1 ms period) according to the applied template has the value of $10 \,\mu$ A. The simulation results confirm the correct operation of the application. To further illustrate the grey level usage potential due to the analog nature of the circuit, corrupted decimal numbers were applied as input to the classifier for control currents as indicated in Figure 10(b) the new output values y_i were as given in Table I and the simulation results as in Figure 17.

The circuit has also been simulated for the 1-line faulty templates shown in Figure 18(a) as follows: from 1 to 2 ms template '1', 1 to 2 ms template '2', and other templates are

similarly applied one by one until template '6'. The simulation results of the template matching application are shown in Figure 19 where F(i) means the control currents are set to classify the template *i*. From Figure 19 it can be seen that, although one or more pixels of the test template have errors, the correct template has been recognized. Moreover corrupted and 1-line faulty inputs applied to the classifier for control currents as indicated in Figure 10(b) and the resulting output values y_i were as shown in Figure 19.

4.3. Quantizer circuit

The CC blocks can be used to construct a quantizer circuit. For example, one can connect 32 CCs in parallel with the same input current each equal to I_{in} and 32 different sets of control currents, thus forming a 5-bit quantizer circuit. The control currents I_1 , I_2 , and I_H for each CC are given in Table III. To simulate the circuit as a quantizer a single triangular input waveform is applied to the input of the circuit and the I–O characteristic versus time is shown in Figure 20. It should be observed that quanta width and peak value can

| Table III. CC control | l currents used in th | e quantizer circuit. |
|-----------------------|-----------------------|----------------------|
|-----------------------|-----------------------|----------------------|

| #CC | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| l ₁ (μA) | 0 | 5 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 | 60 | 66 | 72 | 78 | 84 | 90 |
| $I_2 \; (\mu A)$ | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 |
| $I_{H}~(\muA)$ | 0.1 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |
| #CC | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| $I_1 \; (\mu A)$ | 96 | 102 | 108 | 114 | 120 | 126 | 132 | 138 | 144 | 150 | 156 | 162 | 168 | 174 | 180 | 186 |
| I_2 (μA) | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 | 192 |
| $I_{\rm H}~(\mu {\rm A})$ | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |



Figure 20. Iin-Iout characteristic of the 32-level quantizer circuit.

be tuned as required by proper choice of control currents, providing further design flexibility.

4.4. Comparison of different circuit realizations

The core cells given in [1, 6, 16] and the one proposed in this paper are compared from technology parameters, power consumption, supply voltage, and response time point of views in Table IV. The circuit in reference [16] has been re-simulated with the new supply voltages' and new control currents' values so that the comparison with the circuit presented in this work would be more meaningful; same parameter deviations as specified by AMS CMOS technology were used in the Monte Carlo analyses performed. The last two rows of this table illustrate the operation flexibility provided to the user by the selection of the control currents. For the same circuit presented here CC1-choice of $I_1 = 10\mu$ A, $I_2 = 20\mu$ A, and $I_H = 2\mu$ A yields low power operation, whereas CC2-choice of $I_1 = 100 \mu A$, $I_2 = 200 \mu A$, and $I_H = 20 \mu A$ provides faster operation. The advantages of choices CC1 and CC2 over the circuit presented in [16] are the wider range for the control current values and much lower MC-RTE values as exhibited in the last column of Table IV; another advantage over the circuits given in [1] and [6] which only do classification is the functional flexibility in performing different operations, such as character recognition, quantization, of the proposed circuit because of its topology.

5. CONCLUSION

In this paper, a new kind of current-mode classifier circuit has been proposed and its use illustrated in character recognition, template matching and quantization applications. The circuit is based on subcircuits performing the threshold and subtraction operations. Although the templates used in applications involved only binary values, it has been shown that the classifier circuit can be utilized with analog values as well. The proposed custom tunable classifier circuit can also be useful for other classification and matching applications. The CC's power management, operation speed and tracking performances, have been favorably compared with previously proposed realizations; its use and correct operation as a basic building block in hard-classifiers designed for character recognition, template matching, fault diagnosis applications have been confirmed with SPICE simulations. Monte Carlo analyses performed for several sets of control

| Reference | Technology (µm) | Supply voltage (V) Power | time | Response parameters | MC-RTE | Control |
|---------------|--------------------|-----------------------------|-----------------------------|------------------------|---------|--|
| [1] | 0.6 | 3.3 | 14.95 mW | _ | _ | _ |
| [6] | 0.5 | 3.3 | 90 <i>μ</i> -160 <i>μ</i> W | 20 μ–40 μs | _ | _ |
| [16] | 0.35 | ±1.65 | 0.58 mW | 6 ns | % ±2.1 | $l_1 = 100 \ \mu A$, $l_2 = 200 \ \mu A$, $l_H = 20 \ \mu A$ |
| This work CC1 | 0.35 | ±1.65 | 66 µW | 26 ns | %±1.1 | $l_1 = 10 \ \mu A$, $l_2 = 20 \ \mu A$, $l_H = 2 \ \mu A$ |
| This work CC2 | 0.35 | ±1.65 | 0.66 mW | 5 ns | % ± 1.1 | $l_1 = 100 \ \mu A$, $l_2 = 200 \ \mu A$, $l_H = 20 \ \mu A$ |

Table IV. Comparison table of core cells.

parameter values and 100 runs for each set show that parameter mismatch is of little concern for classification and/or quantization applications as the deviations in magnitude of the outcome values are much lower than the higher error margins. As a byproduct, an undesirable effect in classifier circuits, the hysteresis phenomenon observed in the hardware realization of the previously introduced threshold circuit [16] has been corrected by modifying the circuit at the expense of faster operation. Further applications (i) to binary tree search algorithms, diagnosis problems, etc., and learning algorithms for determining control parameters, (ii) to grey level images as the CC is of analog nature, and (iii) exploiting more complex behaviors when fixed control parameter values are allowed to vary, will follow next.

ACKNOWLEDGEMENTS

This work is part of project 106E139 supported by the Scientific & Technological Research Council of Turkey (TÜBİTAK). We are also grateful to anonymous reviewer for his efforts and comments to improve the paper.

REFERENCES

- Liu B, Chen C, Tsao J. A modular current-mode classifier circuit for template matching application. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 2000; 47(2):145–151.
- Landolt O, Vittoz E, Heim P. CMOS selfbiased Euclidean distance computing circuit with high dynamic range. *Electronics Letters* 1992; 28(4):352–354.
- 3. Hunt E. Artificial Intelligence. Academic Press: New York, 1975.
- Abdel-Aty-Zohdy HS, Al-Nsour M. Reinforcement learning neural network circuits for electronic nose. *IEEE International Symposium on Circuits and Systems*, Orlando, FL, vol. 5, 30 May–2 June 1999; 379–382.
- Yamasaki T, Yamamoto K, Shibata T. Analog pattern classifier with flexible matching circuitry based on principal-axis-projection vector representation. *Proceedings of the 27th European Solid-State Circuits*

Conference, Villach, Austria, 18–20 September 2001; 197–200.

- Yu Peng S, Hasler PE, Anderson DV. An analog programmable multidimensional radial basis function based classifier. *IEEE Transactions on Circuits and Systems I: Analog and Digital Signal Processing* 2007; 54(10):2148–2158.
- Lin G, Shi B. A current-mode sorting circuit for pattern recognition. *Intelligent Processing and Manufacturing* of Materials, Honolulu, Hawaii, 10–15 July 1999; 1003–1007.
- Yamasaki T, Shibata T. Analog soft-patternmatching classifier using floating-gate MOS technology. *IEEE Transactions on Neural Networks* 2003; 14(5): 1257–1265.
- 9. Duda RO, Hart PE. Pattern Classification and Scene Analysis. Wiley: New York, 1973.
- Gharavi-Alkhansari M. A fast globally optimal algorithm for template matching using low-resolution pruning. *IEEE Transactions on Image Process* 2001; 10(4): 526–533.
- Schweitzer H, Bell JW, Wu F. Very fast template matching. Proceedings of the Seventh Europe Conference on Computer Vision IV, 2002; 358–372.
- Chou C-H, Chen Y-C. A VLSI architecture for realtime and flexible image template matching. *IEEE Transactions on Circuits and Systems* 1989; 36(10): 1336–1342.
- Stefano LD, Mattoccia S. Fast template matching using bounded partial correlation. *Machine Vision and Application* 2003; **13**(4):213–221.
- Morgül A, Temel T. Current-mode level restoration: circuit for multi-valued logic. *Electronics Letters* 2005; 41(5):230–231.
- Freitasa DA, Current KW. CMOS current comparator circuit. *Electronics Letters* 1983; 19(17):695–697.
- Yıldız M, Minaei S, Göknar IC. A CMOS classifier circuit using neural networks with novel architecture. *IEEE Transaction on Neural Networks* 2007; 18(6): 1845–1849.

A Low-Power Multilevel-Output Classifier Circuit

Merih Yıldız Department of Electronics and Communications Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul <u>myildiz@dogus.edu.tr</u> Shahram Minaei Department of Electronics and Communications Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul. <u>sminaei@dogus.edu.tr</u>

Abstract— A low power CMOS implementation of a multiinput data classifier with several output levels is presented. The proposed circuit operates in current-mode and can classify several types of analog vector data. An architecture is developed comprising a threshold circuit which operates in the subthreshold region. Using $0.35\mu m$ TSMC technology parameters, SPICE simulation results for a classifier with two inputs are included to verify the anticipated results.

I. INTRODUCTION

The aim of classification is to assign an unknown object to a class containing similar objects. Classifier circuits find use in various fields of applied science such as automatic target recognition, realtime object recognition, pattern recognition, artificial intelligence, neural networks and statistics [1-5].

In this paper a one-dimensional classifier, called Core Circuit (CC) shown in Fig. 1a, with transfer characteristic as in Fig. 1b is proposed. The horizontal position, the width and the height of the transfer characteristic can be adjusted independently by means of external currents I₁, I₂ and I_{H1}. The current I_{in} is the 1-dimensional data for each core circuit and Iout is the output of the classifier. By interconnecting several core circuits and adding the output currents, a multi-output classifier (n-dimensional) can be obtained as shown in Fig. 2. Recently, a classifier circuit has been reported in [6] which consumes relatively high power Also, numerous efforts in balancing the trade-off between power consumption, area and performance have resulted in an acceptable performance. On the other hand, the rapid increasing use of battery operated portable equipment in application areas such as medical electronics and telecommunications increases the importance of low-power and small-sized VLSI (very large scale integrated) circuits' technologies. One solution to achieve low power and acceptable performance is to operate the transistors in the sub-threshold region [7]. Sub-threshold circuits are suitable only for specific applications which do not need high performance but require low power consumption. Neurons with double threshold activation function have been investigated in literature and used to classify data separable by two parallel hyperplanes [8].

The primary aim of this paper is to develop a low-power classifier circuit with n inputs and externally tunable decision regions with different output amplitude for each region. With this purpose in mind, an analog CMOS current-mode realization of a one-dimensional "classifier" circuit (core neuron, core circuit) is proposed. Due to the sub-threshold operation of the transistors in the proposed CC very low power consumption becomes possible. Also, grouping several core circuits in which each subgroup has identical input currents (different from the others), a multi-dimensional, multi-level-output (each level corresponding to the coding of a data class) classifier can be obtained.

İzzet Cem Göknar Department of Electronics and Communications Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul. cgoknar@dogus.edu.tr



Figure 1.a. Core circuit block diagram. Figure 1.b. Transfer characteristic of the core circuit.



Figure 2. Block diagram of a single-output *n*-dimensional classifier.

II. PROPOSED CMOS CORE CIRCUIT

The input-output transfer characteristic shown in Fig. 1.b can be expressed as:

$$I_{out} = \begin{cases} I_{H} & \text{if } I_{1} < I_{in} < I_{2} \\ 0 & \text{otherwise} \end{cases}$$
(1)

where the current I_H determines the level of the output current. The currents I_1 and I_2 are used to shift the horizontal position and adjust the width of the output current.

The block diagram of the CC with transfer characteristic of Fig. 1.b is shown in Fig. 3. It consists of two threshold circuits and a current mirror which acts as a subtractor.

The input-output characteristic of the threshold circuit and its CMOS realization are shown in Figs. 4 and 5, respectively. The current sources I_{in} , I_1 and I_H in Fig. 5 are applied to the circuit using simple current mirrors; all the transistors M_1 through M_6 are identical.



Figure 3. Block diagram of the core circuit



Figure 4. Input-output characteristic of the threshold circuit.



The current I_1 is the threshold value of the circuit; if the input current I_{in} exceeds I_1 the current I_H flows through the output of the circuit. The drain voltage of the transistor M_3 is approximately V_{DD} when the input current I_{in} is smaller than the threshold current I_1 . Moreover, when the current I_{in} exceeds the threshold current I_1 , the drain voltage of the transistor M_3 becomes approximately V_{SS} and the current flowing through M_4 is controlled with the drain voltage of transistor M_3 . As the transistors M_4 and M_5 are working as a switch (when M_4 is on M_5 is off and vice versa), the current I_H flows through the transistors M_4 or M_5 according to the input current I_{in} . So the Input-Output Characteristic of the circuit can be expressed as:



It is important to observe that all the transistors M1-M6 are working in the sub-threshold region.

To realize a CC two threshold circuits and a subtractor circuit are configured as shown in Fig. 6. Thus the classification region is obtained by differencing the output currents of the threshold circuits and to obtain a transfer characteristic as shown in Fig. 1b. The current I_H of the threshold circuits should be chosen equal while the following constraint must be satisfied

$$I_2 > I_1 \tag{3}$$

In Fig. 6 the transistors M_1 - M_6 and M_9 - M_{14} constitute two threshold circuits respectively. The basic current mirror constructed with the transistors M_7 and M_8 performs the desired operation of subtraction. The transistors M_{15} , M_{16} and M_{17} are used to provide the same current I_H (adjusting the output level) to both of the threshold circuits. Similarly, the same approach is also used with transistors M_{18} , M_{19} and M_{20} to provide the input current I_{in} to both threshold circuits.

III. SIMULATIONS OF THE CORE AND CLASSIFIER CIRCUITS

Threshold Circuit- The proposed threshold circuit and therefore the core circuit, used in the simulation examples that follow, are simulated with 0.35 μ m TSMC MOSIS CMOS technology parameters using SPICE. The supply voltages V_{DD} and V_{SS} used in the proposed circuit are ± 1 V respectively and, the dimensions selected for the transistors are given in Table I.



Figure 6. The schematics of the proposed core circuit.

| DIMENSIONS OF THE MOS TRANSISTORS | | | | | | | | |
|-----------------------------------|--|---|--|--|--|--|--|--|
| MOSFET | $\begin{array}{c} \mathbf{M}_{1}, \mathbf{M}_{2}, \mathbf{M}_{3}, \mathbf{M}_{4}, \mathbf{M}_{5,} \\ \mathbf{M}_{6}, \mathbf{M}_{9}, \mathbf{M}_{10}, \mathbf{M}_{11}, \\ \mathbf{M}_{12}, \mathbf{M}_{13}, \mathbf{M}_{14} \end{array}$ | M7, M8, M15, M16, M17, M18, M19, M20 | | | | | | |
| W [µm] | 10 | 20 | | | | | | |
| L [µm] | 1 | 1 | | | | | | |

TABLE I. Dimensions of the MOS transistors

Simulation result of the threshold circuit shown in Fig. 5 is given in Fig. 7 for I_1 =100nA and I_H =20nA.



Figure 7. The Input-Output characteristic of the threshold circuit.

Core Circuit- Next, the SPICE simulation result of the CC of Fig.6, with the same transistor technology parameteres, is shown in Fig. 8 for I_1 =100nA, I_2 =200nA and I_H =20nA. The power consumption of the circuit is found to be 1.63 μ W which is considerably lower than the power consumption obtained in [8] (of the order of 500-800 μ W depending on the control currents).



Figure 8. The Input-Output characteristic of the core circuit.

1-D Classifier- To classify different types of data the outputs of several proposed CCs' can be connected "in parallel" as shown in Fig. 2. In order to achieve the 1-D classifier input-output characteristic as shown in Fig. 9 the following constraints must be satisfied:

$$I_{in1} = I_{in2} = ... = I_{inn} = I_{in}$$
 (4)

$$I_1 < I_2 < I_3 < \dots < I_{(2n-1)} < I_{2n}$$
(5)



Figure 9. Desired Input-Output characteristic for the classifier.

Four core circuits are connected in parallel to construct a 1-D classifier circuit which can classify five different types of data (counting zero also as an output level). SPICE simulation result for a 1-D classifier circuit is shown in Fig. 10. The control currents for the 1-D classifier circuit are given in Table II. The total power dissipation of the circuit is found to be 4.24μ W.

TABLE II. CC PARAMETERS FOR 1-D CLASSIFIER (currents in NA)



Figure 10. The Input-Output characteristic of the 1-D classifier circuit.

2-D Classifier- The block diagram for realizing a 2-D multilevel classifier with two different data types (two input currents I_{in1} , I_{in2}) and with the same transistor technology parameteres as for the threshold circuit, is shown in Fig. 11.

 TABLE III

 CC PARAMETERS LISED IN 2-D CL ASSIFIFR (currents in NA)

| | CCTARAMETERS USED IN 2 D CERTISON TER(CUTORIS II INT) | | | | | | | | |
|---------------------------------------|---|------------------------|----------------------|-----------------------|-----------------------|-----------------------|----------------------|--|--|
| Co | ore Circ | uit-I | Power Dissipation | Co | re Circu | Power Dissipation | | | |
| I ₁ | I ₂ | I _{H1} | 1 66. W | I ₃ | I ₄ | I _{H2} | 2.56. W | | |
| 70 | 140 | 60 | 1.00µ w | 210 | 280 | 120 | 2.30µ w | | |
| Core Circuit-III Power Dissipation | | | | Cor | re Circu | it-IV | Power Dissipation | | |
| I ₅ 60 | I ₆ 140 | I _{H3} 100 | 1.95µW | I ₇ 280 | I ₈ 360 | I _{H4} 80 | 1.95µW | | |

The control currents and the power dissipation for the 2-D multilevel classifier block diagram of Fig. 11, again considerably lower than the design presented in [8], are given in Table III. The same input current I_{in1} is applied to CC-1 and CC-2, and the same current I_{in2} is applied to CC-3 and CC-4.

SPICE simulation results for two different inputs to the classifier circuit are shown in Fig. 12 from which it can be observed that there are 8 different classification regions. So this

kind of architecture configuration can classify two-dimensional data into 8 different types of classes (9, if counting the I_{in1} - I_{in2} plane) each type being encoded with a different output current value. The number of classes can be increased as desired by connecting more CCs in parallel for each type of data.

IV. CONCLUSION

In this paper, a current mode low-power CMOS core circuit has been introduced and using this core circuit, a very flexible neural network like architecture has been presented. This circuit possessing a transfer characteristic with independently tunable heights, width and horizontal position, the proposed architecture can be used for a variety of purposes such as classification, quantization etc.

An n-input m-output classifier with p_i decision regions for the i_{th} type of data can be obtained by creating n group consisting of p_i CCs each and adding the outputs of CCs properly to form the desired m outputs.

Because of the parallel processing characteristic of the circuit, it is well-suited for real-world applications.



Figure 11. Block diagram of a two-dimensional multilevel classifier circuit.



Figure 12. (I_{in1}-I_{in2})-I_{out} characteristic of the proposed two-dimensional multilevel classifier.

REFERENCES

- B. Liu, C. Chen, and J. Tsao, "A Modular Current-Mode Classifier Circuit for Template Matching Application", IEEE Trans. on Circuit and Systems-II, Analog and Digital Sig. Process. vol. 47, No. 2, pp. 145-151, 2000.
- [2] E. Hunt, Artificial Intelligence. New York : Academic, 1975.
- [3] H.S. Abdel-Aty-Zohdy, M. Al-Nsour, "Reinforcement learning neural network circuits for electronic nose" IEEE International Symposium on Circuits and Systems, Orlando, FL, vol. 5, 30 May-2 June, 1999, pp. 379 – 382.
- [4] G. Lin and B. Shi, "A current-mode sorting circuit for pattern recognition" Intelligent Processing and Manufacturing of Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 1003 – 1007.
- [5] G. Lin and B. Shi, "A multi-input current-mode fuzzy integrated circuit for pattern Recognition" Second International Conference on Intelligent

Processing and Manufacturing Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 687-693.

- [6] M. Yıldız, S. Minaei, C. Göknar, "CMOS Realization of a Quantized-Output Classifier Circuit", 13th IEEE International Conference on Electronics, Circuits and Systems, Nice, France, Dec 10 - 13, 2006, pp. 292-295.
- [7] J. Chen, L. T. Clark, T.-H. Chen, "An Ultra-Low-Power Memory With a Subthreshold Power Supply Voltage", IEEE Journel of Solid-State Circuits, Vol. 41, No. 10, October, 2006, pp. 2344-2353.
- [8] M. Yıldız, S. Minaei, C. Göknar, "Current Mode Double Threshold Neuron Activation Function", *Complex Computing-Networks: Brain-like* and Wave-oriented Electrodynamic Algorithms. Springer Proceedings in Physics, Netherlands: Springer, 2006, pp. 267-274.
- [9] A. Morgül and T. Temel, "Curent-mode level restoration : circuit for multi-valued logic", Electronics Letters, Vol.41, No. 5, March, 2005, pp. 230-231.

CMOS Realization of a Quantized-Output Classifier Circuit

Merih Yıldız Department of Electronics and Communication Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul <u>myildiz@dogus.edu.tr</u> Shahram Minaei Department of Electronics and Communication Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul. <u>sminaei@dogus.edu.tr</u>

Abstract— In this paper a CMOS implementation of a multi-input data classifier with several output levels and a different architecture is presented. The proposed circuit operates in current-mode and can classify several types of analog vector data. The classifier circuit's new architecture consists of the interconnections of core cells each possessing a rectangular I_{out}/I_{in} characteristic with custom adjustable 'Size and Position (SaP)'; proper selection of (SaP) changes the classifier into a quantizer. Using 0.35µm TSMC technology parameters, SPICE simulation results for a quantizer and a classifier with two inputs are included to verify the expected results.

I. INTRODUCTION

The aim of classification is to assign an unknown object to a class containing similar objects. Classifier circuits can find applications in various fields of applied science such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks and statistics [1-5].

In this paper a one-dimensional classifier, called core circuit, shown in Fig. 1a with transfer characteristic as in Fig. 1b is used to realize an n-dimensional classifier as shown in Fig. 2 with an architecture resembling that of a 1layer neural network. The current Iin is the 1-dimensional data for each core circuit and Iout is the output of the classifier; it should be observed that by grouping core circuits and adding the outputs in each group a multi-output classifier can be obtained as well. The transfer characteristic of the block in Fig. 1b is similar to a Trapezoidal Activation Function (TAF) [6] which is used in PWL approximation, high-speed folding analog-to-digital converters, fuzzy controllers, etc. The stability properties of neural networks with TAF have been investigated in [7] and a switched capacitor CMOS implementation of a cell with TAF has been proposed in [8]. Neurons with double threshold activation function have been investigated in literature and used to classify data separable by two parallel hyperplanes [9]. The primary aim of this paper is to develop a classifier circuit with n inputs and externally tunable decision regions. In order to achieve this, an analog CMOS current-mode realization of a one-dimensional "classifier" circuit (core neuron) is realized. One-dimensional classifier can be used as a quantizer circuit which in turn can be used for analog digital conversion.

İzzet Cem Göknar Department of Electronics and Communication Engineering, Dogus University, Acibadem, Kadikoy 34722, Istanbul. cgoknar@dogus.edu.tr



Figure 1.a. Core circuit block diagram. Figure 1.b. Transfer characteristic of the core circuit.



Figure 2. Block diagram of an n-dimensional classifier.

The classifier circuit has been simulated: *i*) with one input to illustrate the effects of quantization and *ii*) with two inputs to show the result of two-dimensional multilevel classification. The core neuron of the proposed circuit has a transfer characteristic as shown in Fig. 1.b. The 'Size and (horizontal) Position (SaP)', the width and the height of the transfer characteristic can be adjusted independently by means of external currents I_1 , I_2 and I_{H1} and I_{H2} . Using several core circuits as shown in Fig. 2, a multi-dimensional, multi-level-output (each level corresponding to the coding of a data class) classifier is obtained.

This work is part of project 106E139 supported by the Scientific & Technological Research Council of Turkey (TÜBİTAK).





II. CMOS REALIZATION OF THE CORE CIRCUIT

The input-output transfer characteristic shown in Fig. 1.b can be expressed as:

$$I_{out} = \begin{cases} I_{H1} & \text{for } I_1 < I_{in} < I_2 \\ -I_{H2} & \text{otherwise} \end{cases}$$
(1)

The currents I_{H1} and I_{H2} as shown in Fig. 1.b is the height of the output current. The currents I_1 and I_2 are used to shift the horizontal position and adjust the width of the output current.

The proposed block diagram for realizing the core circuit with transfer characteristic of Fig. 1.b is shown in Fig. 3. It consists of a current-to-voltage input stage with two different outputs V_1 , V_2 , an inverter, a NOR gate and a voltage-to-current output stage. The input stage and the inverter of the core circuit are realized as shown in Fig. 4.



Figure 4. The input stage of the core circuit with the inverter.

The diode-connected transistor M_1 and the bias current I_{BIAS} form a current-to-voltage converter. The transistors M_2 and M_3 are biased with different currents $I_{BIAS}+I_1$ and $I_{BIAS}+I_2$ to form two different threshold currents of the transfer characteristic. The transistors M_1 , M_2 and M_3 are matched. The transistors M_4 and M_5 constitute an inverter which is used to obtain an input-output characteristic with negative jump at threshold current I_2 .

The NOR gate and the output stage of the proposed core circuit are shown in Fig. 5. The desired nonzero part of the

transfer characteristics is obtained by the NOR gate (transistors M_6 , M_7 , M_8 and M_9) as follows: if both inputs of the NOR gate are at low level then the output of the gate is at high level, otherwise at low level. In fact V_{out} can be written as:

$$V_{out} = \begin{cases} V_{DD} & \text{for} & I_1 < I_{in} < I_2 \\ V_{SS} & \text{otherwise} \end{cases}$$
(2)



Figure 5. The NOR gate and the output stage of the core circuit.

SaP parameters: the location of the nonzero portion of the output is fixed with I_1 and I_2 currents. This shift of the intercept points by adjusting the starting and ending boundary points of the specified decision region makes the device custom tunable. The currents I_{H1} and I_{H2} determine the height values of the function. The current output stage of the circuit is obtained by connecting two complementary source-coupled pairs [10].

From Fig. 5 the current relations are: $I_{DM10}+I_0^-=I_{DM11}$, $I_{DM12}+I_0^+=I_{DM13}$, $I_{H1}=I_{DM10}+I_{DM12}$ and $I_{H2}=I_{DM11}+I_{DM13}$.

The output currents I_o^+ and I_o^- can be given in terms of V_{out} as follows:

$$I_{o}^{+} = \begin{cases} I_{H1} & \text{for } V_{out} = V_{DD} \\ -I_{H2} & \text{for } V_{out} = V_{SS} \end{cases}$$
(3)

$$I_{o}^{-} = \begin{cases} -I_{H2} & \text{for } V_{out} = V_{DD} \\ I_{H1} & \text{for } V_{out} = V_{SS} \end{cases}$$
(4)

It should be noted that in the next section the current I_o^+ is used as the output current I_{out} , the current I_{H2} is chosen as 0 and the current I_{H1} denoted by I_H is used to change the height of the transfer characteristic.

III. SIMULATION OF THE MULTILEVEL CLASSIFIER CIRCUIT

To classify different types of data the proposed core circuits' outputs can be connected "in parallel" as shown in Fig. 2. In order to achieve the 1-D input-output characteristic shown in Fig. 6 the following constraints must be satisfied:

$$I_{in-1} = I_{in-2} = \dots = I_{in-n} = I_{in}$$
(5)



Figure 6. Desired Input-Output characteristic for the classifier.

Also the block diagram for realizing a 2-D multilevel classifier with two different input currents (I_{in1}, I_{in2}) is shown in Fig. 7.

The proposed core circuit is simulated using SPICE with 0.35 μ m TSMC MOSIS CMOS technology parameters. The voltage supply used in the proposed circuit is \pm 1.25 V, the current I_{BIAS} is chosen as 0.1 μ A. The dimensions of the transistors are given in Table I.

The control currents and the power dissipation for the 2-D multilevel classifier block diagram of Fig. 7 are shown in Table II. The current I_{in1} is applied to core circuit-1 and core circuit-2, and the current I_{in2} is applied to core circuit-3 and core circuit-4.

SPICE simulation results for two different inputs to the classifier circuit are shown in Fig. 8. It can be seen from Fig. 8 that there are 8 different regions. So this kind of configuration can classify two-dimensional data into 8 different types of classes (9 if counting the I_{in1} - I_{in2} plane) each type being encoded with a different output current value.

To construct a quantizer circuit with eight quanta, eight core circuits need to be connected in parallel with the same input currents (I_{in}) and eight different sets of control

currents. The selected classification parameters I_1 , I_2 and the resolution parameter I_H for each core circuit are given in Table III.

To simulate the circuit as a quantizer single triangle input waveform is applied to the input of the circuit and the input-output characteristic versus time is shown in Fig. 9. It should be observed that quanta width and peak value can be tuned at will thus providing further design flexibility.

TABLE I. Dimensions of the MOS transistors

| MOSFET | M ₁ , M ₂ , M ₃ , M ₄ , M ₆ , M ₇ , M ₈ , M ₉ ,M ₁₀ , M ₁₁ | M_5 | M ₁₂ , M ₁₃ |
|--------|---|-------|-----------------------------------|
| W [µm] | 10 | 5 | 16 |
| L [µm] | 0.7 | 0.7 | 0.7 |

TABLE II. Core Circuit Parameters (currents in µA)

| Core Circuit-I | | | Power Dissipation | Core Circuit-II | | | Power Dissipation |
|------------------|----------------|-----------------|----------------------|-----------------|----------------|----------------------|----------------------|
| I ₁ | I ₂ | I _{H1} | 0.25 11 | I ₃ | I_4 | I _{H2} | 0.77 11 |
| 70 | 140 | 60 | 0.35mW | 210 | 280 | 120 | 0.//mw |
| Core Circuit-III | | | Power Dissipation | Core | e Circui | Power Dissipation | |
| I ₅ | I ₆ | I _{H3} | 0.20 HI | I ₇ | I ₈ | I _{H4} | 0.01 11/ |
| 60 | 140 | 100 | 0.39mW | 280 | 360 | 80 | 0.91mW |

 TABLE III

 Core Circuit (CC) Parameters Used in Quantizer Circuit

| | CC1 | CC2 | CC3 | CC4 | CC5 | CC6 | CC7 | CC8 |
|---------------------------|-----|-----|-----|-----|-----|-----|-----|-----|
| $I_{1}\left(\mu A\right)$ | 0 | 5 | 10 | 15 | 20 | 25 | 30 | 35 |
| $I_{2}\left(\mu A\right)$ | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 |
| $I_{\rm H}(\mu A)$ | 0.1 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |

IV. CONCLUSION

In this paper, a current mode CMOS-only core circuit has been introduced and using this circuit, a very flexible neural network like architecture has been designed and put to use. The core circuit possessing a rectangular shaped transfer characteristic with independently tunable positive and negative heights, width and horizontal position, the proposed architecture can be used as illustrated for many purposes such as classification, quantization etc. Another advantage of the proposed circuit is its ability to operate at low supply voltages and its low power consumption. Its ability to parallel process data makes the circuit well-suited for real-world applications.



Figure 7. Block diagram of a two-dimensional multilevel classifier circuit.



Figure 8. (Iin1-Iin2)-Iout characteristic of the proposed two-dimensional multilevel classifier.



Figure 9. Iin-Iout characteristic of the eight level quantizer circuit.

REFERENCES

- B. Liu, C. Chen, and J. Tsao, "A Modular Current-Mode Classifier Circuit for Template Matching Application", IEEE Trans. On Circuit and Systems-II, Analog and Digital Sig. Process. vol. 47, No. 2, pp. 145-151, 2000.
- [2] E. Hunt, Artificial Intelligence. New York : Academic, 1975.
- [3] H.S. Abdel-Aty-Zohdy, M. Al-Nsour, "Reinforcement learning neural network circuits for electronic nose" IEEE International Symposium on Circuits and Systems, Orlando, FL, vol. 5, 30 May-2 June, 1999, pp. 379 – 382.
- [4] G. Lin and B. Shi, "A current-mode sorting circuit for pattern recognition" Intelligent Processing and Manufacturing of Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 1003 – 1007.
- [5] G. Lin and B. Shi, "A multi-input current-mode fuzzy integrated circuit for pattern Recognition" Second International Conference on Intelligent Processing and Manufacturing Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 687-693.

- [6] M. Kachare, J. Ramirez-Angulo, R. Carvajel, and A. J. Lopez-Martin, "New Low-Voltage Fully Programmable CMOS Triangular/Trapezoidal Fuention Generator Circuits", IEEE Trans. On Circuit and Systems-I, Vol. 52, No. 10, pp. 2033-2040, Oct. 2005.
- [7] E. Bilgili, İ.C. Göknar and O.N. Uçan, "Cellular Neural Networks with Trapezoidal Activation Function," Int. Journal of Circuit Theory and Applications, Vol. 33, No. 5, pp. 393-417,2005.
- [8] D.Y. Aksin, S. Aras, İ.C. Göknar, "CMOS Realization of User Programmable, Single-Level, Double-Threshold Generalized Perceptron," Proceedings of Turkish Artificial Intelligence and Neural Networks Conference, İzmir, Turkey, June 21-23, 2000.
- [9] M. Yıldız, S. Minaei, C. Göknar, "Current Mode Double Threshold Neuron Activation Function", Complex Computing-Networks: Brainlike and Wave-oriented Electrodynamic Algorithms. Springer Proceedings in Physics, Netherlands: Springer, 2006, pp. 267-274
- [10] A. F. Arbel and L. Goldminz, "Output Stage for Current-Mode Feedback Amplifier, Theory and Applications", Analog Integrated Circuits and Signal Processing vol. 2 issue. 3, pp. 243-255, 1992.

Düşük güçlü çok seviyeli CMOS sınıflandırıcı devresi

Merih YILDIZ^{*}, Serdar ÖZOĞUZ^{**}, Shahram MINAEI^{*}, Cem GÖKNAR^{*}

*Doğuş Üniversitesi, Elektronik ve Haberleşme Mühendisliği,Acıbadem, İstanbul **İTÜ Elektronik ve Haberleşme Mühendisliği Bölümü, Ayazağa, İstanbul

Özet

İnsanların günlük yaşamında belirli bir sesi, görüntüyü veva analog bir veriyi tanımak için kullandıkları kuralları tanımlamak oldukça karmaşık bir dizi işlem gerektirmektedir ve hatta bu kuralları tanımlamak bazen mümkün olamamaktadır. Oysa pratikte karşılaşılan örüntü tanıma olaylarını, yazılım ve donanım tabanlı tanıma uygulamalarında belirli kriterlere oturtmak mümkündür. Sınıflandırma yöntemleri ilk olarak örüntü sınıflandırma adı altında görülmeye baslanmış ve ilk algoritmalarda basit yapılar ele alınmıştır; ilk gerçeklenen yapıda en yakın komşu yakınsaması kullanılmıştır. Sınıflandırma işlemi, benzer özellik taşıyan objelerin başka farklı özellikte olanlardan avırt edilmesi seklinde tanımlanabilir ve otomatik hedef belirleme, yapay zekâ, vapav sinir ağları, analog-savısal dönüstürücüler, kuantalama, tıbbi tanı, istatistik gibi cesitli alanlarda kullanılır. Dolaysıyla da, günümüzde, gerek gerçek dünyada gerekse sayısal dünyada verilerin sınıflandırılması büyük önem taşımaktadır. Bugüne kadar sınıflandırma işlemi genellikle çeşitli algoritmalar yardımıyla yazılımsal olarak yapılmaktadır. Oysa birçok uygulamada, sınıflandırma işlemini daha hızlı ve gerçek zamanda yapmak gerektiğinden bu algoritmaların donanımsal olarak gerçeklenmesi çok daha yararlı olacaktır. Ayrıca günümüzde portatif cihazlarında artmasından dolayı donanımsal olarak gerçeklenecek cihazlarda da güç tüketimi büvük önem kazanmıştır. Dolayısıyla sınıflandırıcı devrelerinde bu ihtiyaçları karşılayacak şekilde tasarlanması gerekmektedir. Bu makalede akım-modlu düşük güçte çalışan sınıflandırıcı devresi tasarlanmıştır. Önerilen devrenin benzetimleri için 0.35-µm TSMC teknoloji parametreleri kullanılmıstır.

Anahtar Kelimeler: Sınıflandırıcı devreler, CMOS, akım modlu, düşük güç.

Bu çalışma, 106E139 no'lu TUBİTAK projesi tarafından desteklenmektedir.

Bu makale, birinci yazar tarafından İTÜ Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Programında tamamlanmış olan " Ayarlanabilir CMOS sınıflandırıcı devrelerde yeni olanaklar" adlı doktora tezinden hazırlanmıştır.
A Low-power Multilevel CMOS classifier circuit

Extended abstract

In the everyday life of humans, to define the rules used to recognize a certain sound, image or an analog data necessitates a sequence of complex processes and sometimes it even becomes impossible. However, to develop well-defined software and hardware based criteria in the application of pattern recognition problems, is possible. The aim of classification can be defined as to assign an unknown object to a class containing similar objects (or to distinguish objects having the same properties from those not possessing). Classification is used in a huge variety of applications such as automatic target identification, artificial intelligence, artificial neural networks, analog to digital converters, quantization, medical diagnosis, statistics etc.

Therefore nowadays, be it in the real or digital world, classification of data is becoming increasingly important. But until recently, major work on classification was on developing algorithms used in software packages whereas, in many applications it is becoming more and more important to classify data much faster and in real time, entailing the need for hardware realization of these algorithms. Software approaches are not practical for real-time applications, , the processing is computationally very expensive, consuming a lot of CPU time when implemented as software running on general purpose computers [6]. Whereas the literature abounds with soft algorithms, hard classifiers are seldom encountered. The recent developments in electronic technology has created a perfect medium for the hardware realization of classifier structures which, in turn, will render many classifier application prospects feasible in real time. This paper targets the design and application to real world problems of high speed, tunable, and lowpower *new classifier circuit using* CMOS technology. In this paper a low power CMOS implementation of a multi-input data classifier with several output levels is presented. The proposed circuit operates in current-mode and can classify several types of analog vector data. An Architecture is developed comprising a threshold circuit which operates in sub-threshold region. Using 0.35-µm TSMC technology parameters, SPICE simulation results for a classifier with two inputs are included to verify the results.

Keywords: Classifier circuits, CMOS, current-mode, low-power..

Giriş

Sınıflandırma, belli özellikler taşıyan veri kümesinden aynı özellikleri taşıyanların ayırt edilme işlemi olarak tanımlanır (Hunt 1975). Sınıflandırma işlemi, uygulama bilimlere örnek olarak karakter tanıma (Lin vd., 1999a), otomatik hedef tanıma, nesne tanıma (Lin vd., 1999b), yapay zeka (Abdel-Aty-Zohdy vd., 1999)., yapay sinir ağları (Merih vd., 2006b) ve istatistik gibi konularda kullanılmaktadır (Liu vd.,2000).

Bu makalede Çekirdek Devre (ÇD) olarak isimlendirilen tek boyutlu sınıflandırıcı devresi tasarlanmıştır blok diyagramı Şekil 1a'da ve geçiş karakteristiği Şekil 1b'de gösterilmiştir. Devrede kullanılan I_1 , I_2 and I_H kontrol akımları dışarıdan değiştirilebilecek şekilde gerçeklenmiştir. Buradaki I_{in} akımı tek boyuttlu giriş verisini Iout akımı ise tek boyutlu çıkışı olusturmaktadır. Bu tek boyutlu sınıflandırıcı vardımı ile n-boyutlu sınıflandırıcı devreside Şekil 2'de gösterildiği şekilde tasarlanabilir. Ayrıca çekirdek devre yapıları kullanılarak çok çıkışlı sınıflandırıcı bloklarıda oluşturulabilir. devreleri Literatürde cesitli sınıflandırıcı önerilmistir (Merih vd., 2007b), ancak bu devrelerin güç tüketimlerinin çok olduğu bilinmektedir (Merih vd., 2006a). Diğer tarafdan sınıflandırıcı devrelerin portatif cihazlarda da kullanılacağı düşünüldüğünde, haberlesme elektroniği özellikle ve tıp uygulamalarında, güç tüketimininde önemli bir parametre olduğuda unutulmamalıdır. Güç tüketiminin iyileştirilmesinde kullanılabilecek bir yöntem gerçekleştirilecek devrelerin zayıf evirtimde çalıştırılması olabilmektedir (Merih 2007a). Devrenin zavıf evirtimde vd., çalıştırılması devrenin hız performansının bir miktar azalmasına sebep olsa bile devrenin kullanılacağı uygulamalar yüksek hıza değil daha çok düşük güç tüketimini ihtiyaç duyacaklardır.

Bu makalede hedef düşük güç tüketimi ile çalışan n-girişli karar bölgeleri dışarıdan ayarlanabilen bir sınıflandırıcı devresi tasarlamaktır. Bu amaçla akım modlu tek boyutlu sınfılandırıcı çekirdek devresi tasarlanmıştır.



Şekil 1.a. Çekirdek devre blok diyagramı.



Şekil 1.b. Çekirdek devre geçiş karakteristiği.



Şekil 2. n-boyutlu sınıflandırıcının blok diyagramı.

Önerilmiş CMOS çekirdek devre

Tasarlanmış olan Şekil 1b'de verilen devrenin giriş çıkış karakteristiği (1) ifadesi ile gösterilebilir.

$$I_{out} = \begin{cases} I_{H} & I_{1} < I_{in} < I_{2} & i c c in \\ 0 & d i c c e r \end{cases}$$
(1)

Şekil 1b'deki I_{H1} akımı çıkış akımının seviyesini belirlemektedir. I_1 , I_2 akımları ise karakteristiğin yatay eksen üzerindeki yerini ayarlamak için kullanılır.



Şekil 3. Çekirdek yapının blok diyagramı

Çekirdek devrenin blok diyagramı Şekil 3'de verilmiştir. Bu blok diyagram için iki adet eşik devresi (Morgül vd., 2005) ve bir fark alma devresi bulunmaktadır. Eşik devresinin giriş çıkış karakteristiği Şekil 4'de verilmiştir. Böyle bir giriş çıkış karakteristiği (2) bağıtısı ile ifade edilebilir.

$$I_{out} = \begin{cases} I_{H} & I_{in1} \ge I_{1} & \text{icin} \\ 0 & \text{diger} \end{cases}$$
(2)



Şekil 4. Eşik devresinin giriş çıkış karakteristiği.

Devrede I₁ akımı eşik değeridir. Eğer giriş akımı Iin, I1 akımından büyükse devrenin çıkışındaki akım I_H olur. Giriş akımı I_{in} eşik akımından ufak olduğu zaman da M2 tranzistorunun savak gerilimi yaklaşık olarak V_{DD}'ye eşit olur; giriş akımı eşik akımından büyük olduğu zaman ise M₂ tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur. Diğer taraftan M₄ tranzistorunun akımı M₂ tranzistorunun savak gerilimi ile de kontrol edilmektedir. M₂ tranzistorunun savak gerilimi yaklaşık V_{SS} olduğu zaman M₄ tranzistoru kesime girerek I_H akımının M₅ üzerinden sağlanır. Benzer sekilde akması M_2 tranzistorunun savak gerilimi yaklasık V_{DD} olduğu zaman M₄ tranzistoru iletime geçerek I_H akımının M₄ üzerinden akması sağlanır. M₄ ve M₅ (M₄ açıkken M₅ kapalı, M₅ açık iken M₄ kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M₄ ya da M₅ tranzistorundan akmaktadır. Çekirdek devrenin gerçekleştirlmesi için Şekil 5'de verilen yapı kullanılmıştır. Bu bölgenin olusturulabilmesi için için eşik devrelerinin I_H akımları eşit seçilmeli ve $I_2 > I_1$ sağlanmalıdır. Önerilen çekirdek devre Şekil 6'da verilmiştir. Şekil 6'da M₁-M₆ ve M₉-M₁₄ tranzistorları eşik devrelerini oluşturmaktadır. Çıkış akımlarının farkını almak için kullanılan fark devresi ise basit akım aynası (M7 ve M8) kullanılarak gerçeklenmiştir. M₁₅, M₁₆ ve M₁₇ tranzistorları eşik devrelerine aynı I_H akımını uygulamak için kullanılmıştır. Benzer şekilde M₁₈, M₁₉ ve M₂₀ tranzistorları da girişlere aynı I_{in} akımını uvgulamak icin kullanılmıştır. Önerilen devrenin simülasyonunda 0.35-µm TSMC CMOS teknoloji parametreleri kullanılmıştır. Besleme gerilimleri V_{DD} ve V_{SS} ±1 V olarak alınmıştır. Tranzistor boyutları Tablo 1'de verilmiştir.



Şekil 5. Çekirdek yapının gerçekleştirilme blok şeması.

M. Yıldız, S. Özoğuz, S. Minaei, C. Göknar

| MOSFET | $ \begin{array}{c} M_1, M_2, M_4, M_5, \\ M_6, M_{12}, M_{13}, M_{14} \end{array} $ | $ \begin{array}{ c c c c c c c c c c c c c c c c c c c$ | M ₁ , M ₂ , M ₉ , M ₁₀ |
|--------|---|---|--|
| W [µm] | 10.5 | 21 | 42 |
| L [µm] | 1.05 | 1.05 | 1.05 |

Tablo 1. MOS tranzistorların boyutları



Şekil 6. Çekirdek devre.

Çekirdek devre benzetimi

Çekirdek devrenin oluşturulmasında kullanılan temel blok olan eşik devresinin benzetim sonuçları $I_1=100$ nA ve $I_H=20$ nA için Şekil 7'de verilmiştir.



Şekil 7. Eşik devresinin giriş çıkış karakteristiği.

Çekirdek yapının benzetim sonuçları $I_1=100$ nA, $I_2=200$ nA ve $I_H=20$ nA için Şekil 8'de verilmiştir. Çekirdek devrenin güç tüketimi 1.63 μ W olarak elde edilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.



Şekil 8. Çekirdek yapının giriş çıkış karakteristiği.



Şekil 9. Tek boyutlu sınıflandırıcı yapısı

Tek boyutlu sınıflandırıcı devresi

Tek boyutlu verilerin sınıflandırılması için Şekil 9'da gösterildiği gibi çekirdek yapılar paralel olarak bağlanır ve (3), (4) bağıntıları sağlanacak şekilde kontrol akımları seçilir.

$$I_{in1} = I_{in2} = ... = I_{inn} = I_{in}$$
 (3)

$$I_1 < I_2 < I_3 < \dots < I_{(2n-1)} < I_{2n}$$
 (4)

Kontrol akımlarının uygun olarak seçildiği durumda tek boyutlu sınıflandırıcının giriş çıkış karakteristiği Şekil 10'da verildiği gibi elde edilir. Tek boyutlu devrenin benzetimi için dört adet çekirdek yapı paralel olarak bağlanmıştır. Simülasyon sonucu Şekil 11'de ve kontrol akımları Tablo 2'de verilmiştir.



Şekil 10. Tek boyutlu sınıflandırıcının giriş çıkış karakteristiği

Tablo 2. Tek boyutlu sınıflandırıcı çekirdekdevre kontrol akımları (akımlar nA)

| Çekirdek Devre-I | | | Çekirdek Devre-II | | |
|--------------------|----------------|-----------------|-------------------|----------------|-----------------|
| I ₁ | I ₂ | I _{H1} | I ₃ | I ₄ | I _{H2} |
| 20 | 30 | 10 | 40 | 50 | 20 |
| Çekirdek Devre-III | | | Çekirdek Devre-IV | | |
| I ₅ | I ₆ | I _{H3} | I ₇ | I ₈ | I _{H4} |
| 60 | 70 | 30 | 80 | 90 | 40 |

Simülasyon sonuçlarından görüldüğü gibi x-ekseni de dahil olmak üzere beş farklı tipte veri sınıflandırılmaktadır; devrenin toplam güç tüketimi 0.95µW çıkmaktadır.



Şekil 11. Tek boyutlu sınıflandırıcı giriş çıkış

İki boyutlu sınıflandırıcı devresi

İki boyutlu sınıflandırıcı devrenin simülasyonu icin Sekil 12'de verilen blok kurulmustur. seması Simülasvonda kullanılan her bir çekirdek devrenin kontrol akımları Tablo 3'de verilmiştir. Şekil 12'de gösterildiği gibi Iin1 ve Iin2 girişleri iki boyutlu giriş verilerini göstermektedir. İki boyutlu sınıflandırıcının (Iin1-Iin2)-Iout giriş çıkış karakteristiği Şekil 13'de verilmiştir. Şekil 13'den görüldüğü gibi Iout=0 bölgeside dahil edildiği zaman 9 farklı bölge, veri sınıflandırılabilmektedir.



Şekil 12. İki boyutlu sınıflandırıcının yapısı



Şekil 13. İki boyutlu sınıflandırıcının $(I_{in1}-I_{in2})-I_{out}$ giriş çıkış karakteristiği.

| 1 u 0 10 J. $1 k 1 0 0 $ y $u 1 u 1 $ singlund i lei Çeki u ek u ek e konti 0 1 u kini u 1 (u kini u 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | Tablo 3. | İki boyutl | u sınıflandırıcı | çekirdek devre | kontrol akın | nları (akımlar nz | 1) |
|--|----------|------------|------------------|----------------|--------------|-------------------|----|
|--|----------|------------|------------------|----------------|--------------|-------------------|----|

| Çeki | rdek De | vre-I | e-I Güç Tüketimi Çekirdek Devre -II | | Güç Tüketimi | | |
|--------------------|----------------|-----------------|-------------------------------------|----------------|----------------|-----------------|--------|
| I ₁ | I ₂ | I _{H1} | | I ₃ | I ₄ | I _{H2} | |
| 70 | 140 | 60 | 1.66µW | 210 | 280 | 120 | 2.56µW |
| Çekirdek Devre-III | | Güç Tüketimi | Çekirdek Devre -IV | | Güç Tüketimi | | |
| I ₅ | I ₆ | I _{H3} | | I ₇ | I ₈ | I _{H4} | |
| 60 | 140 | 100 | 1.95µW | 280 | 360 | 80 | 1.95µW |

Sınıflandırma bölgeleri

Şekil 12'de verilen blok diyagramı üç şekilde genelleştirilebilir.

(*i*) Aynı I_{in1} giriş akımı m adet çekirdek devreye ve aynı I_{in2} giriş akımı n adet çekirdek devreye paralel olarak bağlanır. Oluşan devre iki boyutlu m(n+1)+n+1 ($I_{out}=0$ düzlemini de dahil edilirse) veriyi sınıflandırabilmektedir. Bu şekilde oluşturulmuş iki boyutlu sınıflandırıcı taslağı Şekil 14'de verilmiştir. Şekil 14'de kutuların içlerine yazılmış değerler her farklı çıkış seviyesini belirlemektedir. AI ise ayırt edilebilecek en küçük değerdeki akım değeridir.

(*ii*) n adet çekirdek devreyi parallel bağlayıp, her bloğa ayrı bir giriş uygulayarak, çıkışlarıda tek bir noktada toplanarak nboyutlu bir sınıflandırıcı devre oluşturulabilir. Bu durum da ise n-boyutlu ve tek çıkışlı bir sınıflandırıcı devre oluşturulabilir.

(*iii*) (i) ve (ii) numaralı maddelerde verilen durumların kombinasyonları kullanılarak çok seviyeli ve birden fazla karar bölgesi olan sınıflandırıcı yapıları tasarlanabilir.

| | | 1 | | n |
|---|--------|----------------------------|-----|-----------------------------------|
| | | $I+ m \Delta I + \Delta I$ | ••• | $n(I+m\Delta I)+n\Delta I$ |
| 1 | Ι+ ΔΙ | 2I+ (m+2)ΔI | | $(n+1)I + [n(m+1)+1] \triangle I$ |
| | : | | | |
| m | I+ m∆I | 2I + (2m + 1)∆I | | $(n+1)I + [n(m+1) + m] \Delta I$ |
| | | | | |

Şekil 14. İki boyutlu sınıflandırıcı için genelleştirilmiş sınıflandırıcı taslağı.

Sonuçlar

Bu çalışmada düşük güç tüketimli, akım modlu sınıflandırıcı çekirdek devresi tasarlanmıştır. Bu çekirdek devre yapısı kullanılarak gerçekleştirilebilecek uygulama imkanlarından bahsedilmiştir. Önerilmiş olan çekirdek devre yapısındaki kontrol akımlarının dışarıdan ayarlanabilir olması ile sınıflandırma ve kuantalama gibi uygulamalar içinde kullanılabilir. Ayrıca bu çekirdek devrelerin paralel bağlanabiliyor olması yeni uvgulama alanlarında kullanılması icin veni Diğer tarafdan olanaklar getirmektedir. literatürde de benzer devreler gerçeklemiş olsa dahi bu devreler düşük gerilim ya da tüketimini uygun vapılar düsük güç Gerceklenmis olmamaktadır. olan devre düşük güç tüketimi ile çalışmasından dolayıda önemli bir avantaj taşımaktadır.

Kaynaklar

- Abdel-Aty-Zohdy H.S., Al-Nsour M., (1999).
 "Reinforcement learning neural network circuits for electronic nose" IEEE International Symposium on Circuits and Systems, Orlando, FL, vol. 5, 30 May-2 June, 1999, pp. 379 382.
- Hunt E.,(1975). Artificial Intelligence. New York :Academic, 1975.

- Lin G. Ve Shi B., (1999a). "A current-mode sorting circuit for pattern recognition" Intelligent Processing and Manufacturing of Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 1003 – 1007.
- Lin G. Ve Shi B., (1999b). "A multi-input currentmode fuzzy integrated circuit for pattern Recognition" Second International Conference on IntelligentProcessing and Manufacturing Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 687-693.
- Liu B., Chen C. Ve Tsao J.,(2000). "A Modular Current-Mode Classifier Circuit for Template Matching Application", IEEE Trans. on Circuit and Systems-II, Analog and Digital Sig. Process. vol. 47, No. 2, pp. 145-151, 2000.
- Morgül A. ve Temel T., (2005). "Curent-mode level restoration : circuit for multi-valued logic", Electronics Letters, Vol.41, No. 5, March, 2005, pp. 230-231.
- Yıldız M., Minaei S. ve Göknar C., (2006a). "CMOS Realization of a Quantized-Output Classifier Circuit", 13th IEEE International Conference on Electronics, Circuits and Systems, Nice, France, Dec 10 - 13, 2006, pp. 292-295.
- Yıldız M., Minaei S. ve Göknar C., (2006b).
 "Current Mode Double Threshold Neuron Activation Function", Complex Computing-Networks: Brain-like and Wave-oriented Electrodynamic Algorithms. Springer Proceedings in Physics, Netherlands: Springer, 2006, pp. 267-274.
- Yıldız M., Minaei S. ve Göknar C., (2007a). "A Low-Power Multilevel-Output Classifier Circuit", ECCTD 2007, European Conference on Circuit Theory and Design, August 26-30, 2007, Seville, Spain.
- Yıldız M., Minaei S. ve Göknar C., (2007b). "A CMOS Classifier Circuit using Neural Networks with Novel Architecture", IEEE Transaction on NN, vol. 18, vo.6, pp.1845-1849, 2007.

Linearly Weighted Classifier Circuit

Merih Yıldız^{1,2}

Shahram Minaei²

Serdar Özoğuz¹

1. Department of Electronics and Communications Engineering, Istanbul Technical University, Maslak, Istanbul, Turkey. E-mail: myildiz@dogus.edu.tr

2. Department of Electronics and Communications Engineering, Dogus University, Acibadem, Kadikoy, 34722, Istanbul, Turkey. E-mail: <u>sminaei@dogus.edu.tr</u>

ic

Abstract— In this paper a CMOS realization of a linearly weighted classifier circuit which is called classifier block is proposed. The proposed classifier block is composed of Linearly Weighted Circuits (LWC) and CMOS Core Circuits (CC). The proposed circuit can classify linearly non-separable data. The weights of the classifier circuit are achieved with LWC blocks. Using 0.35 μ m AMS technology parameters, SPICE simulation results for a LWC and classifier block are included to verify the expected results.

Keywords- Classifier, Linearly non-separable, CMOS

I. INTRODUCTION

The aim of classification is to assign an unknown object to a class containing similar objects. Classifier circuits can find applications in various fields of applied science such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks and statistics [1-4]. Although some data are linearly separable some are not. In literature some classifiers are proposed which can classify especially mesh grid partitioned data [5]. Mesh grid partitioned data is a special case of linearly non-separable data.

The neural network classifiers constructed with perceptron can classify linearly separable data [6]. But in real world, the data are not always linearly separable and scattered as mesh grid partitioned or more different, so in this paper a more general data classification is investigated. Basic type of linearly non-separable data is shown in Fig. 1.



Figure 1. Linearly non-separable data regions.

Fig. 1 shows a two-dimensional data (x_1, x_2) to be classified into different classes A and B separated with a gray filled region. In the literature mesh grid partitioned data regions which is a special case of the linearly non-separable data regions of Fig. 1 is classified with core circuits (CCs) [7].

are shown in Fig. 2a and Fig. 2.b, respectively. The currents I_1 , I_2 and I_H are the control currents of the CC. The input-output transfer characteristic of CC shown in Fig. 2.b can be expressed as:

$$I_{out} = \begin{cases} I_H & if \quad I_1 < I_{in} < I_2 \\ 0 & otherwise \end{cases}$$
(1)



Figure 2.a. CC block diagram. Figure 2.b. Transfer characteristic of the CC.

The CMOS implementation of the CC and a detailed explanation were given in reference [7].

In this paper a new classifier block is proposed to classify linearly non-separable data as shown in Fig. 1. The classification is achieved as the multiplication of inputs x_1 and x_2 with appropriate weight coefficient w_1 and w_2 and the sum of these products is the input current I_{in} of the CC as shown in Fig. 2. From Fig. 3 it can be seen that $I_{in} = w_1 x_1 + w_2 x_2$.



Figure 3. Linearly non-separable classifier block.

II. REALIZATION OF THE LINEARLY NON-SEPARABLE CLASSIFIER

The 3-D image for I_{out} - $(x_1$ - $x_2)$ characteristic of Fig. 3 is given in Fig. 4. According to the data, the boundary values formed with the coefficients w_1 , w_2 , I_1 and I_2 are calculated with Fisher's linear discriminant analysis [8-11]. In Fig. 4 I_H is chosen as to classify the type of class. The value of I_{out} which is the height of the 3-D image in Fig. 4 determines the class of data.

This work is part of project 106E139 supported by the Scientific & Technological Research Council of Turkey (TÜBİTAK).



Figure 4. Linearly non-separable data regions of Fig 3.

The block diagram of LWC is shown in Fig. 5 and the transfer characteristic of the block can be expressed as:

$$y = w x \tag{2}$$

In Fig. 5, y is the output, x is the input and w is the weight coefficient. In the circuit realization of the LWC, the input x is a voltage signal and the output y is a current signal. In fact, the LWC operates in transconductance mode. The data input can be applied from the same source and the output of the LWC can be connected in parallel as the circuit is working in transconductance mode.



Figure 5. Block diagram of LWC.

The block diagram shown in Fig. 3 can be constructed with LWC blocks as shown in Fig. 6. As the outputs of LWC blocks are current they can be summed easily by tying them and applying to the input of the CC.

More complex data regions can be achieved by parallel connection of Fig. 3 as shown in Fig. 7. In Fig. 7 four LWC blocks and two CCs are used to form the linearly non-separable data classification block so a two-dimensional, multi-level-output (output level is adjusted with I_H currents which corresponds to a different data class) classifier is obtained.

The MATLAB simulation of Fig.7 is performed with weight coefficients and CC control parameters of $w_1=2$, $w_2=0.8$, $w_3=2$, $w_4=4$, $I_{H1}=0.5$, $I_1=4$, $I_2=5.5$, $I_{H2}=1$, $I_3=0.5$ and $I_4=10$. The 3-D image of I_{out} -(x_1 - x_2) characteristic of Fig. 7 is given in Fig. 8.



Figure 6. Linearly non-separable data classifier block with LWCs.



Figure 7. Parallel connection of linearly non-separable classifier block.



Figure 8. 3-D image of I_{out} -(x₁-x₂) characteristic of Fig.7.

The heights in Fig. 8 are adjusted with the control currents I_{Hi} (*i*=1, 2). So different classes of data can be separated with these height values. From Fig. 8 it can be seen that four different classes of data can be classified (counting $I_{out}=0$ plane).

Realization of the linearly weighted circuit

The LWC can be constructed with a dual output secondgeneration current conveyors (DO-CCII). The circuit is designed so that the input data is a voltage and the output data is a current. The block diagram of the DO-CCII is given in Fig. 9.



Figure 9. Block diagram of the DO-CCII.

The DO-CCII is a four-port network defined by the following matrix equation:

$$\begin{bmatrix} V_x \\ I_y \\ I_{z+} \\ I_{z-} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_x \\ V_y \\ V_{z+} \\ V_{z-} \end{bmatrix}$$
(3)

The CMOS realization of the DO-CCII is shown in Fig 10 [12]. The terminal Y is a voltage input terminal with high impedance while the terminal X is an output voltage terminal with low impedance following the voltage at terminal Y. The output current is obtained at high output impedance terminal Z. The transistors M_7 and M_8 constitute a differential pair with active load of M_{10} - M_{11} . The transistor M_9 provides a negative feedback on the terminal X to reduce its impedance. The transistor M_1 provides the biasing current for the differential pair of M_7 and M_8 , and M_2 - M_3 and M_{12} - M_{13} are used to transfer the X terminal current to the output terminal Z_+ . The transistors M_4 - M_6 and M_{14} - M_{16} are used to obtain an inverted current of X terminal at Z- terminal. Note that the current mirrors have unity gain and all the transistors are working in saturation region.



Figure 10. The CMOS realization of DO-CCII.

The block diagram of the LWC using a DO-CCII is shown in Fig. 11. The voltage V_y is the input and the current I_{z+} and I_{z-} are the output of the circuit in Fig. 11. The output currents of the circuit in Fig. 11 can be expressed as:

$$I_{z+} = \frac{V_y}{R_1} \frac{R_2}{R_3}$$
(4)

$$I_{z-} = -\frac{V_y}{R_1} \frac{R_2}{R_3}.$$
 (5)



Figure 11. LWC configuration with DO-CCII.

The resistance R_1 is used to convert the voltage input data V_y , to current. Besides, the ratio R_2/R_3 can be used to obtain appropriate weight value. It should be mentioned that the DO-CCII is used to provide both positive and negative weight values if needed.

III. SIMULATION RESULTS OF THE LWC AND CLASSIFIER BLOCK

The proposed LWC is simulated using SPICE program with 0.35 μ m AMS CMOS technology parameters. The supply voltage used in the proposed circuit is ± 2 V. The bias voltage V_B is chosen as 0.4 V. The dimensions of the transistors used for the DO-CCII are given in Table I.

TABLE I. THE DIMENSIONS OF THE TRANSISTORS USED IN DO-CCII.

| MOSFET | M ₁ , M ₇ , M ₈ | M ₂ ,M ₃ ,M ₄ , M ₅ , M ₆ | $\begin{array}{c} M_{10}, M_{11}, M_{12}, \\ M_{13}, M_{14}, M_{15}, M_{16} \end{array}$ | M9 |
|--------|---|---|--|-----|
| W [µm] | 28 | 56 | 20 | 112 |
| L [µm] | 0.7 | 0.7 | 0.7 | 0.7 |

The input-output characteristic of the circuit in Fig. 11 for $R_1=10 \text{ k}\Omega$ and different values of $R_2/R_3=k$ is given in Fig. 12. The resistor $R_3=1 \text{ k}\Omega$ and different R_2 values of $1 \text{ k}\Omega$, $2 \text{ k}\Omega$, $2.5 \text{ k}\Omega$, $5 \text{ k}\Omega$, $10 \text{ k}\Omega$, and $20 \text{ k}\Omega$ are selected to obtain different *k* values indicated in Fig. 12.



Figure 12. I_{z+} -V_v characteristic of LWC for different values of k.

The layout of the DO-CCII in Fig. 10 based on 0.35 μ m AMS CMOS technology using MENTOR program is shown in Fig. 13. The area of the layout in Fig. 13 is approximately 460 mm². The post layout simulation for the V_x - V_y characteristics of the circuit in Fig. 13 is given in Fig. 14. In addition the post layout simulation for the I_x - I_z characteristics of the circuit in Fig. 15.



Figure 13. The layout of the DO-CCII.



Figure 14. The V_x - V_y characteristics layout of the DO-CCII.



Figure 15. The I_x - I_z characteristics of the DO-CCII.

It can be seen from the post layout simulation that the output currents $I_{z+} = 502 \ \mu\text{A}$ and $I_{z-} = -510 \ \mu\text{A}$ are obtained for the input current $I_x = 500 \ \mu\text{A}$. So the I_{Z+} and I_{Z-} currents follow the current I_x till 1 mA.

For the simulation of Fig. 7, the LWC-1 parameters $R_2=20$ k Ω , $R_3=10$ k Ω , LWC-2 parameters $R_2=10$ k Ω , $R_3=10$ k Ω , LWC-3 parameters $R_2=20$ k Ω , $R_3=10$ k Ω and LWC-4 parameters $R_2=10$ k Ω , $R_3=25$ k Ω are selected. Moreover $R_1=10$ k Ω for all of the four LWC blocks. The control current of the CCs are given in Table II. The variables x_i (i=1, 2) and w_i (i=1... 4) in Fig. 7 are formed with V_y/R_1 , and R_2/R_3 respectively, as shown in Fig.11. So the simulation result of Fig. 7 is given in Fig. 16 where x_i (i=1, 2) values are as currents and there are 3 different regions. So this kind of configuration can classify two-dimensional data into 4 different types of classes (4 if counting the x_1 - x_2 plane) each type being encoded with a different output current value.

TABLE II. THE CC CONTROL CURRENT PARAMETERS

| | CC-1 | CC-2 |
|----------------------|------|------|
| I ₁ [µA] | 40 | 80 |
| I ₂ [μA] | 60 | 100 |
| I _{H1} [µA] | 10 | 20 |



Figure 16. Simulation result of classifier block.

IV. CONCLUSION

In this paper a classifier circuit, which can classify linearly non-separable data, is proposed. The resistors used in the LWC help to adjust the weight coefficients. While the LWC operates in transconductance mode and the CC one is in current mode. Thus the output value of the classifier circuit is current and more complex data types can also be classified by parallel connection of the classifier blocks. The simulation results of the LWC and classifier block is also given to show the performance of the circuit.

REFERENCES

- B. Liu, C. Chen, and J. Tsao, "A Modular Current-Mode Classifier Circuit for Template Matching Application," *IEEE Trans. On Circuit and Systems-II, Analog and Digital Sig. Process.*, vol. 47, No. 2, pp. 145-151, 2000.
- [2] E. Hunt, Artificial Intelligence. New York : Academic, 1975.
- [3] H.S. Abdel-Aty-Zohdy, M. Al-Nsour, "Reinforcement learning neural network circuits for electronic nose" in *Proc. IEEE International Symposium on Circuits and Systems*, Orlando, FL, vol. 5, 30 May-2 June, 1999, pp. 379 – 382.
- [4] G. Lin and B. Shi, "A current-mode sorting circuit for pattern recognition" in Proc. Intelligent Processing and Manufacturing of Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 1003 – 1007.
- [5] Yıldız, M., Minaei, S., Göknar, C., "A CMOS Classifier Circuit using Neural Networks with Novel Architecture," *IEEE Transaction on Neural Networks.*, vol. 18, no. 6, pp. 1845-1849, Nov. 2007.
- [6] E. Bilgili, İ.C. Göknar and O.N. Uçan, "Cellular Neural Networks with Trapezoidal Activation Function," *Int. Journal of Circuit Theory and Applications*, Vol. 33, No. 5, pp. 393-417, 2005.
- [7] Yıldız, M., Minaei, S., Göknar, C., "Realization and Template Matching Application of a CMOS Classifier Circuit," *Applied Electronics 2008*, Pilsen, Czech Republic, 10-11 September 2008, pp. 231-234.
- [8] Qian, D., "Modified Fisher's Linear Discriminant Analysis for Hyperspectral Imagery," *IEEE, Geoscience and Remote Sensing Letters*, vol. 4, pp. 503-507, 2007
- [9] Qi, L., Donald, W. T., "Principal Feature Classification," IEEE Transaction on Neural Networks., vol. 8, no. 1, pp. 155-160, 1997
- [10] Çevikalp, H., "Theoretical Analysis of Linear Discriminant Analysis Criteria," *IEEE 14th Signal Processing and Communications Applications*, Antalya, Turkey, 17-19 April, 2006, pp. 1-4.
 [11] Jiang, G., Xiaoqing D., Youshou W., "On improvement of Multiple
- [11] Jiang, G., Xiaoqing D., Youshou W., "On improvement of Multiple Discriminant Analysis Method for Discriminative Feature Extraction," in *Proc. Systems, Man, and Cybernetics, Conference*, vol. 2, October 12-15,1999, pp. 915 – 920.
- [12] Surakampontorn, W., "Accurate CMOS-based Current Conveyors," *IEEE Transactions on Instrumentation and Measurement*, vol. 40, pp. 699-702, 1991.

<u>İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ</u>

AYARLANABİLİR CMOS SINIFLANDIRICI DEVRELERDE YENİ OLANAKLAR

DOKTORA TEZİ Merih YILDIZ

Anabilim Dalı : Elektronik ve Haberleşme Mühendisliği

Programı: Elektronik Mühendisliği

NİSAN 2009

<u>İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ</u>

AYARLANABİLİR CMOS SINIFLANDIRICI DEVRELERDE YENİ OLANAKLAR

DOKTORA TEZİ Merih YILDIZ (504022107)

Tezin Enstitüye Verildiği Tarih :13 Ocak 2009Tezin Savunulduğu Tarih :29 Nisan 2009

Tez Danışmanı :Doç. Dr. Serdar ÖZOĞUZ (İTÜ)Eş Danışman :Doç. Dr. Shahram MINAEI (DÜ)Diğer Jüri Üyeleri :Prof. Dr. Cem GÖKNAR (DÜ)Prof. Dr. Ali ZEKİ (İTÜ)Doç. Dr. Nil TARIM (İTÜ)Prof. Dr. Ece Olcay GÜNEŞ (İTÜ)Prof. Dr. Tülay YILDIRIM (YTÜ)

NİSAN 2009

ÖNSÖZ

Doktora çalışmalarım boyunca bana fikirler veren ve benden her türlü yardımı esirgemeyen sayın hocalarım Prof. Dr. Cem Göknar, Doç. Dr. Serdar Özoğuz ve Doç. Dr. Shahram Minaei'ye, serim çizimlerinde bana yardımcı olan Engin Deniz'e çok teşekkür ederim. Ayrıca tez izleme sürecindeki öneri ve fikirlerinden dolayı tez izleme komitesinde bulunan Prof. Dr. Ali Zeki ve Doç. Dr. Nil Tarım'a teşekkür ederim. Doktora tezimin başlangıç aşamasında ve devamında desteğinden dolayı da Prof. Dr. M. Sait Türköz hocamı da saygı ile anarım.

Bu çalışmalarım sırasında sabır ve desteği için sevgili eşim Başak Yıldız'a da çok teşekkür ederim.

Nisan 2009

Merih YILDIZ Y. Mühendis

iv

İÇİNDEKİLER

<u>Sayfa</u>

| ÖNSÖZ | iii |
|---|-------|
| İÇİNDEKİLER | V |
| KISALTMALAR | . vii |
| ÇİZELGE LİSTESİ | ix |
| ŞEKİL LİSTESİ | xi |
| ÖZET | . XV |
| SUMMARY | xvii |
| 1. GİRİŞ | 1 |
| 1.1 Literatür Taraması | 2 |
| 1.1.1 Yazılımsal yöntemler | 2 |
| 1.1.2 Donanımsal yapılar | 4 |
| 1.2 Ele Alınan Konu ve Problem | 9 |
| 1.3 Çalışmada Yapılanlar | . 10 |
| 2. SINIFLANDIRICI DEVRESİ TASARIMI VE BENZETİMLERİ | .13 |
| 2.1 Sınıflandırıcı Yapısı Blok Diyagramı | . 13 |
| 2.2 Çekirdek Devreler | . 16 |
| 2.2.1 Çekirdek devre-1 | . 16 |
| 2.2.2 Çekirdek devre-2 | . 25 |
| 2.2.2.1 Çekirdek devre-2'nin zayıf evirtim benzetimleri | . 35 |
| 2.2.2.2 Çekirdek devre-2'nin ayrık elemanlar ile gerçeklenmesi | . 40 |
| 2.2.2.3 Çekirdek devre-2'nin serimi | . 41 |
| 2.2.3 Çekirdek devrelerin karşılaştırılması | . 42 |
| 2.3 Çekirdek Devreler ile Elde Edilebilen Bölgeler | . 43 |
| 3. EĞİK IZGARALI SINIFLANDIRICILAR | . 45 |
| 3.1 Bölgelerin Oluşturulması | . 45 |
| 3.2 ÇAD Devresi ve Benzetimleri | . 50 |
| 4. SINIFLANDIRMA ALGORİTMALARI VE DEVRELERE | |
| UYGULANMASI | . 59 |
| 4.1 Fisher Tabanlı Algoritma ile Çift Eşik Doğrularının Bulunmalarına Genel | |
| Bakış | . 59 |
| 4.1.1 Fisher tabanlı algoritma ile çift eşik doğrularının bulunması | . 60 |
| 4.1.2 Fisher tabanlı algoritma ile çift eşik doğrularının bulunmasında | |
| genelleştirilmiş hal | . 65 |
| 4.2 Eğiticili Perseptron Öğrenme Algoritması ile ÇAD ve ÇD Parametrelerinin | |
| Bulunması | . 67 |
| 5. SINIFLANDIRICI DEVRE UYGULAMALARI | .71 |
| 5.1 Kuantalayıcı | . 71 |
| 5.2 Karakter Tanıma | . 72 |
| 5.3 Iris ve Haberman Verisinin Sınıflandırılması | . 78 |
| 5.3.1 Iris verisinin Fisher tabanlı algoritma ile sınıflandırılması | . 78 |

| 5.3.2 Haberman verisinin Fisher tabanlı algoritma ile sınıflandırılması | . 83 |
|---|------|
| 5.3.3 İris verisinin perseptron öğrenme algoritması ile sınıflandırılması | . 87 |
| 5.3.4 Haberman verisinin perseptron öğrenme algoritması ile sınıflandırılması | 90 |
| 6. SONUÇLAR VE ÖNERİLER. | .95 |
| KAYNAKLAR | .97 |
| EKLER1 | 105 |

KISALTMALAR

| ÇD | : Çekirdek Devre |
|----------------|---|
| ÇAD | : Çarpan Devresi |
| DI | : Dikdörtgen Izgara |
| YSA | : Yapay Sinir Ağları |
| KHA | : Kazanan Hepsini Alır |
| AÇ | : Akım Çoğullayıcı |
| n-D | : n Boyutlu |
| CCII | : İkinci Kuşak Akım Taşıyıcı |
| DO-CCII | : Çift Çıkışlı İkinci Kuşak Akım Taşıyıcı |
| DBSB | : Dört Boyutlu Sınıflandırıcı Bloğu |
| | |

viii

ÇİZELGE LİSTESİ

<u>Sayfa</u>

| Çizelge 2.1 : Çekirdek Yapı Değişkenleri | . 16 |
|---|------|
| Çizelge 2.2 : Çekirdek devre-1 yapısının MOS tranzistor boyutları. | . 20 |
| Cizelge 2.3 : 1×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları | . 23 |
| Cizelge 2.4 : 2×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları | . 24 |
| Çizelge 2.5 : ÇD-2'de kullanılan MOS tranzistor boyutları | . 30 |
| Çizelge 2.6 : ÇD-2 ile gerçeklenen 1×1-D kontrol akımları (µA olarak) | . 34 |
| Çizelge 2.7 : ÇD-2 ile gerçeklenen 2×1-D sınıflandırıcı devresi kontrol akımları | |
| (akımlar µA olarak ifade edilmiştir). | . 34 |
| Çizelge 2.8 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 1-D sınıflandırıc | 21 |
| yapısı kontrol akımları (akımlar nA). | . 39 |
| Çizelge 2.9 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 2-D sınıflandırıc | 21 |
| yapısı kontrol akımları (akımlar nA). | . 39 |
| Çizelge 2.10 : Akım kaynakları yerine kullanılan direnç değerleri | . 40 |
| Çizelge 2.11 : Sınıflandırıcı devrelerin karşılaştırılması. | . 42 |
| Çizelge 3.1 : ÇAD devresi MOS tranzistorların boyutları. | . 57 |
| Çizelge 5.1 : Kuantalayıcı Yapısında Kullanılan Çekirdek Devre Parametreleri | . 72 |
| Çizelge 5.2 : Farklı örüntüler için y _i (i=1,,5) çıkış değerleri | . 75 |
| Çizelge 5.3 : n çıkışlı akım çoğullayıcı devresi MOS tranzistorların boyutları | . 80 |
| Çizelge 5.4 : Şekil 5.11'deki iris verisi sınıflandırıcısı test kümesi. | . 81 |
| Çizelge 5.5 : İris sınıflandırıcısı ÇAD direnç değerleri | . 82 |
| Çizelge 5.6 : İris sınıflandırıcısı çekirdek devre kontrol akımları | . 82 |
| Çizelge 5.7 : Şekil 5.15'deki Haberman verisi sınıflandırıcısı test kümesi | . 86 |
| Çizelge 5.8 : Haberman sınıflandırıcısı ÇAD direnç değerleri | . 86 |
| Çizelge 5.9 : Haberman sınıflandırıcısı çekirdek devre kontrol akımları | . 86 |
| Çizelge 5.10 : Şekil 5.17'deki iris verisi sınıflandırıcısı test kümesi. | . 88 |
| Çizelge 5.11 : Şekil 5.17'de c1 sınıfı için ÇAD direnç değerleri ve ÇD akımları | . 90 |
| Çizelge 5.12 : Şekil 5.17'de c2 sınıfı için ÇAD direnç değerleri ve ÇD akımları | . 90 |
| Çizelge 5.13 : Şekil 5.19'daki haberman verisi sınıflandırıcısı test kümesi | . 92 |
| Çizelge 5.14 : Şekil 5.19'da c1 sınıfı için ÇAD direnç değerleri ve ÇD akımları | . 93 |
| Çizelge 5.15 : Sınıflandırıcı başarımları karşılaştırması | . 93 |

ŞEKİL LİSTESİ

<u>Sayfa</u>

| Şekil 1.1 : Donanımsal gerçeklenmiş sınıflandırıcı bloğu [36]. | 5 |
|--|------|
| Sekil 2.1 : Çekirdek yapı geçiş karakteristiği. | . 13 |
| Sekil 2.2 : Čekirdek vapinin blok divagrami. | . 13 |
| Sekil 2.3 : n×1 boyutlu sınıflandırıcı devrenin blok diyagramı | . 14 |
| Sekil 2.4 : Tek boyutlu sınıflandırıcı ile elde edilmek istenilen giriş-çıkış | |
| karakteristiği | . 15 |
| Sekil 2.5 : 2×1 boyutlu cok seviveli sınıflandırıcı yapısı blok seması. | . 15 |
| Sekil 2.6 : MATLAB programi ile elde edilen 2×1 boyutlu cok seviyeli siniflandır | 101 |
| devresinin (x_1-x_2) -v karakteristiği. | . 16 |
| Sekil 2.7 : Akım-modlu cekirdek devre-1 yapısının blok diyagramı. | . 17 |
| Sekil 2.8 : Akım-modlu cekirdek devre-1 yapısının gecis karakteristiği. | . 17 |
| Sekil 2.9 : Cekirdek devre-1 ic vapısının islevsel divagramı. | . 17 |
| Sekil 2.10 : Cekirdek devre-1'in giris katı ve evirici | . 18 |
| Sekil 2.11 : NOR kapısı ve cıkıs katı. | . 19 |
| Sekil 2.12 : Cekirdek devre-1 yapısı için I _{out} akımının I _{in} akımı ile değişim | |
| karakteristiği ($I_{H2} \neq 0$). | . 20 |
| Sekil 2.13 : Cekirdek devre-1 vapısı için I _{out} akımının I _{in} akımı ile değisim | |
| karakteristiği ($I_{H2}=0$). | . 20 |
| Sekil 2.14 : Cekirdek devre-1 vapısı icin V _{out} geriliminin I _{in} akımı ile değisim | |
| karakteristiği | . 21 |
| Sekil 2.15 : Giriş-çıkış işareti yayılma gecikmesi. | . 21 |
| Sekil 2.16 : Cekirdek devre-1'in yayılma gecikmesi benzetim sonucu | . 22 |
| Sekil 2.17 : Cekirdek devre-1 ile gerçeklenen 1×1-D sınıflandırıcı blok diyagramı. | . 22 |
| Şekil 2.18 : CD-1 ile gerçeklenen 1-D sınıflandırıcı için I _{out} akımının I _{in} akımı ile | |
| değişim karakteristiği. | . 23 |
| Sekil 2.19 : CD-1 ile gerçeklenen 2×1-D sınıflandırıcı blok diyagramı. | . 24 |
| Şekil 2.20 : CD-1 ile gerçeklenen 2×1-D sınıflandırıcı (I _{in1} -I _{in2})-I _{out} karakteristiği. | . 25 |
| Şekil 2.21 : Akım-modlu ÇD-2'nin blok yapısı. | . 25 |
| Şekil 2.22 : Akım-modlu ÇD-2 blok diyagramı iç yapısı [55] | . 26 |
| Şekil 2.23 : Eşik devresinin giriş-çıkış karakteristiği. | . 26 |
| Şekil 2.24 : ÇD-2 yapısının gerçekleştirilme blok şeması | . 27 |
| Şekil 2.25 : Pozitif geribeslemeli eşik devresi devre şeması. | . 28 |
| Sekil 2.26 : Pozitif geribeslemeli eşik devresi osiloskop çıktısı | . 28 |
| Şekil 2.27 : Pozitif geribeslemeli eşik devresi benzetim karakteristiği | . 28 |
| Şekil 2.28 : Eşik devresi devre şeması | . 29 |
| Şekil 2.29 : ÇD-2 sınıflandırıcı devresi giriş-çıkış karakteristiği. | . 29 |
| Şekil 2.30 : ÇD-2 devre şeması [55] | . 30 |
| Şekil 2.31 : Eşik devresinin giriş-çıkış karakteristiği. | . 31 |
| Şekil 2.32 : ÇD-2 yapısı giriş-çıkış karakteristiği. | . 31 |
| Şekil 2.33 : ÇD-2 yapısı yayılma gecikmesi benzetimi | . 32 |

| Şekil 2.34 : ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi | . 33 |
|---|--------------|
| Şekil 2.35 : Tek boyutlu sınıflandırıcının giriş-çıkış karakteristiği. | . 33 |
| Şekil 2.36 : ÇD-2 ile gerçeklenen 1×1-D sınıflandırıcı giriş-çıkış benzetimi | . 34 |
| Şekil 2.37 : Zayıf evirtimde çalışan eşik devresi giriş-çıkış karakteristiği | . 36 |
| Şekil 2.38 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği | . 36 |
| Şekil 2.39 : Zayıf evirtimde çalışan ÇD-2 için yayılma gecikmesi benzetimi | . 37 |
| Şekil 2.40 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği Monte | |
| Carlo analizi. | . 38 |
| Şekil 2.41 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 1-D sınıflandırıcı | |
| giriş-çıkış karakteristiği | . 38 |
| Şekil 2.42 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 2-D sınıflandırıcı | |
| (I _{in1} -I _{in2})-I _{out} giriş-çıkış karakteristiği. | . 39 |
| Şekil 2.43 : ÇD-2 yapısının testinde kullanılan akım kaynağı modeli | . 40 |
| Şekil 2.44 : ÇD-2 giriş-çıkış kararteristiği osiloskop sonucu. | . 40 |
| Şekil 2.45 : ÇD-2'nin serim çizimi. | . 41 |
| Şekil 2.46 : ÇD-2'nin serim sonrası giriş-çıkış karakteristiği. | . 41 |
| Şekil 2.47 : 2×1 boyutlu sınıflandırıcının farklı çıkış seviyeleri için genel hali | . 44 |
| Şekil 3.1 : Lineer olarak sınıflandırılamayan veri kümesi | . 45 |
| Şekil 3.2 : Eğik Izgaralı veri sınıflandırıcısı. | . 46 |
| Şekil 3.3 : Dikdörtgen Izgara olmayan veri bölgeleri | . 46 |
| Şekil 3.4 : ÇAD yapısı blok diyagramı sembolik gösterimi | . 47 |
| Şekil 3.5 : Şekil 3.2'deki bloğun iç yapısı. | . 47 |
| Şekil 3.6 : Paralel bağlanmış sınıflandırıcı devresi | . 48 |
| Şekil 3.7 : Eğik ızgaralı veri sınıfları | . 48 |
| Şekil 3.8 : Eğik ızgaralı sınıflandırıcı çıkışının 3-D I _{out} -(x ₁ -x ₂) karakteristiği | . 49 |
| Şekil 3.9 : Verilerin ÇAD ve ÇD kullanılarak sınıflandırılması | . 49 |
| Şekil 3.10 : ÇAD devresi blok şeması. | . 50 |
| Şekil 3.11 : CCII devresi blok diyagramı. | . 51 |
| Şekil 3.12 : CCII devre şeması [65] | . 52 |
| Şekil 3.13 : DO-CCII devre blok diyagramı | . 52 |
| Şekil 3.14 : DO-CCII devre şeması. | . 53 |
| Şekil 3.15 : DO-CCII devresi için V_x 'in V_y ile değişim karakteristiği | . 53 |
| Şekil 3.16 : DO-CCII devresi için $ \varepsilon_v $ 'nin V _y ile değişim karakteristiği | . 54 |
| Sekil 3.17 : DO-CCII icin I_{7+} ve I_{7-} akımlarının I_{7-} ile değisim karakteristiği | . 54 |
| Sekil 3 18 · DO-CCII devresi icin $ c $ 'nin L ile değişim karakteriştiği | 55 |
| Şeki 5.10 . Do con devresi için $ c_{i1} $ min için degişim karakteristiği | |
| Şekil 3.19 : DO-CCII devresı ıçın $ \varepsilon_{i2} $ 'nın I_x ile değişim karakteristiği | . 55 |
| Şekil 3.20 : ÇAD yapısının CCII ve DO-CCII yapıları ile gerçeklemesi | . 55 |
| Şekil 3.21 : ÇAD devresi için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim | |
| karakteristiği ($R_2/R_3 = 5$) | . 56 |
| Sekil 3.22 : Cesitli $k = R_1 / R_2$ değerleri için CAD devresi V _v - I _{z+} karakteristiği | . 56 |
| Sakil 1 1 • Lineer olarak sınıflandırılamayan örnek veri kümesi | 50 |
| Sakil 1.2 • Histogram karakteristiği | . <i>5)</i> |
| Sekil 4 3 • Cift eşik doğrularının gösterilimi | . 00 64 |
| Sekil 4.4 : n girişli tek katlı nersentron vanışı | 68 |
| Sekil 5.1 : Sekiz seviveli kuantalavici devre blok vanisi [55] | 71 |
| Sekil 5.2 : Sekiz seviyeli kuantalayıcı devresi II. karakteristiği | 72 |
| Sekil 5.3 : Örnek örüntüler | 73 |
| Sekil 5.4 : Sahlon hücre gösterimi | 73 |
| geni et . , guoron nuere Bosternin. | . , 5 |

| Şekil 5.5 : Karakter tanıma sınıflandırıcısı blok diyagramı | . 74 |
|---|------|
| Şekil 5.6 : İlk beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları | . 75 |
| Şekil 5.7 : Son beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları | . 76 |
| Şekil 5.8 : Hatalı test şablonları. | . 77 |
| Şekil 5.9 : Karakter tanıma sınıflandırıcısı hata düzeltme benzetim sonucu | . 77 |
| Şekil 5.10 : İzdüşürülmüş iris verilerinin orijine uzaklıkları | . 79 |
| Şekil 5.11 : İris verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu | |
| oluşturulmuş) | . 79 |
| Şekil 5.12 : Akım çoğullayıcı devre şeması [65] | . 80 |
| Şekil 5.13 : Şekil 5.11'deki İris verisi sınıflandırıcısı benzetim sonucu. | . 83 |
| Şekil 5.14 : İzdüşürülmüş haberman verilerinin orijine uzaklıkları. | . 84 |
| Şekil 5.15 : Haberman verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu | |
| oluşturulmuştur). | . 85 |
| Şekil 5.16 : Şekil 5.15'deki haberman verisi sınıflandırıcısı benzetim sonucu | . 87 |
| Şekil 5.17 : İris verisi sınıflandırıcı bloğu (öğrenme algoritması sonucu | |
| oluşturulmuş) | . 89 |
| Şekil 5.18 : Şekil 5.17'deki iris verisi sınıflandırıcısı benzetim sonucu. | . 90 |
| Şekil 5.19 : Haberman verisi sınıflandırıcı bloğu (perseptron öğrenme algoritması | |
| sonucu oluşturulmuş). | . 91 |
| Şekil 5.20 : Şekil 5.19'daki Haberman verisi sınıflandırıcısı benzetim sonucu | . 93 |
| - | |

xiv

AYARLANABİLİR CMOS SINIFLANDIRICI DEVRELERDE YENİ OLANAKLAR

ÖZET

Bu çalışmada, ayarlanabilir sınıflandırıcı devreleri ve uygulama alanları incelenmiştir. Sınıflandırma işlemi, benzer özellik taşıyan objelerin farklı özellikte olanlardan ayırt edilmesi şeklinde tanımlanabilir ve otomatik hedef belirleme, yapay zekâ, yapay sinir ağları, analog-sayısal dönüştürücüler, tıbbi tanı, kuantalama, görüntü işleme, istatistik gibi konularda kullanım alanı bulur. Diğer yandan, gerek gerçek dünyada gerekse sayısal dünyada, verilerin sınıflandırılması büyük önem taşımaktadır. Sınıflandırma yöntemleri ilk olarak 1960'lı yıllarda örüntü sınıflandırma adı altında görülmeye başlanmış ve ilişkin algoritmalarda basit yapılar ele alınmıştır; ilk gerçeklenen yapıda en yakın komşu yakınsaması kullanılmıştır.

Bugüne kadar sınıflandırma işlemi, çeşitli algoritmalar yardımıyla genellikle yazılımsal olarak yapılmıştır. Oysaki gerçek zamanda çalışma gerektiren bazı uygulamalarda, sınıflandırma işleminin donanımsal olarak da gerçeklenmesi önem kazanmaktadır.

Bu amaçla, çalışmanın donanımsal gerçeklemeyle ilgili kısmında, önce çekirdek devre diye adlandırılan temel bir yapı tasarlanmış ve bu çekirdek devrelerden oluşan çok girişli-çok çıkışlı bir sınıflandırıcı mimarisi geliştirilmiştir. Bu sınıflandırıcı mimarisi ile sınıflandırılabilen ve sınıflandırılamayan veri kümeleri incelenmiş, sınıflandırılamayan veri kümelerinin ayırt edilebilmesi için çekirdek devre yapıları ile kullanılabilecek Çarpan Devre yapısı gerçekleştirilmiştir. Dolayısıyla gerek sadece çekirdek devre yapıları, gerekse çarpan devre yapıları ile beraber kullanılarak veri kümelerinin uygun kontrol parametreleri ile sınıflandırılabileceği gösterilmiştir. Bu kontrol parametrelerinin sağlayan eğitim algoritmaları da incelenmiştir.

Sonuç olarak bu çalışmada, sınıflandırma işlemini donanımsal yapılar ile gerçekleştirebilecek, ayarlanabilir, eğitilebilen yeni sınıflandırıcı devreleri tasarlanmış, sağladıkları yeni olanakların gerçek dünyada bulunan ve de önemli olan uygulamalarda incelenmesi ile elde edilen sonuçlar verilmiş ve sınıflandırma konusundaki etkinlikleri ortaya konulmuştur.

xvi

NEW POSSIBILITIES IN TUNABLE CMOS CLASSIFIER CIRCUITS

SUMMARY

In this thesis, new possibilities in CMOS classifier circuits and their applications are investigated. The aim of classification is to assign an unknown object to a class containing similar objects. Classifier circuits can find applications in various fields of applied science such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks, analog digital converters, quantizers and statistics. Therefore, classification is especially important in the real world applications or in the digital world. Firstly, basic classification methods using the nearest neighbourhood algorithm have been seen in 1960 as pattern recognition.

Nowadays classification is generally achieved with the help of some algorithms aided with computer programs. However, hardware implementation of classifier circuits are important for the some applications that require real-time processing.

For that reason, in this thesis firstly hardware implementation of a basic classification unit called core cell is presented. A multiple-input and multiple-output classification topology is constructed with these core cells. The data sets that can be classified or non-classified with that multiple-input and multiple-output classifier circuits have been investigated. A Scaler Circuit has been realized with core cells and used to classify data sets. As a result it is shown that the data sets, with only core cells or together with scaler circuits, can be classified with the appropriate control parameters. Learning algorithms have been investigated, developed and applied to obtain these control parameters.

To conclude, in this thesis custom tunable CMOS classifier circuits have been designed, tested and applied. The test applications have been chosen from real world problems and the results have verified the effective performance of the classifier topologies and circuits.

1. GİRİŞ

Sınıflandırma genellikle otomatik hedef belirleme, yapay zekâ, yapay sinir ağları, analog-sayısal dönüştürücüler, tıbbi tanı, kuantalama, görüntü işleme, istatistik gibi konularda kullanılmaktadır [1, 2]. Bu konuda yapılan çalışmalar yazılımsal ve donanımsal olmak üzere iki kısımda incelenebilir. Literatürde gerek yazılımsal gerekse donanımsal sınıflandırıcılarla ilgili birçok yayınlara rastlanmaktadır [3,4]. Çok yaygın olan yapay sinir ağı tabanlı sınıflandırıcılara ise hem yazılımsal hemde donanımsal olarak bakmak daha doğru olur. Genellikle donanımsal olarak tasarlanmış sınıflandırıcılar, farklı ağ yapıları sentezleyen Yapay Sinir Ağlarının (YSA) gerçeklemesidir. YSA algoritmalarının büyük bir çoğunluğu bilgisayara uyarlanabilmektedir. YSA, biyolojik nöron hücresinin yapısı ve öğrenme özelliklerinden esinlenerek geliştirilmiş bir hesaplama sistemi olup sınıflandırmaya çok elverişlidir. Bu ağların mimarisini ve öğrenme algoritmalarını gelistirmeve yönelik literatürde birçok çalışmalar bulunmaktadır [5, 6]. Bu çalışmaların geliştirdikleri modeller yazılım ve donanım ortamında kullanım sağlamışlardır [7-9]. YSA'nın seçilmesindeki başlıca neden ise, çok karmaşık ve çok fazla parametre içeren durumlarda bile kullanılabiliyor olmalarıdır [10]. Yazılımsal olarak çalıştırıldıklarında sakıncalarının başında, gerçek zamanda çalıştırılamamaları, öğrenme algoritma süresinin fazla olması, sınıflandırma yapacağı nesnelerin birbirine çok benzer özellikler göstermesi gelir; ayrıca parametre sayısı çok arttığında yavaş çalışır hale gelmeleri ise diğer bir sakıncadır [11]. Bütün bunların ana nedeni, paralel çalışma esasına göre ortaya atılmış YSA'nın, ardışıl makinelere yönelik algoritmalarla çalıştırılmasıdır; dolayısıyla mevcut sınıflandırıcı algoritmalarının yazılımsal olmalarına karşın, hızlı ve gerçek zamanda çalışma arzu edildiğinde, donanımsal olarak gerçeklenmeleri gerekliliği doğmuştur [12].

1.1 Literatür Taraması

1.1.1 Yazılımsal yöntemler

Genel bakış açısından, bir sınıflandırıcı tasarımı iki farklı kategoride ele alınabilir: eğitimli ve eğitimsiz tasarım. Eğitimli tasarımda daha önce tanımlanmış bilinen bir giris sınıfının bilinen çıkışlarıyla, sınıflandırıcının aynı girişlere karşı düşen çıkışları karşılaştırılarak bir hata terimi tanımlanır. Bu terim sıfır oluncaya kadar, sınıflandırıcının parametreleri değistirilir. Eğitimsiz sınıflandırmada ise bir veri eğitim kümesi bulunmayıp, bir yakınlık kuralına göre verilerin gruplandırılmasının düşünülebilir [2]. Literatürde yapılması olarak kullanılan sınıflandırma yaklaşımlarının çoğunluğu eğitimli sınıflandırma kategorisinde yer almaktadır [13]. Bu kategoride yapay sinir ağları ve en yakın komşu gibi yaklaşımlar kullanılmaktadır [3]. YSA tabanlı olan sınıflandırma yapıları, günümüzde en yaygın kullanılan, başarımını ispatlamış çok güçlü sınıflandırıcı türleridirler [14]. YSA'ya yönelik ilk çalışma, 1943 yılında Mc-Culloch ve Pitts'in tarafından yapılmıştır [15]. Bu calışmada basit mantık fonksiyonlarının gerçekleştirildiği matematiksel bir model kullanılmıştır. Daha sonraki çalışmalarda farklı mimari ve öğrenme yapısına sahip ağlar geliştirilmiştir [16]. Bu ağlardan çok katmanlı algılayıcılar ve radval tabanlı fonksiyon ağları literatürde sınıflandırma gibi pratik uygulamalarda oldukça yaygın olarak kullanılmıştır [17]. Ayrıca YSA'ların genelleme yeteneklerinin olması, çok büyük boyutlu örüntü tanıma uygulamaları açısından da oldukça önemlidir [18, 19].

Sınıflandırma işleminin bir başka yöntemi de, örüntüleri bulundukları uzayda tanımlı bir mesafe fonksiyonuna (metrik) göre, kendilerine en yakın sınıflara minimum hata ile atamaktır [20].

Örüntü sınıflandırma işlemi temel olarak iki gruba ayrılabilir: bunlardan ilki matematiksel ve istatistiksel tabanlı olanlar, ikincisi ise YSA tabanlı algoritmalardır. Sınıflandırıcılar başarım düzeyi açısından karşılaştırıldığında, YSA'ları kullanılarak elde edilen sınıflandırıcı sonuçları istatistiksel tabanlı sınıflandırıcı sonuçlarından daha başarılı olduğu görülmüştür; ancak öğrenme sürelerinin de uzun olduğu bilinmektedir [19-21]. Bazı çalışmalarda karma yöntemler de önerilmiştir [22].

Diğer bir sınıflandırma yöntemi ise destek vektör makinesidir. Bu yöntem iki sınıflı veriye ait nokta kümesini ayıran en iyi bir soyut-düzlem bulmaya çalışan bir tekniktir. Diğer bir ifade ile bu yöntem veri sınıflarını uygun soyut-düzlemler ile ayırmaya çalışır. Uygun soyut-düzlemin bulunması ise sınıflandırılacak veriler ile soyut-düzlemin mesafesinin minimizasyonu ile yapılır. Bu yöntemin sakıncaların başında ise, belirli veriler için en iyi ayırma düzlemi bulunduktan sonra yeni veriler geldiğinde, sadece yeni verilere değil eski verilere de ihtiyaç duyulması gelir [23].

Literatürde veri sınıflandırması radyal tabanlı fonksiyonlar kullanılarak da yapılmaktadır [24]. Buradaki veri sınıflandırma yönteminde, sınıflandırma sonuçları belli olan belirli veriler için ağ parametreleri saptanır ve bu parametreler benzer dağılım gösteren başka veriler için de kullanılır. Başka bir deyişle tekrar eğitilmesi gerekmemektedir. Ancak radyal tabanlı fonksiyonlar kullanıldığında doğru sonuç elde etme kesinliği azalırken hızlı sonuç alınabilmektedir; oysa destek vektör makinesi ile durum tam tersi olmaktadır [4].

Örüntü sınıflandırma için kullanılan bir yöntem de vektör kuantalayıcıdır. Bu yöntem örüntü tanımada olduğu kadar, özellikle veri sıkıştırılmasında da kullanılmaktadır. Vektör kuantalayıcı yapılarından, giriş dizisinin örnek dizilerden en çok hangisine benzediğine bulmakta yararlanılır [20,25]. Vektör kuantalama, genellikle daha hızlı bir performans elde etmek için, özellikleri birbirine yakın bir işaret kümesinin, tek bir vektörle temsil edilerek, tek bir kodla kodlanması şeklinde tanımlanabilir.

Bu veri sınıflandırma yöntemlerinden başka literatürde, sınıflandırma yazılımlarında kullanılabilen, k-en-yakın komşu (k-th nearest neighbourhood) ve Voronoi algoritması gibi çeşitli yöntemler de vardır [3]. Bu yöntemlerden, Voronoi diyagramı yaklaşımı örüntü sınıflandırmada farklı veri girişlerinin ayırt edilmesi amacıyla kullanılmaktadır [13].

Son olarak, yine literatürde özellikle sınıflandırma işleminin bir alt kümesi olan analog-sayısal dönüştürme işlemi, parça parça lineerleştirme yaklaşımı ve de nörobulanık sistemlerde kullanılan trapezoidal geçiş fonksiyonu da ayrıntılı olarak incelenmiştir [26-28]. Trapezoidal geçiş fonksiyonu yardımı ile görüntü işleme konusunda yapılmış çeşitli uygulamalar bulunmaktadır [29]. Ayrıca bir çeşit veri sınıflandırma yöntemi olarak düşünülebilecek kuantalama da, kodlama ve veri sıkıştırma uygulamalarında yaygın olarak kullanılmaktadır [30].

1.1.2 Donanımsal yapılar

Literatürde donanımsal olarak gerçekleştirilmiş sınıflandırıcı yapıları, gerçek zamanlı uygulamalar için elverişli olmaktadırlar. Bu yapılar genellikle yapı birimi bir nöron modeli olan, farklı ağ yapıları sentezleyen, programlanabilen veya programlanamayan devre ve tümdevre mimarileri olarak karşımıza çıkmaktadır. Özellikle yapay sinir ağlarının donanım gerçeklemeleri olan nöroişlemciler, standart mikroişlemciler ile uygulamalarda kullanılmaktadır. Böylece nörol hesaplanmalardan donanımsal olarak faydalanılmış olunur. Ancak bu donanım uygulamaları giriş-çıkış sayısı, ağ tipi, sabit aktivasyon fonksiyonu, sınırlı çalışma aralığı gibi bazı kısıtlamalar içermektedir. Ancak YSA'nın yazılım ortamında elde edilen esneklik, matematik işlem kabiliyetinin üstünlüğü gibi yeteneklere donanım gerçeklemelerinin getirdiği kısıtlamalar nedeniyle tam olarak ulaşılamamaktadır [5]. Bunun nedeni donanımların yazılım ortamındaki kadar esnek olmayışıdır. Ayrıca yazılım ortamında geliştirilen YSA algoritmalarının donanım uygulamaları için uygun olmayışı, işlem karmaşıklığının yüksek oluşu gibi nedenler gösterilebilir. Literatürde yer alan YSA'nın donanım gerçeklemeleri, uygulama amacına ve kullanılan teknolojiye bağlı olarak çeşitlilik göstermektedir. Bu donanım gerçeklemeleri genel olarak uygulamaya özgü, özel amaçlı tümdevre yapıları ile genel amaçlı tümdevre yapıları olarak karşımıza çıkmaktadır. Belli bir sınıflandırma mimarisi ve algoritmasına uygun tasarlanan tümdevreler dışında standart işlemciler üzerinde de çeşitli topolojiler sentezlenebilmektedir. Uygulamaya özgü tasarlanan tümdevreler, belirli bir uygulamaya göre tasarlandığından, sınırlı büyüklüklere sahip olup böyle bir tümdevrenin başka bir uygulama için kullanımı uygun değildir [31]. Genel amaçlı tümdevreler bu kısıtlamayı en aza indirmek amacıyla tasarlanmıştır. Gerçeklenen tümdevreler genel veya özel amaçlı olması açısından farklılık göstermesi dışında, uygulanan mimarinin analog, sayısal veya karma tasarlanmış olmasına, VLSI tasarım tekniği ve teknolojisine, eğiticili veya eğiticisiz öğrenme kuralına sahip olmasına, ağırlıkların tüm devre üzerinde saklanabiliyor veya saklanamıyor olmasına, tümdevre üzerinde saklanan ağırlıkların analog veya sayısal olmasına, eğitim işleminin tümdevre üzerinde yapılıp yapılmamasına, öğrenme sürecinde kullanılan algoritmaya bağlı olarak literatürde çeşitlilik göstermektedir [32,33]. Ayrıca literatürde yapay sinir ağı donanımları ile ilgili çalışmalar da ayrıntılı bir şekilde ele alınmıştır [34].

Literatürde kullanılan diğer bir donanımsal sınıflandırıcı yapısı da, yük tabanlı, sabit ağırlıklı olan Hamming sınıflandırıcısıdır [35]. Gerçeklenen bu devrenin üstünlükleri, tek besleme gerilimi (+5 V) kullanılıyor ve statik güç tüketimi olmadan da yüksek hızlı çalışabiliyor olmasıdır; ancak en önemli sakıncası ise ağırlıkları değiştirilemediğinden sabit programlı olmasıdır.

Sınıflandırmada, iki vektör dizisinin farkını alarak sınıflandırma işlemini gerçekleştiren devreler de mevcuttur. Böyle bir devre 0.35 µm CMOS teknolojisi kullanılarak önerilmiştir [36].



Şekil 1.1 : Donanımsal gerçeklenmiş sınıflandırıcı bloğu [36].

Donanımsal olarak gerçeklenen iki vektör dizisinin farkını hesaplayan genel bir sınıflandırıcı yapısı Şekil 1.1'de gösterilmiştir. Bu şekildeki $(x_1, ..., x_i, ..., x_n)$ giriş dizisi vektörünü $((T_{i1}, T_{i2}, ..., T_{1n}), i=1, ...m)$ ise giriş vektörüne uzaklıkları hesaplanacak olan sabit örnek vektör dizilerini göstermektedir. Kazanan Hepsini Alır (KHA) bloğu ise hesaplanmış uzaklıkların karşılaştırılıp çıkışa yollandığı bloktur. Gerçeklenen bu devreler, genel olarak iki temel bloktan oluşmaktadır. Uzaklık hesaplamalarının yapılmakta olduğu ilk bloğun yapısının özelliği, paralel bağlanmış temel fark alma devrelerinden oluşmasıdır. Bu bloğun yapısı gereği, uzaklık hesabı, algoritma olarak sınıflandırmada kullanılan bir yöntemin donanımsal olarak gerçeklenmesiyle yapılmaktadır. İkinci blokta ise genellikle KHA tipinde çıkış katları kullanılmaktadır [37]. Çıkış katlarında kullanılan bu yapıların temel amacı ise basit bir karşılaştırma yapmaktan ibarettir. Bunun nedeni, literatürde önerilen donanımsal sınıflandırıcı yapılarında, iki veri arasındaki Hamming uzaklığının hesaplanması ve elde edilen uzaklık çıktılarının birbirleriyle karşılaştırılması yapılarak sınıflandırma işleminin gerçeklenmesidir [38]. Literatürde sabit ağırlıklı Hamming sınıflandırıcılarının kullanılmasının bir başka sebebi ise yüksek hızlı uygulamalara elverişli olmalarıdır. Böyle bir sınıflandırıcı devresi 2.4 µm CMOS teknolojisi kullanılarak gerçekleştirilmiş ve tam şayıların tanınmasında kullanılmıştır; sınıflandırma hızı 10 MHz olarak verilmiştir [39]. Bu tip yapılarda ağırlıkların değişmiyor olması başka uygulamalarda kullanılamaması açısından bir dezavantaj olarak karsımıza çıkmaktadır. Hamming uzaklığı kullanan sınıflandırıcılar analog olarak donanımsal gerçekleştirildiklerinde eşleşme problemi ile hatalı karar vermeye sebep olmaktadırlar; dolayısıyla eşleşme hataları ve sınırlamaları literatürde incelenmiştir [40].

Sınıflandırma işleminde kullanılan diğer bir yaklaşım ise, yakınlık ölçütü olarak Euclid uzaklığının kullanıldığı yöntemlerdir [41]. Bu yöntemler, hücre devrelerinin paralel bağlanmasıyla temel uzaklık hesabı sağlandığı için, alan verimliği açısından çok elverişlidir. Literatürde, temel devre yapısının 4 tranzistor ve 2 kapasite elemanı ile gerçeklendiği örnekler vardır [38]. Ancak temel sakınca, bu devrelerde tasarım esnasında kapasite değerlerinin seçilmesi zorunluluğundan dolayı ağırlık katsayılarının kullanım aşamasında değiştirilememesidir. Dolayısıyla ağırlık katsayılarının değiştirilebilir olması sınıflandırıcıların genel amaçlı kullanımı açısından önem taşımaktadır. Tasarımlarda ağırlık değerleri hem analog hemde sayısal olarak tasarlanmıştır. Bu ağırlık değerlerinin sayısal ve tamsayı biçiminde önerildiği ve böylelikle kuantalama hatasının oluşmasının da engellendiği çalışmalar bulunmaktadır [42,43].

Sınıflandırmada başka bir yaklaşımda eğitilici öğrenme algoritmasına dayanan destek vektör makinesidir. Bu destek vektör makinesi doğrusal ayırt edilebilen iki sınıf problemlerinin çözümünden yola çıkarak doğrusal olarak ayırt edilemeyen veya

çoklu sınıf problemlerinin çözümünde kullanılan bir yöntemdir. Gerçek zamanlı sınıflandırma uygulamaları bu öğrenme algoritmasının eğitim süresinin uzun olmasından dolayı zor bir problem olmaktadır. Eğitimde harcanan bu hesaplama süresinin iyilestirilmesi amacıyla kısıtlı bir eğitim kümesi kullanılarak donanımsal olarak destek vektörleri elde edilmiştir. Bu kısıtlı kümeden elde edilen vektörler sınıflandırma başarımını azaltırken, yapının karmaşıklığının da azalmasını sağlamıştır. Literatürde bu şekilde tasarlanmış bir devre 0.5 µm CMOS teknolojisi kullanılarak gerçekleştirilmiş ve güç tüketimi 5.9 mW olarak verilmiştir [44]. Sınıflandırmada destek vektör makinesinin eğitim amacıyla kullanıldığı bir başka çalışmada ise, devre mimarisinde yüzen kapılar kullanılarak, analog işlemci blokları zayıf evirtimde çalışması sonucu nW'lar ile mW'lar mertebesinde düşük güç tüketim değerlerine ulaşılmıştır [45]. Destek vektör makineleride radyal tabanlı fonksiyonlar kullanılarak donanımsal olarak gerçeklenmiştir [46]. Bu fonksiyonlarda kullanılan Gauss fonksiyonunun ortalama değeri ve varyansı yüzen kapılar kullanılarak ayarlanabilmektedir. Gerçeklemelerin analog tasarımlarında akım aynaları ve logaritma tabanlı süzgeçler de kullanılmıştır. Ancak bu durum geniş yapay sinir ağlarında çok elverişli olmadığı için, direnç ve kuvvetlendiriciler tercih edilmiştir [46]. Ayrıca radyal tabanlı fonksiyon sınıflandırıcısına KHA yapısı da eklenerek analog vektör kuantalayıcıya dönüştürülmüştür [47].

Karar ağaçlarıda sınıflandırmada kullanılan yöntemler arasında yer almaktadır. Bu yöntem geniş veri kümelerinde dahi yüksek doğrulukla çalışmaktadır. Ancak veri boyutlarının geniş olduğu uygulamalarda yoğun algoritmalardan dolayı işlem süresi çok uzun olmaktadır. Bunun için karar ağacı sınıflandırıcısı donanımsal olarak FPGA (Field Programmable Gate Array) yapıları kullanılarak gerçeklenmiştir [48]. Sayısal tasarımların gürültü bağışıklığının iyi olmasından dolayı, yüksek doğruluğa sahip çıkışlar üretebilmektedir. Ayrıca sayısal tasarım tekniklerinden dolayı, donanım tanımlama dilleri (VHDL) kullanılarak gerçekleştirilen sayısal bir tasarım, FPGA yongalarına kolayca aktarılabilmekte ve hızlı prototipler üretilebilmektedir. Donanım tanımlama dilleri kullanarak tasarlanmış sınıflandırıcı mimarileri literatürde bulunmaktadır [49]. Sayısal tasarımların gürültü bağışıklığı gibi avantajlarının yanında, sayısal yapı blokları ile fonksiyon gerçekleştirmek analog devrelere göre çok daha fazla sayıda tranzistor gerektirmektedir. Yapılan
çalışmalarda genelde analog ve sayısal tasarım tekniklerinin olumlu özelliklerini bir arada kullanan karma tümdevre tasarımları yer almaktadır.

Bütün bu yukarıda sözü edilen sınıflandırıcı donanımlarına ek olarak literatürde örüntü tanımada kullanılan donanımsal sınıflandırıcı yapısı da karşımıza çıkmaktadır. Donanımsal olarak, 2 μ m CMOS teknolojisi ile 2.2×2.2 mm² alan üzerine gerçeklenmiş ve 6 adet nöron içeren örüntü sınıflandırıcı yapısı mevcuttur [50]. Bu yapıda kullanılan öğrenme kısmı mikrobilgisayar aracılığı ile gerçekleştirilmiştir ve devre mW'lar mertebesinde güç tüketmektedir [50]. Yüz ve karakter gibi karmaşık görsel örüntülerin sınıflandırılması için resimlerin sayısal bir kamera yardımıyla alınması ve sayısal işaret işleme teknikleri kullanılarak mikroişlemciler aracılığı ile işlenmesi sağlanır. Ancak bu işlemler büyük miktarda güç tüketimi gerektirmektedir. Bunların daha az güç tüketerek gerçekleştirilmesi için analog donanımsal görsel örüntü sınıflandırıcılar önerilmiştir [51]. Literatürde +5 V besleme gerilimi ile beslenen, 0.35 μ m CMOS teknolojisi ile gerçeklenen ve 1.25 mW güç tüketen uygulamalar bulunmaktadır [51].

Örüntü tanımada birçok sınıflandırıcıyı birleştirerek kullanmak gelişmiş bir yöntem olarak bilinmektedir. Literatürde karar ağaçlarının sınıflandırıcı olarak kullanıldığı örüntü tanıma devreleri de mevcuttur [52]. Ancak bu tip sınıflandırıcılarda genel başarım yüksek dahi olsa fazla miktarda bellek gereksinimi ve hesaplama süresinin uzunluğu bir dezavantaj olmakta ve gerçek zamanlı kullanılmalarına engel teşkil etmektedir. Akıllı fotosensörlerde, düşük çözünürlüklü örüntü sınıflandırıcı uygulamalarında kullanılabilirler. Ayrıca yapay sinir ağlarının farklı örnekler için eğitilebilme yetenekleri sayesinde fotosensor dizileri ile birlikte kullanılarak CMOS devreler ile gerçeklenmiş örüntü sınıflandırma uygulamaları mevcuttur [53].

Böylece, yukarıda literatürdeki mevcut sınıflandırıcı devreleri incelenmiş ve bu sınıflandırıcıların donanımsal olarak gerçekleştirilmesi için pek çok farklı yapıda donanıma özel öğrenme algoritmaları geliştirilmiştir. Donanımsal sınıflandırıcı tasarım teknikleri analog, sayısal ve karma olacak biçimde tümdevre yapıları gerçeklenmiştir. Bu yapıların bir kısmı genel amaçlı, bir kısmı da özel amaçlı tasarımlardır.

Tüm bu tasarımlarda, temel sınıflandırıcı yapısı kırmık alanında fazla yer kaplamayacak, güç tüketimi düşük olacak ve üretilmiş olan temel sınıflandırıcı tümdevreleri aralarında bağlanabilecek ve böylelikle çok girişli devrelerin elde edilmesine olanak tanıyacak bir sınıflandırıcı devresi gerçeklenmelidir. Ayrıca bu tümdevre gerçeklemesi ağırlıkları dışarıdan ayarlanabilecek şekilde tasarlanmalı ve ağırlık katsayılarının bulunması için tümdevreye uygun öğrenme algoritmaları geliştirilmelidir.

1.2 Ele Alınan Konu ve Problem

Bu çalışma kapsamında ele alınacak sorunlar ve beklentiler aşağıdaki gibi özetlenebilir.

- a) Literatürde, genel amaçlı esnek bir şekilde kullanılabilecek gerek kırmık alanında az yer kaplayacak, gerekse düşük güç tüketimine sahip olacak ayarlanabilir temel bir sınıflandırıcı yapısına ihtiyaç vardır. Sözü edilen sınıflandırıcı temel devrelerin aynı şekilde paralel, seri, ard arda ya da herhangi bir biçimde bağlanarak daha yüksek boyutlu verilerin sınıflandırılmasına olanak vermesini sağlayacak sınıflandırıcı yapılarına gereksinim duyulmaktadır.
- b) Diğer taraftan literatürde incelenen devrelerin çok önemli bir sakıncası da sabit programlı olmalarıdır. Başka bir deyişle bu devrelerin çeşitli veri tiplerine göre, verileri sınıflara ayıran bölgeleri belirleyen devre parametrelerinin kullanıcı tarafından saptanması ve/veya bir uygulamadan ötekine değiştirilebilmesine imkan tanıyacak tasarımlar gerçekleştirilmelidir.
- c) Yukarıda a) şıkkında bahsedilen devreler ile çok özel bir yapıda olan dikdörtgen ızgara bölgelerle ayrıştırılabilen veri tipleri sınıflandırılabilmektedir. Bu veri tipleri dışında kalan verilerin sınıflandırılması için uygun bir yöntem geliştirilmelidir. Ayrıca bu tasarımlara uyarlanabilecek öğrenme algoritmaları da oluşturulmalı ve yapılara uyarlanmalıdır.

1.3 Çalışmada Yapılanlar

Tez çalışmasının sınıflandırıcı devrelerinin tasarlanmasında yürütülen yöntem aşağıdaki şekilde özetlenmiştir:

Bölüm 2'de sınıflandırma amacıyla, önce 1.2 a)'da sözü edilen iki adet temel sınıflandırıcı devre bloğu tasarlanmıştır. Tasarlanan sınıflandırıcı devre blokları temel bir yapı niteliği taşımakta; diğer bir ifade ile tek boyutlu bir sınıflandırıcı devresi gibi davranmaktadırlar. Bu yapıya çekirdek devre denilmiş, hem yazılımsal hem de donanımsal gerçeklenerek benzetimleri ve karşılaştırılmaları sunulmuştur. Bu devreler ile elde edilebilecek sınıflandırma bölgeleri incelenmiş, bu devreler ile kurulan değişik topolojiler sayesinde ne tür farklı veri tiplerinin sınıflandırılabildiği gösterilmiştir. Çekirdek devrelerin çeşitli şekilde bağlanması ile çok boyutlu sınıflandırıcı devreleri gerçekleştirilmiştir. Böylece sınıflandırıcı devrelerin bazı veri tipleri için esnek yapıda tasarlanmış olmasının getirdiği kullanım kolaylıkları da gösterilmiştir. 1.2 b)'de, çekirdek devrenin kontrol parametrelerinin kullanıcı tarafından saptanabilmesi sağlanmıştır. Bu bölümde ayrıca devre parametrelerindeki değişimin devre davranışına etkisini incelemek amacıyla Monte Carlo analizleri yapılmış ve değişim etkisinden söz edilmiştir.

Yukarıda 1.2 c) ile ifade edilen sorunu gidermek amacıyla çekirdek devrelerin önüne yeni bir kat eklenerek oluşturulan Çarpan Devresi (ÇAD) ve benzetimleri Bölüm 3'te incelenmiştir. ÇAD yapısının nasıl gerçekleştirildiği ve bu ÇAD yapısı ile önerilen devrelerin beraber kullanımı ile hangi tür verilerin sınıflandırılabileceği de gösterilmiştir.

Önceki bölümlerde önerilen yapılar, özgün ve farklı olduklarından sınıflandırma algoritmaları ve geliştirilen devrelere uygulanması Bölüm 4'te ele alınmıştır. Fisher tabanlı algoritma, perseptron öğrenme algoritmaları incelenmiş, bu algoritmaların önerilen devreler ile nasıl kullanılacağı gösterilmiştir. Çift eşik doğrularının bulunmasında Fisher'in lineer diskriminant analizi yönteminden yararlanılarak yeni bir yöntem geliştirilmiştir.

Gerçekleştirilmiş olan sınıflandırıcının uygulamalarına Bölüm 5'de yer verilmiştir. Kuantalayıcı ve karakter tanıma uygulaması, önerilen devrelerin çeşitli uygulama alanlarında kullanılmasına örnek olarak verilmiştir. Geliştirilen sınıflayıcının veri kümelerine uygulanması amacıyla İris ve Haberman verileri Fisher tabanlı ve perseptron öğrenme algoritmaları ile sınıflandırılmış, sonuçlar karşılaştırılarak üstünlükleri gösterilmiştir.

6. bölüm tez kapsamında yapılan çalışmalar ve elde edilen gelişmelere ilişkin sonuç ve yorumların verilmesine ayrılmıştır. Bu bölümde ayrıca, tezde ele alınmış olan problemin devamı niteliğinde yapılabilecek çalışmalar için öneriler de bulunulmuştur.

2. SINIFLANDIRICI DEVRESİ TASARIMI VE BENZETİMLERİ

2.1 Sınıflandırıcı Yapısı Blok Diyagramı

Çalışmada öncelikle, basit donanımlarla sınıflandırıcıların nasıl gerçeklenebileceği ele alınmış ve sınıflandırma işlemi için temel bir çekirdek yapı ve bu yapının sınıflandırabileceği veri kümeleri incelenmiştir. Bu temel çekirdek yapılar ile oluşturulan topolojilerle daha karmaşık veri kümelerinin de sınıflandırılabileceği gösterilmiş ve gerçeklemeleri yapılmıştır. Çekirdek yapı (ÇY) taşı tek boyutlu bir sınıflandırıcı yapısı olarak düşünülmüş ve tasarlanmıştır. Gerçeklenmesi istenen geçiş eğrisi Şekil 2.1 ve blok diyagramı da Şekil 2.2'de verilmiştir. Bu çekirdek yapı temel sınıflandırıcı yapısını oluşturmaktadır ve çok boyutlu sınıflandırıcı yapılarının elde edilmesinde kullanılmıştır.



Şekil 2.1 : Çekirdek yapı geçiş karakteristiği.



Şekil 2.2 : Çekirdek yapının blok diyagramı.

Tasarımım, Şekil 2.1'deki karakteristikteki r_1 , r_2 ve k parametreleri dışarıdan kullanıcı tarafından kolayca değiştirilebilecek ve uygulanacakları sınıflama probleminden kolayca elde edilebilecek biçimde yapılmıştır. Giriş-çıkış karakteristiği aşağıdaki gibi ifade edilir:

$$y = \begin{cases} k & r_1 < x < r_2 \\ 0 & di \check{g} er \ haller de \end{cases}$$
(2.1)

Bu yaklaşımın bir üstünlüğü de çekirdek yapıların çeşitli şekillerde aralarında bağlanarak farklı tipteki ve boyuttaki verileri sınıflandırmak için kullanılabilmesidir; örneğin Şekil 2.3'de gösterildiği gibi paralel bağlanarak $n \times 1$ boyutlu bir sınıflandırıcı sistemi gerçekleştirilebilir. Daha karmaşık topolojiler ileriki aşamalarda verilecektir.



Şekil 2.3 : *n*×1 boyutlu sınıflandırıcı devrenin blok diyagramı.

Şekil 2.3'deki yapının çeşitli özel halleri göz önüne alınabilir; bir özel durum

$$x_1 = x_2 = \dots = x_n = x, \qquad r_1 < r_2 < r_3 < \dots < r_{(2n-1)} < r_{2n}$$
(2.2)

şeklinde verilebilir.

Bu durum için Şekil 2.4'de gösterilen giriş-çıkış karakteristiği elde edilir. Şekil 2.4'deki yapıdan da görüldüğü gibi bu şekilde gerçeklenmiş bir topoloji, tek boyutlu verileri, çıkış büyüklüğünün genliğine bakarak *n* tane farklı sınıfa ayırabilmektedir. Bu yapıdan aynı zamanda *x* büyüklüğünün kuantalanmasında da yararlanılabilir.



Şekil 2.4 : Tek boyutlu sınıflandırıcı ile elde edilmek istenilen giriş-çıkış karakteristiği.

Sınıflandırıcı yapılarının diğer bir uygulaması olarak, iki boyutlu ve çok seviyeli bir sınıflandırıcı tasarımı Şekil 2.5'deki topoloji ile elde edilebilir. Bu şekilde verilen topolojide çekirdek yapı-I ve II'ye x_1 giriş değişkeni, çekirdek yapı-III ve IV'e x_2 giriş değişkeni uygulanmıştır. Tasarlanan devreler akım modunda çalıştıklarından çıkışta bir toplama devresi yoktur. Böylece elde edilen sınıflandırıcının giriş-çıkış karakteristiğini daha iyi görebilmek amacıyla kontrol değişkenlerinin Çizelge 2.1'de verilen değerleri kullanılarak MATLAB benzetimi yapılmıştır.



Şekil 2.5 : 2×1 boyutlu çok seviyeli sınıflandırıcı yapısı blok şeması.

Çizelge 2.1 : Çekirdek Yapı Değişkenleri

| Çekirdek Yapı | | | | | | | | | | | |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | Ι | | | II | | | III | | | IV | |
| r_1 | r_2 | k_1 | r_3 | r_4 | k_2 | r_5 | r_6 | k_3 | r_7 | r_8 | k_4 |
| 10 | 20 | 10 | 40 | 50 | 40 | 10 | 20 | 30 | 40 | 50 | 50 |

Bu durumda Şekil 2.6'daki gibi bir giriş-çıkış karakteristiği elde edilebilir. Bu karakteristikten görüldüğü üzere 2 girişli, 1 çıkışlı ve çok seviyeli (bu örnekte 8 farklı sınıf, 0 düzlemi de dahil edilirse 9 farklı sınıf) sınıflandırıcı yapısı elde edilir.



Şekil 2.6 : MATLAB programı ile elde edilen 2×1 boyutlu çok seviyeli sınıflandırıcı devresinin (x_1 - x_2)-y karakteristiği.

Tasarlanmış ve matematiksel modeli verilmiş devrenin gerçeklenmesi Bölüm 2.2'de ele alınmıştır.

2.2 Çekirdek Devreler

2.2.1 Çekirdek devre-1

Çalışmada tek boyutlu bir sınıflandırıcı devresi olarak, akım modunda çalışmak üzere tasarlanan ve çekirdek devre-1 olarak adlandırılan yapının işlevsel davranışını gösteren blok diyagramı ve geçiş karakteristiği sırasıyla Şekil 2.7 ve Şekil 2.8'de verilmiştir [54]. Şekil 2.8'deki geçiş karakteristiğinden de görüldüğü gibi giriş değerleri I_1 ve I_2 arasında ise çıkış I_{H1} değerini, bunların dışında ise çıkış I_{H2} değerini almaktadır.



Şekil 2.7 : Akım-modlu çekirdek devre-1 yapısının blok diyagramı.



Şekil 2.8 : Akım-modlu çekirdek devre-1 yapısının geçiş karakteristiği.

Tasarlanmış olan çekirdek devre-1'in işlevsel blok diyagramı Şekil 2.9'da verildiği gibidir. Bu blok diyagramında giriş katı, giriş akımını gerilime dönüştürmek için kullanılırken NOR kapısı ile çıkışın, I_1 ile I_2 akımların arasında I_{H1} değerini alması sağlanmıştır. Karakteristiğinde sıfır olmayan kısmın elde edilmesi için kullanılmıştır. Çıkış değişkenin akım olması ise çıkış katı ile sağlanmaktadır.



Şekil 2.9 : Çekirdek devre-1 iç yapısının işlevsel diyagramı [54].

Şekil 2.8'de verilen giriş-çıkış karakteristiğinin tanım bağıntısı aşağıda gösterildiği biçimdedir:

$$I_{out} = \begin{cases} I_{H1} & I_1 < I_{in} < I_2 \\ -I_{H2} & di \check{g} er \ haller de \end{cases}$$
(2.3)

Şekil 2.9'daki giriş katı ile birlikte tasarlanan evirici devresi Şekil 2.10'da verilmiştir.



Şekil 2.10 : Çekirdek devre-1'in giriş katı ve evirici.

Buradaki diyot bağlı M_1 tranzistoru ve I_{BIAS} akımı giriş akımını gerilime dönüştürmek için kullanılır. M_2 ve M_3 tranzistorları da $I_{BIAS}+I_1$ ve $I_{BIAS}+I_2$ akımları ile kutuplanmıştır. Dolayısıyla devrede kullanılacak olan iki farklı eşik değeri elde edilmiş olur. Diğer bir deyişle buradaki M_1 , M_2 ve M_3 tranzistorlarından oluşan kat, hem bir eşik devresi hem de akımı gerilime dönüştürmek amacıyla kullanılmıştır. M_4 ve M_5 tranzistorları ise evirici olarak çalışmaktadır. Giriş katı ile V_1 ve V'_2 gerilimleri elde edilmiştir.

Devrede kullanılan NOR kapısı ve çıkış katı gerçeklenmesi Şekil 2.11'de verilmiştir. Çıkış karakteristiğinde sıfır olmayan kısmın elde edilmesi için NOR kapısı kullanılmıştır. Bu yapıdaki NOR kapısı M_6 , M_7 , M_8 ve M_9 tranzistorlarından oluşmaktadır. Dolayısıyla V_{out} çıkışı,

$$V_{out} = \begin{cases} V_{DD} & I_1 < I_2 \\ Vss & di ger haller de \end{cases}$$
(2.4)

ifadesi ile verilir.

Son olarak, kullanılan M_{10} , M_{11} , M_{12} ve M_{13} tranzistorları çıkış katını oluşturmaktadır. Çıkış katında kullanılan I_{H1} ve I_{H2} akım kaynakları geçiş karakteristiğindeki tepe noktalarının ayarlanmasını sağlamaktadır.



Şekil 2.11 : NOR kapısı ve çıkış katı.

Çıkış katındaki I_o^+ ve I_o^- akımları da V_{out} gerilimi cinsinden aşağıdaki ifadeler ile verilir:

$$I_{o}^{+} = \begin{cases} I_{H1} & V_{out} = V_{DD} \\ -I_{H2} & V_{out} = V_{SS}, \end{cases}$$
(2.5)

$$I_{o}^{-} = \begin{cases} -I_{H2} & V_{out} = V_{DD} \\ I_{H1} & V_{out} = V_{SS} \end{cases}$$
(2.6)

Dolayısıyla, tasarlanmış olan devrede giriş katında kullanılan I_1 ve I_2 akımları geçiş karakteristiğinin sıfır olmayan bölgesinin ayarlanmasına olanak tanımaktadır.

Tasarlanmış olan devrenin SPICE benzetimleri için çıkış akımı I_{out} için I_o^+ seçilmiştir ve 0.35 µm AMS CMOS SPICE teknoloji parametreleri kullanılmıştır (parametreler Ek A'da verilmiştir). Besleme gerilimi V_{DD} ve $V_{SS} \pm 1.25$ V olarak alınmış, kutuplama akımı $I_{BIAS}=10$ µA ve kontrol akımları $I_1=40$ µA, $I_2=80$ µA ve $I_{HI}=I_{H2}=20$ µA olarak seçilmesi durumu için I_{out} akımının I_{in} akımı ile değişim karakteristiği Şekil 2.12'de verilmiştir. Çekirdek devre-1 yapısının oluşturulmasında kullanılan tranzistor boyutları Çizelge 2.2'de verilmiştir.



Şekil 2.12 : Çekirdek devre-1 yapısı için I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2} \neq 0$).

Çalışmanın devamında I_{H2} akımı 0 alınmış, I_{H1} akımı I_H olarak gösterilmiş ve benzetimler yapılmıştır. Kontrol akımları I_1 =40 µA, I_2 =80 µA ve I_H =20 µA alınarak I_{out} akımının I_{in} akımı ile değişim karakteristiği çizdirilmiş ve Şekil 2.13'de verilmiştir. Çıkışın gerilim olması durumu içinse, V_{out} geriliminin I_{in} akımı ile değişim karakteristiği Şekil 2.14'de gösterilmiştir.

Çizelge 2.2 : Çekirdek devre-1 yapısının MOS tranzistor boyutları.

| MOSFET | W [µm] | L [µm] |
|--|--------|--------|
| $M_1, M_2, M_3, M_4, M_6, M_7, M_8, M_9, M_{10}, M_{12}$ | 10.5 | 0.7 |
| M ₅ , M ₁₁ , M ₁₃ , | 5.25 | 0.7 |



Şekil 2.13 : Çekirdek devre-1 yapısı için I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2}=0$).



Şekil 2.14 : Çekirdek devre-1 yapısı için V_{out} geriliminin I_{in} akımı ile değişim karakteristiği.

Çekirdek devre-1'in çalışma performansını belirlemek açısından yayılma gecikmesi de incelenmiştir. Yayılma gecikmesi incelemesinde girişe bir kare dalga uygulanmış ve çıkış işareti gözlenmiştir. Devrenin yayılma gecikmesi olarak Şekil 2.15'de gösterilen t_y süresi ile t_d süresinin aritmetik ortalaması kullanılmıştır; t_y süresi için, giriş işareti yükselme durumunda iken genliğinin %50'sine ulaştığı an ile çıkış genliğinin %50'ye ulaştığı an arasındaki fark, t_d ise t_y ile benzer şekilde işaretlerin düşme durumunda hesaplanan fark alınmıştır.



Şekil 2.15 : Giriş-çıkış işareti yayılma gecikmesi.

Çekirdek devre-1'in yayılma gecikmesi benzetimi için kontrol akım parametreleri I_1 =40 µA, I_2 =80 µA ve I_H =20 µA olarak ayarlanmıştır. Girişe genliği 70 µA olan bir kare dalga uygulanmış ve benzetim sonucunda gecikme, Şekil 2.16'da gösterildiği gibi 5 ns olarak bulunmuştur.



Şekil 2.16 : Çekirdek devre-1'in yayılma gecikmesi benzetim sonucu.

1×1 boyutlu bir sınıflandırıcı devre örneği için, dört adet çekirdek devre-1 yapısı Şekil 2.17'de gösterildiği biçimde paralel olarak bağlanır. Bu tek boyutlu sınıflandırıcı yapısında kontrol akımları ile, uygun bölgeye düşen veri ayırt edilmektedir. Çekirdek devre-1 yapılarına uygulanan kontrol akımları Çizelge 2.3'de verilmiştir.



Şekil 2.17 : Çekirdek devre-1 ile gerçeklenen 1×1-D sınıflandırıcı blok diyagramı.

| Çekirdek Devre-1 | I_1 (μ A) | I_2 (μ A) | $I_H(\mu A)$ |
|------------------|------------------|------------------|--------------|
| Ι | 20 | 80 | 130 |
| II | 130 | 180 | 90 |
| III | 230 | 310 | 100 |
| IV | 370 | 440 | 140 |

Cizelge 2.3 : 1×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.

Çekirdek devre-1 yapıları ile oluşturulmuş 1×1 boyutlu sınıflandırıcının SPICE benzetimi sonucu elde edilen I_{out} akımının I_{in} akımı ile değişim karakteristiği Şekil 2.18'de verilmiştir.



Şekil 2.18 : ÇD-1 ile gerçeklenen 1-D sınıflandırıcı için *I*_{out} akımının *I*_{in} akımı ile değişim karakteristiği.

Çekirdek devre-1 yapıları kullanılarak 1×1 boyutlu veriler dışında 2×1 hatta daha büyük $n \times m$ boyutlu veriler de sınıflandırılabilmektedir. Şekil 2.19'da da çok boyutlu verilerin sınıflandırılabilmesi amacıyla benzer topolojiden yararlanarak çekirdek devre-1 blokları ile kurulan yapı verilmiştir. Bu yapıda 4 adet çekirdek devre-1 yapısı birbirine bağlanmış ve çekirdek devre I ve II'ye birinci veri kümesi, III ve IV'e ise ikinci veri kümesi giriş olarak uygulanmıştır. Böylece 2×1 boyutlu veri sınıflandırıcı yapısı oluşturulmuştur.



Şekil 2.19 : ÇD-1 ile gerçeklenen 2×1-D sınıflandırıcı blok diyagramı.

İki boyutlu sınıflandırıcının benzetimini yapmak amacıyla Çizelge 2.4'de verilen kontrol parametreleri seçilmiş olup her bir çekirdek devre-1'in güç tüketimleri de ayrı ayrı gösterilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.

| Çekirdek Devre-1 | <i>I</i> ₁ (µA) | I_2 (μ A) | $I_{H}(\mu A)$ | Güç Tüketimi |
|------------------|----------------------------|------------------|----------------|--------------|
| Ι | 70 µA | 140 µA | 60 µA | 0.46 mW |
| II | 210 µA | 280 µA | 120 µA | 1.03 mW |
| III | 60 µA | 140 µA | 100 µA | 0.51 mW |
| IV | 280 µA | 360 µA | 80 µA | 1.21 mW |

Çizelge 2.4 : 2×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.

Çekirdek devre-1 yapıları ile oluşturulmuş 2×1 boyutlu sınıflandırıcı devresinin $(I_{in1}-I_{in2})-I_{out}$ karakteristiğinin benzetimi Şekil 2.20'de verilmiştir.



Şekil 2.20 : ÇD-1 ile gerçeklenen 2×1 -D sınıflandırıcı (I_{in1} - I_{in2})- I_{out} karakteristiği.

Şekil 2.20'de görüldüğü gibi Şekil 2.19'daki gibi tasarlanmış iki boyutlu sınıflandırıcı devresi ile 8 farklı veri kümesi ayırt edilebilmektedir. Eğer sıfır düzlemi de bir sınıflama sonucu olarak kabul edilirse, 9 farklı sınıfa ait veri sınıflandırılmış olur.

2.2.2 Çekirdek devre-2

Giriş-çıkış karakteristiği Şekil 2.1'de verilen devre farklı bir açıdan yaklaşarak tasarlanmıştır. Devrenin blok yapısı Şekil 2.21'de verilmiştir. Gerçekleştirilen devre ilk yapıdan farklı olarak eşik devreleri kullanılarak oluşturulmuştur.



Şekil 2.21 : Akım-modlu ÇD-2'nin blok yapısı.

Çekirdek devre işlevsel diyagramı Şekil 2.22'de gösterildiği biçimde iki adet eşik devresi ve bir fark devresinden oluşmaktadır.



Şekil 2.22 : Akım-modlu ÇD-2 blok diyagramı iç yapısı [55].

Eşik devresinin giriş-çıkış karakteristiği Şekil 2.23'de gösterilmiş ve tanım bağıntısı

$$I_{out} = \begin{cases} I_H & I_1 < I_{in} \\ 0 & di ger & haller de \end{cases}$$
(2.7)

şeklinde verilmiştir.



Şekil 2.23 : Eşik devresinin giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının gerçekleştirilmesi için Şekil 2.24'de verilen blok diyagram kullanılmıştır. Bu yapı da iki eşik devresi ve bir fark alma devresinden oluşmaktadır. İki eşik devresinin çıkış akımlarının farkı alınarak Şekil 2.29'da gösterildiği gibi sınıflandırma bölgesi oluşturulmuştur. Bu bölgenin oluşturulması için eşik devrelerinin I_H akımları eşit seçilmeli ve $I_2 > I_1$ sağlanmalıdır.



Şekil 2.24 : ÇD-2 yapısının gerçekleştirilme blok şeması.

Çekirdek devre-2'nin gerçekleştirilmesi için kullanılan eşik devresi Şekil 2.25 ile gösterilmiştir [56]. Devrede I_1 akımı eşik değeridir. Eğer giriş akımı I_{in} , I_1 akımından büyükse devrenin çıkışındaki akım I_H ve M₃ tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur; giriş akımı I_{in} eşik akımından küçük olduğu zaman ise M₃ tranzistorunun savak gerilimi yaklaşık olarak VDD'ye eşittir. Diğer taraftan M4 tranzistorunun akımı M3 tranzistorunun savak gerilimi ile de kontrol edilmektedir. M₄ ve M₅ (M₄ açıkken M₅ kapalı, M₅ açık iken M₄ kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M₄ ya da M₅ tranzistorundan akmaktadır. Şekil 2.25'deki devre CD4007 CMOS tranzistor entegresi kullanılarak sınanmış ve osiloskop ekranında kullanılan eşik devresinde histerezis karakteristiği olustuğu gözlenmiştir. Devrenin test düzeneği kurulurken akım kaynakları yerine direnç bağlanmıştır. Bu dirençlerin uçlarına uygulanan gerilim ile kontrol akım ve giriş akım değerleri elde edilmiştir. Pozitif geribeslemeli eşik devresini sınamak için I_1 akımı $R_1=20$ k Ω direnç üzerinden $V_{R1}=1.7$ V olacak şekilde uygulanmıştır. Benzer şekilde $R_H=20$ k Ω , $V_H=1.5$ V alınarak uygulanmış ve böylece uygun akım değerleri I_1 = 85 µA ve I_H = 75 µA olacak şekilde sağlanmıştır. Çıkışa bağlanan 100 kΩ'luk direnç ile akımın değişimi osiloskopta gözlenerek akım değeri ölçülmüştür. Kurulan devrenin giriş-çıkış karakteristiği test sonuçları osiloskobun X-Y özelliği kullanılarak Sekil 2.26'da verilmistir. Bu sekilde X ekseni için osiloskop skalası 0.5 V/div, Y ekseni içinse 1 V/div seçilmiştir. Devrenin benzetim sonucu için kontrol akımları $I_1=85 \mu A$ ve $I_H=75 \mu A$ olacak şekilde ayarlanmış, devrenin aynı histerezis davranışını gösterdiği, giriş önce arttırılarak, sonra da azaltılarak uygulandığında elde edilmiş ve sonucu Şekil 2.27'de verilmiştir.



Şekil 2.25 : Pozitif geribeslemeli eşik devresi devre şeması.



Şekil 2.26 : Pozitif geribeslemeli eşik devresi osiloskop çıktısı.



Şekil 2.27 : Pozitif geribeslemeli eşik devresi benzetim karakteristiği.

Şekil 2.25'deki devrede pozitif geribeslemeyi sağlayan M₃ tranzistoru çıkartılarak histerezis karakteristiğinin oluşmamasının sağlandığı hem donanım hem benzetimle sınanmış, gerçekleştirilen yeni eşik devresinin şeması Şekil 2.28'de sunulmuştur. Bu yapıda kullanılan kaynaklar basit akım aynaları olarak devreye uygulanmıştır.



Şekil 2.28 : Eşik devresi devre şeması.

Devrede I_1 akımı eşik değeridir. Eğer giriş akımı I_{in} , I_1 akımından büyükse devrenin çıkışındaki akım I_H ve M₂ tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur. Giriş akımı I_{in} eşik akımından küçük olduğu zaman da M₂ tranzistorunun savak gerilimi yaklaşık olarak V_{DD} 'ye eşittir. Diğer taraftan M₃ tranzistorunun akımı M₂ tranzistorunun savak gerilimi ile de kontrol edilmektedir. M₂ tranzistorunun savak gerilimi yaklaşık V_{SS} olduğu zaman M₃ tranzistoru kesime girerek I_H akımının M₄ üzerinden akması sağlanır. Benzer şekilde M₂ tranzistorunun savak gerilimi yaklaşık V_{DD} olduğu zaman M₃ tranzistoru iletime geçerek I_H akımının M₃ üzerinden akması sağlanır. M₃ ve M₄ (M₃ açıkken M₄ kapalı, M₄ açık iken M₃ kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M₃ ya da M₄ tranzistorundan akmaktadır.

Böylece eşik devreleri kullanılarak gerçekleştirilmiş olan çekirdek devre-2 yapısının giriş-çıkış karakteristiği Şekil 2.29'da verilmiştir.



Şekil 2.29 : ÇD-2 sınıflandırıcı devresi giriş-çıkış karakteristiği.

Giriş-çıkış karakteristiği Şekil 2.29'da verilen devrenin tanım bağıntısı da,

$$I_{out} = \begin{cases} I_H & I_1 < I_{in} < I_2 \\ 0 & di ger haller de \end{cases}$$
(2.8)

şeklinde verilir.

Çekirdek devre-2 yapısının şeması Şekil 2.30'da verilmiştir. Çekirdek devre-2 yapısında kullanılan kontrol ve giriş akımları basit akım aynası kullanılarak devreye uygulanmıştır. Şekil 2.30'da M_1 - M_4 ve M_8 - M_{11} tranzistorları eşik devrelerini oluşturmaktadır. Çıkış akımlarının farkını almak için kullanılan fark devresi ise basit akım aynası (M_6 ve M_7) kullanılarak gerçeklenmiştir. M_{13} , M_{14} ve M_{15} tranzistorları eşik devrelerine aynı I_H akımını uygulamak için kullanılmıştır. Benzer şekilde M_{16} , M_{17} ve M_{18} tranzistorları da girişlere aynı I_{in} akımını uygulamak için kullanılmıştır.

Önerilen çekirdek devre-2 yapısının benzetiminde 0.35 μ m AMS CMOS teknoloji parametreleri kullanılmış ve besleme gerilimleri V_{DD} ve $V_{SS} \pm 1.65$ V olarak alınmıştır. Tranzistor boyutları Çizelge 2.5'de verilmiştir.



Şekil 2.30 : ÇD-2 devre şeması [55].

| Cizelge 2.5 | CD-2 | 'de kullanılan | MOS | tranzistor | boyutları |
|-------------|------|----------------|-----|------------|-----------|
|-------------|------|----------------|-----|------------|-----------|

| MOSFET | W [µm] | L [µm] |
|--|--------|--------|
| M ₁ , M ₂ , M ₃ , M ₄ , M ₅ , M ₈ , M ₉ , M ₁₀ , M ₁₁ , M ₁₂ | 10.5 | 1.05 |
| $M_6, M_7, M_{13}, M_{14}, M_{15}, M_{16}, M_{17}, M_{18}, M_{19}, M_{20}, M_{21}, M_{22}$ | 35.5 | 1.05 |

 $I_1=50 \ \mu\text{A}$ ve $I_H=20 \ \mu\text{A}$ için eşik devresinin benzetim sonuçları Şekil 2.31'de verilmiştir.



Şekil 2.31 : Eşik devresinin giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının benzetim sonuçları I_1 =40 µA, I_2 =80 µA ve I_H =20 µA için Şekil 2.32'de gösterilmiştir. Devrenin güç tüketimi 0.38 mW olarak elde edilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.



Şekil 2.32 : ÇD-2 yapısı giriş-çıkış karakteristiği.

Çekirdek devre-2'nin yayılma gecikmesini veren benzetim için kontrol akım parametreleri I_1 =40 µA, I_2 =80 µA ve I_H =20 µA olarak ayarlanmış ve girişe genliği 70 µA olan bir kare dalga uygulanmıştır. Benzetim sonucunda, gecikme Şekil 2.33'den hesaplanmış ve 4 ns olarak bulunmuştur.



Şekil 2.33 : ÇD-2 yapısı yayılma gecikmesi benzetimi.

Çekirdek devre-2 yapısında kontrol akım parametrelerinin $I_1=10 \ \mu$ A, $I_2=20 \ \mu$ A ve $I_H=2 \ \mu$ A olarak ayarlanması durumunda, yayılma gecikmesi 28 ns olarak bulunmuştur. Bu yayılma gecikmesinin, kontrol akım değerlerinin küçük seçilmesi ile artmasındaki sebep, ÇD-2 yapısında bulunan parazitik kapasitelerin, yüksek kontrol akım değerleriyle daha hızlı dolması ve düşük kontrol akım değerleriyle daha yavaş dolması şeklinde açıklanabilir.

Şekil 2.30'da verilen devrede özellikle M₁₃ ve M₁₄ tranzistorlarının eşleşme problemi sonucu, fark devresi çıkışında çıkış akımının I_H değerinde beklenenden hatalı sonuçlar elde edilebilir. Bu hata aynı zamanda üretim sırasında devredeki tranzistorların kanal boyu ve kanal genişliğinde oluşabilecek sapmalardan da kaynaklanmaktadır. I_H çıkış akımı değerindeki sapmaları görebilmek için devredeki tüm tranzistorların kanal boyu uzunluğu L parametresinde 0.11 µm ve kanal genişliği W parametresinde 0.4 µm'lik sapma durumunda 100 farklı değer göz önüne alınarak Monte Carlo analizi yapılmıştır. Benzetim için kontrol akımları I_H , I_1 ve I_2 0 μ A ile 200 μ A arasında bir çok nominal akım için test edilmiştir. Bu test sonucunda I_H akımı için nominal değerinden maksimum sapma %3.1, I_1 ve I_2 akımları için de maksimum sapma %2.3 olarak gözlenmiştir. Diğer taraftan akım değeri 0 µA olması gereken bölgede $(I_{in}>I_2)$ ise akımın değeri minimum 1.1 µA, maksimum 1.5 µA olmaktadır. Ayrıca örnek olarak $I_H=20 \mu A$, $I_1=40 \mu A$ ve $I_2=80 \mu A$ değerleri için Monte Carlo analizi giriş-çıkış karakteristiği benzetim sonucu Şekil 2.34'de verilmiştir. Sınıflandırma uygulamalarında bu sapma değerleri göz önüne alınarak kontrol akımların seçilmesi durumunda yanlış sınıflandırmaya neden olabilecek sapmalar engellenmis olur.



Şekil 2.34 : ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi.

Tek boyutlu verilerin sınıflandırılması için, Şekil 2.17'de gösterildiği gibi, çekirdek devre-2 yapıları paralel olarak bağlanır ve

$$I_{in1} = I_{in2} = \dots = I_{inn} = I_{in}$$
(2.9)

$$I_1 < I_2 < I_3 < \dots < I_{(2n-1)} < I_{2n}$$
(2.10)

ifadeleri ile verilen koşullar sağlanacak şekilde kontrol akımları seçilir.

Kontrol akımlarının uygun olarak seçildiği durumda 1×1 boyutlu sınıflandırıcının giriş-çıkış karakteristiği Şekil 2.35'de verildiği biçimde elde edilir.



Şekil 2.35 : Tek boyutlu sınıflandırıcının giriş-çıkış karakteristiği.

1×1 boyutlu devrenin benzetimi için 4 adet çekirdek devre-2 yapısı paralel olarak bağlanmıştır. Benzetim sonucu Şekil 2.36'da ve kontrol akımları da Çizelge 2.6'da verilmiştir. Benzetim sonuçlarından görüldüğü gibi "0" bölgesi de dahil beş farklı tipte veri sınıflandırılmaktadır. Devrenin toplam güç tüketimi 0.95 mW'tır.



Şekil 2.36 : ÇD-2 ile gerçeklenen 1×1-D sınıflandırıcı giriş-çıkış benzetimi.

Çizelge 2.6 : CD-2 ile gerçeklenen 1×1 -D kontrol akımları (μ A olarak).

| | ÇD-1 | 1 | (| ÇD-2 | 2 | (| ÇD-3 | 3 | | ÇD-4 | 1 |
|-------|-------|----------|-------|-------|----------|-------|-------|----------|-------|-------|----------|
| I_1 | I_2 | I_{H1} | I_3 | I_4 | I_{H2} | I_5 | I_6 | I_{H3} | I_7 | I_8 | I_{H4} |
| 20 | 30 | 10 | 40 | 50 | 20 | 60 | 70 | 30 | 80 | 90 | 40 |

İki boyutlu sınıflandırıcı devrenin benzetimi için Şekil 2.19'da verilen blok şeması çekirdek devre-2 yapıları kullanılarak kurulmuştur. Benzetimde kullanılan her bir çekirdek devre-2 yapısının kontrol akım ve güç tüketim değerleri Çizelge 2.7'de verilmiştir.

Çizelge 2.7 : ÇD-2 ile gerçeklenen 2×1 -D sınıflandırıcı devresi kontrol akımları (akımlar μ A olarak ifade edilmiştir).

| Çekirdek Devre-1 | I_1 (μ A) | I_2 (μ A) | $I_{H}(\mu A)$ | Güç Tüketimi |
|------------------|------------------|------------------|----------------|--------------|
| Ι | 70 µA | 140 µA | 60 µA | 0.86 mW |
| II | 210 µA | 280 µA | 120 µA | 1.83 mW |
| III | 60 µA | 140 µA | 100 µA | 1.19 mW |
| IV | 280 µA | 360 µA | 80 µA | 1.75 mW |

Çekirdek devre-2 ile gerçekleştirilen 2×1-D sınıflandırıcı yapısının SPICE benzetim sonucu Şekil 2.20'dekine benzer biçimde elde edilir. Çekirdek yapılar paralel bağlanarak daha fazla sayıda verinin sınıflandırılması sağlanabilir.

2.2.2.1 Çekirdek devre-2'nin zayıf evirtim benzetimleri

Son zamanlarda zayıf evirtimde çalışan analog devreler özellikle güç tüketiminin küçük olmasının önemli olduğu uygulamalarda önem kazanmıştır. Bu uygulamaların başında ise pille beslenen, az güç tüketen ve küçük boyutlu tıbbi cihazlar gelmektedir. MOS tranzistorların zayıf evirtimde çalıştırılması sayesinde kurulan devrelerin güç tüketimleri µW'lar mertebesinde olmaktadır.

MOS tranzistorda, V_{GS} geçit-kaynak geriliminin V_T eşik geriliminden büyük olması halinde ($V_{DS} \ge V_{GS} - V_T$, $V_{GS} \ge V_T$), tranzistorun akım gerilim ilişkisi doyma bölgesinde

$$I_D = \frac{k}{2} (V_{GS} - V_T)^2$$
(2.11)

karesel bağıntısı ile verilir. Bu bağıntıdaki *k* MOS tranzistorun geçiş iletkenlik parametresidir. Genellikle, $V_{GS} < V_T$ için I_D savak akımı ihmal edilir. Gerçekte, $V_{GS} < V_T$ için yüzeye yakın bölgelerde bir elektron yoğunluğu bulunduğundan, savak akımı sıfır değildir. Bu bölgeye zayıf evirtim bölgesi denir. Başka bir deyişle, geçit kaynak geriliminin eşik geriliminin üzerinde olduğu bölge kuvvetli evirtim bölgesi, bu gerilimin eşik geriliminin altında olduğu bölge de zayıf evirtim bölgesidir. V_{GS} gerilimi V_T eşik gerilimine yaklaştıkça, MOS tranzistorun I_D - V_{GS} karakteristiği karesel bağımlılıktan üstel bağımlılığa dönüşür. Bir MOS tranzistorun zayıf

$$I_{D} = I_{s} e^{\left(\frac{q(V_{GS} - V_{T} - V_{offset})}{nkT}\right)} \left(1 - e^{-\frac{qV_{DS}}{kT}}\right)$$
(2.12)

bağıntısıyla verilebilir. Bu bağıntıda V_{GS} , V_{DS} büyüklükleri sırası ile geçit-kaynak ve savak-kaynak gerilimlerini vermektedir. I_s akım katsayısı, T sıcaklık, V_{offset} gerilimi -0.1 ile 0.1 V arasında değişen bazı gerilim terimlerinin toplamı, n zayıf evirtim salınım parametresi, k Boltzmann sabiti ve q bir elektronun yüküdür [57]. Çekirdek devre-2'nin oluşturulmasında kullanılan temel blok olan eşik devresinin zayıf evirtimde çalıştırılması için, kontrol akımları I_1 =50 nA ve I_H =20 nA olarak seçilmiş ve eşik devresinin benzetim sonucu Şekil 2.37'de verilmiştir [58].



Şekil 2.37 : Zayıf evirtimde çalışan eşik devresi giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının benzetim sonuçları I_1 =50 nA, I_2 =100 nA ve I_H =20 nA için Şekil 2.38'de verilmiş ve güç tüketimi 0.43 µW olarak elde edilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.



Şekil 2.38 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği.

Benzer şekilde çekirdek devre-2'nin yayılma gecikmesi benzetimi için kontrol akım parametreleri I_1 =50 nA, I_2 =100 nA ve I_H =20 nA olarak ayarlanmış ve girişe genliği 70 nA olan bir kare dalga uygulanmıştır. Benzetim sonucunda, Şekil 2.39'dan hesaplanan giriş-çıkış dalga şekilleri arasındaki yayılma gecikmesi 2.8 µs olarak bulunmuştur.



Şekil 2.39 : Zayıf evirtimde çalışan ÇD-2 için yayılma gecikmesi benzetimi.

Şekil 2.30'da verilen çekirdek devre-2 yapısındaki tranzistorlar zayıf evirtimde çalıştırıldığında M_{13} ve M_{14} tranzistorlarının eşleşme problemi sonucu fark devresi çıkışında, çıkış akımının I_H değerinde beklenenden hatalı sonuçlar elde edilebilir. Bu hata aynı zamanda üretim sırasında devredeki diğer tranzistorların kanal boyu ve kanal genişliğinde oluşabilecek sapmalardan da kaynaklanır. I_H çıkış akımı değerindeki sapmaları görebilmek için devredeki tüm tranzistorların kanal boyu uzunluğu L parametresinde 0.11 µm ve kanal genişliği W parametresinde 0.4 µm'lik sapma durumunda 100 farklı değer göz önüne alınarak Monte Carlo analizi yapılmıştır.

Benzetim için kontrol akımları I_H , I_1 ve I_2 0 nA ile 200 nA arasında bir çok nominal akım için test edilmiştir. Bu test sonucunda I_H akımı için nominal değerinden maksimum sapma %5.5, I_1 ve I_2 akımları için de maksimum sapma %4.2 olarak gözlenmiştir. Diğer taraftan akım değeri 0 nA olması gereken bölgede ($I_{in}>I_2$) ise akımın değeri minimum 2.3 nA, maksimum 3.3 nA olmaktadır.

Ayrıca örnek olarak I_H =20 nA, I_1 =50 nA ve I_2 =100 nA değerleri için Monte Carlo analizi giriş-çıkış karakteristiği benzetim sonucu Şekil 2.40'da verilmiştir. Dolayısıyla sınıflandırma uygulamalarında bu sapma değerleri göz önüne alınarak kontrol akımların seçilmesi durumunda yanlış sınıflandırmaya neden olabilecek sapmalar engellenmiş olur.



Şekil 2.40 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi.

 1×1 boyutlu sınıflandırıcı devrenin benzetimi için dört adet çekirdek devre-2 yapısı paralel olarak Şekil 2.17'deki biçimde bağlanmıştır. Benzetim sonucu Şekil 2.41'de ve çekirdek devre-2 yapılarının kontrol akım ve güç tüketim değerleri de Çizelge 2.8'de verilmiştir. Şekil 2.41'deki benzetim sonuçlarından görüldüğü gibi *x*-ekseni de dahil olmak üzere beş farklı tipte veri sınıflandırılmaktadır; devrenin toplam güç tüketimi 1.62 µW çıkmaktadır.



Şekil 2.41 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 1-D sınıflandırıcı giriş-çıkış karakteristiği.

2×1 boyutlu sınıflandırıcı devrenin benzetimi için Şekil 2.19'daki yapı, çekirdek devre-2 blokları kullanılarak kurulmuştur. Benzetimde kullanılan her çekirdek devre-2'nin kontrol akımları ve güç tüketimleri Çizelge 2.9'da verilmiştir.

Çizelge 2.8 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 1-D sınıflandırıcı yapısı kontrol akımları (akımlar nA).

| | ÇD-1 | 1 | | ÇD-2 | 2 | | ÇD-3 | 3 | | ÇD-4 | 1 |
|-------|-------|----------|-------|-------|----------|-------|-------|----------|-------|-------|----------|
| I_1 | I_2 | I_{H1} | I_3 | I_4 | I_{H2} | I_5 | I_6 | I_{H3} | I_7 | I_8 | I_{H4} |
| 20 | 30 | 10 | 40 | 50 | 20 | 60 | 70 | 30 | 80 | 90 | 40 |

Çizelge 2.9 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 2-D sınıflandırıcı yapısı kontrol akımları (akımlar nA).

| Çekirdek Devre-1 | I_1 (nA) | I_2 (nA) | $I_{H}(\mathbf{nA})$ | Güç Tüketimi |
|------------------|------------|------------|----------------------|--------------|
| Ι | 70 | 140 | 60 | 0.86 µW |
| II | 210 | 280 | 120 | 1.83 μW |
| III | 60 | 140 | 100 | 1.18 μW |
| IV | 280 | 360 | 80 | 1.74 μW |

Şekil 2.19'da gösterildiği gibi I_{in1} ve I_{in2} girişleri iki boyutlu giriş verilerini göstermektedir. İki boyutlu sınıflandırıcının $(I_{in1}-I_{in2})-I_{out}$ giriş-çıkış karakteristiği benzetim sonucu Şekil 2.42'de verilmiştir. Bu şekilden görüldüğü gibi $I_{out}=0$ bölgesi de dahil edildiği zaman 9 farklı veri sınıflandırılabilmektedir.



Şekil 2.42 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçeklenen 2-D sınıflandırıcı $(I_{in1}-I_{in2})-I_{out}$ giriş-çıkış karakteristiği.

2.2.2.2 Çekirdek devre-2'nin ayrık elemanlar ile gerçeklenmesi

Şekil 2.30'daki devre CD4007 CMOS tranzistor entegresi kullanılarak kurulmuştur. V_{DD} =5 V, V_{SS} = -5 V seçilmiştir. Devre içindeki her bir akım kaynağı yerine Şekil 2.43'de gösterildiği gibi seri bir direnç konularak kontrol ve giriş akımları elde edilmiştir. Direnç uçlarına bağlanan gerilim kaynakları ile istenilen kontrol ve giriş akım değerleri sağlanmıştır. Akım kaynakları yerine kullanılan direnç değerleri Çizelge 2.10'da verilmiştir.



Şekil 2.43 : ÇD-2 yapısının testinde kullanılan akım kaynağı modeli.

Çizelge 2.10 : Akım kaynakları yerine kullanılan direnç değerleri.

| I _{in} | I_1 | I_2 | I_H |
|-----------------------------|--------------------------|----------------------------|--------------------------|
| $R_{in}=20 \text{ k}\Omega$ | $R_1=20 \text{ k}\Omega$ | $R_2 = 20 \text{ k}\Omega$ | $R_H=39 \text{ k}\Omega$ |

Çekirdek devre-2'yi sınamak için, V_{R1} = 1.7 V, dolayısıyla I_1 = 85 µA, V_{R2} =2.4 V, dolayısıyla I_2 =120 µA ve I_H = 40 µA olacak şekilde ayarlanmış; çıkışada 100 kΩ'luk direnç bağlanmıştır. Bu direnç, akımın değişimini osiloskopta gözleyerek akımı gerilime dönüştürmek için kullanılmıştır. Kurulan devrenin giriş-çıkış karakteristiğinin test sonuçları osiloskobun X-Y özelliği kullanılarak Şekil 2.44'de verilmiştir; osiloskobun X ekseni skalası 1 V/div, Y ekseni içinse 2 V/div seçilmiştir.



Şekil 2.44 : ÇD-2 giriş-çıkış kararteristiği osiloskop sonucu.

2.2.2.3 Çekirdek devre-2'nin serimi

Devre şeması Şekil 2.30'da verilmiş olan yapının serimi MENTOR programı kullanılarak 0.35 μ m AMS CMOS teknoloji parametreleriyle Şekil 2.45'deki biçimde tasarlanmıştır. Serimin gerçekleştirildiği kırmık alanı 756 μ m² olarak elde edilmiştir.



Şekil 2.45 : ÇD-2'nin serim çizimi.

Çekirdek devre-2 yapısının serim sonrası benzetimi I_1 =40 µA, I_2 =80 µA ve I_H =20 µA için Şekil 2.46'da verilmiştir.



Şekil 2.46 : ÇD-2'nin serim sonrası giriş-çıkış karakteristiği.

Bu şekilden de görüldüğü gibi çekirdek devre-2'nin Şekil 2.32'de verilen SPICE benzetimi ile serim sonrası benzetiminin giriş-çıkış karakteristiği uyuşmaktadır.

2.2.3 Çekirdek devrelerin karşılaştırılması

Çizelge 2.11'de, tezde tasarlanmış olan sınıflandırıcı devrelerin literatürde bulunan benzer özellikler taşıyan devreler ile karşılaştırılması özetlenmiştir. Çizelge 2.11'de verilmiş olan karşılaştırmada [54] numaralı kaynaktaki devrenin, I_1 =40 µA, I_2 =80 µA ve I_H =20 µA seçilmesi durumunda güç tüketimi 0.12 mW ve yayılma gecikmesinin 5 ns seviyesinde elde edildiği görülürken aynı akım değerleri için [71] numaralı kaynakta verilen devrenin güç tüketimi 0.38 mW olurken yayılma gecikmesi 4 ns olmuştur. Çekirdek devre-2 yapısındaki tranzistorların zayıf evirtimde çalışması sayesinde güç tüketimi 0.63 µW ve yayılma gecikmesi de 1.8 µs olarak elde edilmiştir [58].

| Kaynak | Üretim Teknolojisi | Besleme Gerilimi | Güç Tüketimi | Yayılma Gecikmesi | Açıklama |
|--------|-----------------------|---------------------|--------------|----------------------|---|
| [1] | 0.6 μm CMOS | 3.3 V | 14.95 mW | - | - |
| [47] | 0.5 μm CMOS | 3.3 V | 90 μW-160 μW | 20 µs-40 µs | - |
| [59] | 1.2 μm CMOS | 3 V | 25 mW | - | |
| [51] | 0.35 μm CMOS | 5 V | 1.25 mW | - | - |
| [60] | 0.5 μm CMOS | $\pm 2.5 \text{ V}$ | 10 mW | 0.4 µs-0.6 µs | - |
| [61] | 0.35 μm CMOS | $\pm 2 V$ | 2.5 mW | - | - |
| [36] | 0.35 μm CMOS | 5 V | - | - | - |
| [39] | 0.5 μm CMOS | 4 V | 80 nW-840 nW | - | - |
| [54] | 0.35 μm CMOS | ± 1.25 V | 0.12 mW | 5 ns | ÇD-1, <i>I</i> ₁ =40 μA, <i>I</i> ₂ =80 μA <i>I_H</i> =20 μA |
| [58] | 0.35 μm CMOS | ± 1.65 V | 0.63 µW | 1.8 µs | $CD-2$, $I_1=50$ nÅ, $I_2=100$ nA $I_H=40$ nA |
| [55] | 0.35 μm CMOS | ± 1.65 V | 91 µW | 28 ns | ÇD-2, <i>I</i> ₁ =8 μA, <i>I</i> ₂ =16 μA, <i>I</i> _H =4 μA |
| [55] | 0.35 μm CMOS | ± 1.65 V | 0.38 mW | 4 ns | ÇD-2, <i>I</i> ₁ =40 μA, <i>I</i> ₂ =80 μA, <i>I</i> _H =20 μA |

Çizelge 2.11 : Sınıflandırıcı devrelerin karşılaştırılması.

Çekirdek devrelerinin sınıflandırıcı olarak kullanılacağı düşünüldüğünden, öncelikli olarak kullanılacak uygulamada istenilen çalışma değerleri göz önüne alınarak yapı seçilmelidir. Bunun için önerilen devrelerde güç tüketimi, besleme gerilimi ve yayılma gecikmeleri incelenmiştir. Her ne kadar devrenin yayılma gecikmesinden

feragat edilmiş olsa bile, güç tüketimi konusunda bir iyileşme sağlanmıştır. Dolayısıyla sınıflandırma için önem taşıyan özellikler iyileştirildiği gibi, ayrıca sınıflandırma uygulamasına göre bu sakıncalar ve üstünlükler göz önüne alınarak uygun devre de seçilebilir. Ancak her iki çekirdek devre topolojisi de önerilmiş olan uygulama alanlarında kullanılabilir.

Tez çalışmasının devamında çekirdek devre olarak çekirdek devre-2 yapısı seçilmiştir. Gerek uygulamalarda, gerekse algoritma anlatımlarında kullanılan ve şekillerde çekirdek devre olarak gösterilen blok çekirdek devre-2 yapısıdır.

2.3 Çekirdek Devreler ile Elde Edilebilen Bölgeler

Tasarlanmış olan iki adet çekirdek devre yapısı da uygun kontrol akımları seçilerek sınıflandırıcı yapısı olarak kullanılabilir. Diğer taraftan, bu aşamaya kadar gerçekleştirilen çekirdek devreler kullanılarak geliştirilen sınıflandırıcı devreleri, Şekil 2.6 ile gösterildiği gibi, dikdörtgen ızgaralarla (DI) ayrıştırılabilen verileri sınıflama yeteneğine sahiptir.

Ayrıca çekirdek devre yapıları kullanılarak Şekil 2.5'deki yapı üç farklı şekilde genelleştirilebilir:

- a) *m* tane çekirdek bloğa x_1 değişkeni ve *n* tane çekirdek bloğa x_2 değişkeni uygulandıktan sonra tüm çıkışlar paralel bağlanır. Bu durumda elde edilen sınıflandırma yapısı iki boyutlu (*m*+1)×(*n*+1) tane farklı veriyi sınıflandırır. Bu şekildeki bir topolojinin veri tanımasını yapabilmesi için çıkışının ne değerler alması gerektiği Şekil 2.47'de gösterilmiştir. Blokların içinde yazılı değerler sınıflandırılacak verinin ait olduğu sınıfın seviyesini (temel çekirdek yapısı için *k* parametresi) göstermektedir.
- b) n tane çekirdek bloğa farklı giriş değişkenleri uygulandıktan sonra çıkışlarını paralel bağlayarak $n \times 1$ -boyutlu bir sınıflandırma yapısı elde edilebilir.
- c) (a) ve (b) maddelerindeki durumların kombinasyonunu kullanarak ve çeşitli çıkışları da aralarında gruplayarak çıkışı da çok boyutlu (bir vektör olan), farklı karar bölgeleri elde edilebilir.
| | | 1 | 2 | | n | |
|---|---------------|--------------------|----------------------------|-----|---------------------------------|--|
| | | $2k+(m+1)\Delta k$ | $2(k+m\Delta k)+2\Delta k$ | *** | $n(k+m\Delta k)+n\Delta k$ | |
| 1 | k+Δk | 2k+(m+2)∆k | 3k+(2m+3)∆k | | (n+1)k+ [n(m+1)+1]∆k | |
| | | | | | | |
| 2 | $k+2\Delta k$ | 2k+(m+3)∆k | 3k+(2m+4)∆k | | (n+1)k+ $[n(m+1)+2]\Delta k$ | |
| | * | | | | | |
| m | k+m∆k | 2k+(2m+1)∆k | 3k+(3m+2)∆k | | (n+1)k+ $[n(m+1)+m]\Delta k$ | |
| | | | | | | |

Şekil 2.47 : 2×1 boyutlu sınıflandırıcının farklı çıkış seviyeleri için genel hali.

Çalışmanın devamında yukarıda bahsedilen üç farklı durum göz önüne alınarak sınıflandırıcı çekirdek devre blokları çeşitli uygulama alanlarında kullanılmış ve benzetimleri yapılmıştır.

3. EĞİK IZGARALI SINIFLANDIRICILAR

3.1 Bölgelerin Oluşturulması

Literatürde, "perseptron" türü yapay sinir ağlarının sınıflandırabileceği verilerin ne türden olacağı araştırılırken bunların lineer olarak ayrılabilen (bölgelerin bir hiperdüzlemle ayrıştırılabildiği özel durum) veriler olduğu gösterilmiştir [19,62]. Ancak bir çok veri türünün lineer olarak ayrıştırılamadığı bilindiğinden, çalışmanın bu kısmında daha genel dağılımı olan verilerin sınıflandırıcı devreler ile nasıl ayırt edilebileceği ele alınmıştır. Lineer olarak ayrıştırılamayan verilerin sınıflandırılması için önerilmiş olan bir yöntem ise verilerin Şekil 3.1'de gösterildiği biçimde taranmış bölgeler ile ayrıştırılmasına dayanmaktadır [63].



Şekil 3.1 : Lineer olarak sınıflandırılamayan veri kümesi.

Anlatım kolaylığı sağlaması açısından, iki boyutlu iki farklı sınıftan oluşan ve lineer olarak ayrıştırılamayan Şekil 3.1'deki gibi bir veri kümesi ele alalım; bu türden bir veri yapısı için bölgeleri ayıran eğrilerin bir hiperdüzlem (2-D durumunda eğimi sonlu ve sıfırdan farklı bir doğru) olması gerekir. Eşdeğer bir başka yaklaşım ise çekirdek devrenin girişine sınıflandırılacak verilerin (x_1 , x_2) belli katsayılarla ile çarpılmış (ağırlıklı) toplamını (lineer kombinasyonu) uygulamaktır. Ağırlık katsayıları ve çekirdek devrenin kontrol akımları uygun bir sınıflandırma algoritması ile belirlenerek, Şekil 3.3'de gösterilen bölgelerin oluşması sağlanır.



Şekil 3.2 : Eğik Izgaralı veri sınıflandırıcısı.

Şekil 3.2'de gösterilen yapıda I_{in} girişinin elde edilmesinde kullanılan x_1 girişinin belli bir katsayı ile çarpılmasını sağlayan Çarpan Devresi (ÇAD) yapısı Şekil 3.4'de ve tanım bağıntısı da,

 $y = xw \tag{3.1}$

olarak verilir. Bu bağıntıdaki w bir sabit olup, giriş büyüklüğünü sadece ölçeklemektedir. Gerçekleştirilmiş olan çekirdek devre yapıları akım-modlu olduğu için (3.1) ifadesindeki xw terimi de akım modludur. Buradaki x terimi sınıflandırılacak verinin özelliklerini belirleyen bir değişkeni göstermektedir.



Şekil 3.3 : Dikdörtgen Izgara olmayan veri bölgeleri.



Şekil 3.4 : ÇAD yapısı blok diyagramı sembolik gösterimi.

Şekil 3.2'deki bloğun ÇAD yapıları kullanılarak elde edilen blok diyagramı Şekil 3.5'de gösterilmiştir.



Şekil 3.5 : Şekil 3.2'deki bloğun iç yapısı.

Şekil 3.2'deki yapılar paralel bağlanarak Şekil 3.6'daki eğik ızgaralı sınıflandırıcı elde edilir; eğik ızgaralı sınıflandırıcının ayırdığı iki boyutlu veri bölgelerinin gösterilimi Şekil 3.7'de, bu bölgelerdeki verileri işleyen sınıflandırıcı çıkışının üç boyutlu I_{out} -(x_1 - x_2) karakteristiği ise Şekil 3.8'de verilmiştir. Buradaki üç boyutlu şekilde yükseklik değeri çekirdek devrelerin I_H parametresi ile ayarlanmaktadır. Diğer bir ifade ile, sınıflandırılacak veriler bu yüksekliklerin aldığı değerlere göre saptanmaktadır. Dolayısıyla tipik birer eğri örneği olan Şekil 3.7'de ve Şekil 3.8'de her renk farklı bir veri sınıfına karşı gelmektedir.



Şekil 3.6 : Paralel bağlanmış sınıflandırıcı devresi.



Şekil 3.2'deki yapılardan ikiden fazla sayıda paralel yapı bağlanarak ya da girişlerine 2 veri değişkeninden daha fazlasının lineer kombinezonu uygulanarak, daha farklı ve çok sayıda bölgeler de oluşturulabilir. ÇAD devresinin donanımsal gerçeklenmesi Bölüm 3.2'de incelenmiştir.



Şekil 3.8 : Eğik ızgaralı sınıflandırıcı çıkışının 3-D I_{out} - (x_1-x_2) karakteristiği.

Şekil 3.9'da gösterilen, lineer olarak sınıflandırılamayan ve de dikdörtgen ızgara bölgeler ile ayrıştırılamayan veriler, ÇAD ve ÇD yapıları kullanılarak sınıflandırılabilmektedir. Bu verilerin sınıflandırılması biçimsel olarak aşama aşama Şekil 3.9 a-d'de gösterilmiştir.



Şekil 3.9 : Verilerin ÇAD ve ÇD kullanılarak sınıflandırılması.

Dolayısıyla bu çeşit bir veri kümesinin sınıflandırılabilmesi için uygun doğruların, başka bir deyişle ÇAD devresinin parametreleri olan w_i 'ler (ağırlık katsayıları) ve ÇD parametreleri I_1 , I_2 (kontrol akımları) değerlerinin bulunması gerekmektedir. Şekil 3.9'da kesikli çizgiler ile gösterilen bu doğrulara çift eşik doğruları denilmektedir.

Bu çalışmada çift eşik doğrularının elde edilmesi için Fisher'in lineer diskriminant analiz yönteminden faydalanılmıştır [64]. Şekil 3.8'de gösterilen yüksekliklerin sayısal değerleri ÇD parametresi I_H ile ayarlanmaktadır. Dolayısıyla her farklı yükseklik değeri ayrı bir sınıfı göstermektedir. Şekil 3.6'da gösterilmiş bir sınıflandırıcı yapısı uygulaması Bölüm 5.3'de verilmiştir.

3.2 ÇAD Devresi ve Benzetimleri

Çarpan devresinin blok diyagramı Şekil 3.4'te gösterilmiştir. Tanım bağıntısı (3.1) ifadesi ile verilen bu yapı, uygulamalarda kullanım kolaylığı sağlanması için giriş değişkeni gerilim, çıkış değişkeni akım olarak tasarlanmıştır. Dolayısıyla gerçekleştirilmiş olan devre blok şeması olarak Şekil 3.10'da gösterilmiştir.



Şekil 3.10 : ÇAD devresi blok şeması.

Bu devrenin tanım bağıntıları da

$$I_{out}^{+} = \frac{V_{in}}{R_1} \frac{R_2}{R_3}$$
(3.2)

$$I_{out}^{-} = -\frac{V_{in}}{R_1} \frac{R_2}{R_3}$$
(3.3)

ifadeleri ile verilmiştir.

Şekil 3.10'da kullanılan R_1 direnci, V_{in} giriş geriliminin akıma dönüştürülmesini sağlamak amacıyla kullanılmıştır. Ayrıca, aynı direnç uygulamalarda karşılaşılacak giriş verilerinin uygun değerlere normalizasyonu için de kullanılmaktadır. Diğer taraftan R_2/R_3 oranı ise giriş geriliminin, dolayısıyla akımının belli bir çarpan ile çıkışa aktarılmasını sağlamaktadır. Burada kullanılan elemanların direnç olması gerek çarpım gerekse normalizasyon işlemi sırasında esneklik sağlamaktadır. Ayrıca (3.2) ve (3.3) ifadelerinden görüldüğü gibi çıkış değişkeni, akım boyutunda oluşturulduğundan çekirdek devre yapılarına uygulanabilmektedir.

ÇAD devresinin devre şeması iki adet İkinci Kuşak Akım Taşıyıcı (CCII) elemanı kullanılarak gerçekleştirilmiştir [65]. Bu CCII yapılarında z ucu çıkış kabul edilmek üzere, tanım bağıntısı aşağıdaki gibi

$$\begin{bmatrix} V_x \\ I_y \\ I_z \end{bmatrix} = \begin{bmatrix} 0 & \beta & 0 \\ 0 & 0 & 0 \\ \alpha & 0 & 0 \end{bmatrix} \begin{bmatrix} I_x \\ V_y \\ V_z \end{bmatrix}$$
(3.4)

matris biçiminde ifade edilen CCII yapısının blok diyagramı Şekil 3.11'de ve devre şeması da Şekil 3.12'de gösterilmiştir.



Şekil 3.11 : CCII devresi blok diyagramı.



Şekil 3.12 : CCII devre şeması [65].

Benzer şekilde Çift Çıkışlı İkinci Kuşak Akım Taşıyıcı (DO-CCII) elemanının tanım bağıntısı aşağıda verilmiştir.

$$\begin{bmatrix} V_x \\ I_y \\ I_{z+} \\ I_{z-} \end{bmatrix} = \begin{bmatrix} 0 & \beta & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \alpha_1 & 0 & 0 & 0 \\ -\alpha_2 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_x \\ V_y \\ V_{z+} \\ V_{z-} \end{bmatrix}$$
(3.5)

Aynı yapının blok diyagramı Şekil 3.13'de ve devre şeması Şekil 3.14'de gösterilmiştir.



Şekil 3.13 : DO-CCII devre blok diyagramı.



Şekil 3.14 : DO-CCII devre şeması.

DO-CCII devresinin benzetiminde 0.35 µm AMS CMOS teknoloji parametreleri ve besleme gerilimleri de V_{DD} ve $V_{SS} \pm 1.65$ V olarak kullanılmıştır. Kutuplama akımının 18 µA olması için V_B gerilimi 0.8 V olarak seçilmiştir. Şekil 3.14'de verilen devre için V_x geriliminin V_y ile değişim karakteristiği Şekil 3.15'de verilmiştir.



Şekil 3.15 : DO-CCII devresi için V_x 'in V_y ile değişim karakteristiği.

İdeal olarak DO-CCII devresinde $\alpha_1 = \alpha_2 = 1 \text{ ve } \beta = 1$ 'dir. Ancak devredeki idealsizliklerden dolayı $\beta = 1 - \varepsilon_v$, $\alpha_1 = 1 - \varepsilon_{i1}$ ve $\alpha_2 = 1 - \varepsilon_{i2}$ değerleriyle verilebilir. Buradaki ε_v ve ε_{ik} (k = 1, 2; $|\varepsilon_v| <<1$; $|\varepsilon_{ik}| <<1$) sayıları sırası ile gerilim ve akım

izleme hatalarını vermektedir. Gerilim izleme hatasının incelenmesi için $|\varepsilon_v|$ 'nin V_y ile değişim karakteristiği Şekil 3.16'da verilmiştir. Şekilden görüldüğü gibi gerilim izleme hatası maksimum % 2.4 olmaktadır. ε_v 'nin değişimi 0 noktasında tanımsız olmasından dolayı bu nokta civarı için karakteristik çizdirilememiştir.



Şekil 3.16 : DO-CCII devresi için $|\varepsilon_v|$ 'nin V_y ile değişim karakteristiği.

Ayrıca aynı devrede I_{z+} ve I_{z-} akımlarının I_x akımını takip etme başarımlarının gösterildiği karakteristik ise Şekil 3.17'de verilmiştir.



Şekil 3.17 : DO-CCII için I_{z+} ve I_{z-} akımlarının I_x ile değişim karakteristiği.

Benzer şekilde akım izleme hatalarının incelenmesi için $|\varepsilon_{i1}|$ ve $|\varepsilon_{i2}|$ 'nin I_x ile değişim karakteristikleri sırası ile Şekil 3.18'de ve Şekil 3.19'da verilmiştir.



Şekil 3.18 : DO-CCII devresi için $|\varepsilon_{i1}|$ 'nin I_x ile değişim karakteristiği.



Şekil 3.19 : DO-CCII devresi için $|\varepsilon_{i2}|$ 'nin I_x ile değişim karakteristiği.

Karakteristiklerden görüldüğü gibi giriş akımı -200 μ A ile 200 μ A aralığında değiştirildiği zaman, maksimum akım izleme hatası $|\varepsilon_{i1}| = \% 0.13$ ve $|\varepsilon_{i2}| = \% 1.6$ 'dır.

Çalışmada sınıflandırma için kullanılacak temel yapılardan biri olan ÇAD devresi, Şekil 3.11 ve Şekil 3.13'deki yapılar kullanılarak Şekil 3.20'de gösterildiği biçimde oluşturulmuştur. Bu yapının akım gerilim bağıntısı aşağıdaki ifadeler ile verilir:

$$I_{z+} = \frac{V_y}{R_1} \frac{R_2}{R_3}$$
(3.6)

$$I_{z-} = -\frac{V_y}{R_1} \frac{R_2}{R_3}$$
(3.7)

Çıkış değişkeni olarak (3.6) ve (3.7) ifadesindeki I_{z+} ve I_{z-} akımları kullanılmış ve uygulamalarda (3.2) ifadesinde gösterildiği şekilde I_{out} olarak alınmıştır.



Şekil 3.20 : ÇAD yapısının CCII ve DO-CCII yapıları ile gerçeklemesi.

Şekil 3.20'de verilen devrenin benzetimi için R_1 =25 k Ω , R_2 =25 k Ω ve R_3 =5 k Ω seçilerek V_y ucuna -0.8 V ile 0.8 V arasında değişen gerilim uygulanmış ve R_1 direnci üzerindeki akım giriş akımı olarak alınmıştır. Dolayısıyla giriş akımı -32 µA ile 32 µA aralığında değiştirilmiştir. $R_2 / R_3 = 5$ için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği benzetim sonucu Şekil 3.21'de verilmiştir.



Şekil 3.21 : ÇAD devresi için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği ($R_2 / R_3 = 5$).

ÇAD devresinde R_2/R_3 oranlarının farklı seçilmesi durumunda giriş-çıkış karakteristiği için gerçekleştirilmiş olan benzetim sonucu Şekil 3.22'de verilmiştir. Bu benzetimde $R_2=25$ k Ω alınmış ve R_3 değerleri şekildeki *k* değerlerini sağlayacak biçimde seçilmiştir.



Şekil 3.22 : Çeşitli k= R_2 / R_3 değerleri için ÇAD devresi V_y - I_{z+} karakteristiği.

Önerilen ÇAD devresinin benzetiminde kullanılan tranzistor boyutları Çizelge 3.1'de verilmiştir.

| MOSFET | W [µm] | L [µm] |
|---|--------|--------|
| M ₁ , M ₄ , M ₅ | 21 | 1.05 |
| M_6, M_7 | 6.3 | 1.05 |
| M ₂ , M ₃ , M ₁₀ , M ₁₁ , M ₁₂ | 42.7 | 1.05 |
| M ₈ , M ₉ , M ₁₃ , M ₁₄ , M ₁₅ | 11.2 | 1.05 |

Çizelge 3.1 : ÇAD devresi MOS tranzistorların boyutları.

4. SINIFLANDIRMA ALGORİTMALARI VE DEVRELERE UYGULANMASI

4.1 Fisher Tabanlı Algoritma ile Çift Eşik Doğrularının Bulunmalarına Genel Bakış

Sınıflandırılacak veriler 2 boyutlu uzayda, c_1 ve c_2 sınıflarına ait olmak üzere, Şekil 4.1'de gösterildiği gibi verilmiş olsun. Fisher'in lineer diskriminant analizi geliştirilerek bu verilerin, \vec{v} vektörü ile doğrultusu verilen ve orijinden geçen bir doğru üzerine izdüşürülmelerinden yararlanarak ayrıştırılmasına çalışılacaktır [64].



Şekil 4.1 : Lineer olarak sınıflandırılamayan örnek veri kümesi.

Bu izdüşüm doğrusu üzerine izdüşürülen verilerin merkeze (orijine) uzaklıkları göz önüne alınarak Şekil 4.2'de gösterildiği gibi histogramları çizilir. Şekil 4.2'de μ_{pl} ve μ_{p2} verilerin orijine olan uzaklıklarının ortalamasını, σ_1 ve σ_2 ise standart sapmalarını göstermektedir.



Şekil 4.2 : Histogram karakteristiği.

Histogram eğrilerinden faydalanılarak iki eşik doğrusu elde edilir. Ancak izdüşüm doğrusu, verilerin en iyi ayrıştırılmasının sağlanması açısından:

a) μ_{p1} ve μ_{p2} birbirinden maksimum uzaklıkta

b) σ_1 ve σ_2 de olabildiğince küçük

koşullarını sağlayacak şekilde seçilmelidir.

4.1.1 Fisher tabanlı algoritma ile çift eşik doğrularının bulunması

 \vec{x} vektörünü

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$
(4.1)

şeklinde verilen iki boyutlu ve iki farklı sınıftan oluşan veri kümesinin bir elemanı olarak alalım; \vec{x} örneklerinin, \vec{v} birim vektörü ile temsil edilen izdüşüm doğrusuna izdüşümleri $\vec{v}^T \vec{x}$ iç çarpımıyla gösterilir. μ_{p1} ve μ_{p2} sırası ile birinci ve ikinci sınıfa ait verilerin orijine olan ortalama izdüşüm uzaklıkları, $\vec{\mu}_1$ ve $\vec{\mu}_2$ ise birinci sınıf ve ikinci sınıf veri değerlerinin aritmetik ortalamalarından oluşan vektörler olsun. Bu durumda

$$\mu_{p1} = \vec{v}^T \vec{\mu}_1 \tag{4.2}$$

$$\boldsymbol{\mu}_{p2} = \vec{\boldsymbol{v}}^T \vec{\boldsymbol{\mu}}_2 \tag{4.3}$$

bağıntıları geçerlidir [64].

Ayrıca

$$y = \vec{v}^T \vec{x} \tag{4.4}$$

ifadesi orijine olan izdüşüm uzaklıklarını gösterdiğine göre, birinci ve ikinci sınıf verileri için sırası ile saçılmalar (standart sapmanın eşdeğeri):

$$s_{p1}^{2} = \sum_{y_{i} \in C_{1}} (y_{i} - \mu_{p1})^{2}$$
(4.5)

$$s_{p2}^{2} = \sum_{y_{i} \in C_{2}} (y_{i} - \mu_{p2})^{2}$$
(4.6)

ifadeleri ile verilir.

Bu ifadelerde verilen y_i 'ler eğitim kümesindeki her bir elemanın orijine olan izdüşüm uzaklığıdır.

Sınıflandırma işleminin en az hata ile gerçekleştirilebilmesi için,

$$J(\vec{v}) = \frac{(\mu_{p1} - \mu_{p2})^2}{(s_{p1} + s_{p2})^2}$$
(4.7)

ifadesinin maksimum kılınması gerekir; dolayısıyla öyle bir \vec{v} bulunmalıdır ki $J(\vec{v})$ ifadesi maksimum olsun.

Birinci ve ikinci sınıf verilerin sınıf içi saçılma matrisleri:

$$S_{1} = \sum_{\vec{x} \in C_{1}} (\vec{x} - \vec{\mu}_{1}) (\vec{x} - \vec{\mu}_{1})^{T}$$
(4.8)

$$S_{2} = \sum_{\vec{x} \in C_{2}} (\vec{x} - \vec{\mu}_{2}) (\vec{x} - \vec{\mu}_{2})^{T}$$
(4.9)

ifadeleri ile tanımlanır ve S_W matrisi

$$S_w = S_1 + S_2$$
 (4.10)

olarak yazılırsa, (4.5) ifadesi ile verilen saçılma terimleri:

$$s_{p1}^{2} = \sum_{y_{i} \in C_{1}} (y_{i} - \mu_{p1})^{2} = \sum_{y_{i} \in C_{1}} (\vec{v}^{T} \vec{x} - \vec{v}^{T} \vec{\mu}_{1})^{2} = \sum_{y_{i} \in C_{1}} (\vec{v}^{T} (\vec{x} - \vec{\mu}_{1}))^{T} (\vec{v}^{T} (\vec{x} - \vec{\mu}_{1}))$$
(4.11)

$$s_{p1}^{2} = \sum_{y_{i} \in C_{1}} \left(\left(\vec{x} - \vec{\mu}_{1} \right)^{T} \vec{v} \right)^{T} \left(\left(\vec{x} - \vec{\mu}_{1} \right)^{T} \vec{v} \right) = \sum_{y_{i} \in C_{1}} \vec{v}^{T} \left(\vec{x} - \vec{\mu}_{1} \right) \left(\vec{x} - \vec{\mu}_{1} \right)^{T} \vec{v} = \vec{v}^{T} S_{1} \vec{v}$$
(4.12)

biçiminde ifade edilebilirler.

Benzer şekilde (4.6) ifadesi de şöyle yazılabilir:

$$s_{p2}^2 = \vec{v}^T S_2 \vec{v}$$
 (4.13)

Bu durumda saçılma matrisleri toplamı

$$s_{p1}^{2} + s_{p2}^{2} = \vec{v}^{T} S_{1} \vec{v} + \vec{v}^{T} S_{2} \vec{v} = \vec{v}^{T} S_{w} \vec{v}$$
(4.14)

olur.

İki sınıfın ortalamaları arasındaki ayrılabilirliğin bir ölçütü olan sınıflar arası saçılma matrisi

$$S_{B} = (\vec{\mu}_{1} - \vec{\mu}_{2})(\vec{\mu}_{1} - \vec{\mu}_{2})^{T}$$
(4.15)

ifadesi ile verilir.

Diğer taraftan (4.7) ifadesinin pay terimi aşağıdaki şekilde yazılır:

$$(\mu_{p1} - \mu_{p2})^2 = (\vec{v}^T \vec{\mu}_1 - \vec{v}^T \vec{\mu}_2)^2 = \vec{v}^T (\vec{\mu}_1 - \vec{\mu}_2) (\vec{\mu}_1 - \vec{\mu}_2)^T \vec{v} = \vec{v}^T S_B \vec{v}$$
(4.16)

Dolayısıyla (4.14) ve (4.16) eşitlikleri kullanılarak oluşturulmuş

$$J(\vec{v}) = \frac{(\mu_{p1} - \mu_{p2})^2}{(S_{p1}^2 + S_{p2}^2)} = \frac{\vec{v}^T S_B \vec{v}}{\vec{v}^T S_w \vec{v}}$$
(4.17)

ifadesinden görüldüğü gibi sınıflandırma işleminin en uygun şekilde yapılması sınıflar arası saçılma matrisine (S_B) ve sınıf içi saçılma matrisine (S_W) bağlı çıkmaktadır.

Böylece problem $J(\vec{v})$ ölçütünü maksimum kılan \vec{v} vektörünü, yani izdüşüm doğrusunu bulmak şeklinde ifade edilebilir hale gelir. Bunun için ölçütün gradyenini sıfıra eşitleyen

$$\frac{\partial}{\partial v}J(\vec{v}) = \frac{\left(\frac{\partial}{\partial v}\vec{v}^T S_B \vec{v}\right)\vec{v}^T S_w \vec{v} - \left(\frac{\partial}{\partial v}\vec{v}^T S_w \vec{v}\right)\vec{v}^T S_B \vec{v}}{\left(\vec{v}^T S_w \vec{v}\right)^2} = 0$$
(4.18)

ifadesi kullanılarak

$$\frac{\partial}{\partial v}J(\vec{v}) = \frac{(2S_B\vec{v})\vec{v}^T S_w\vec{v} - (2^T S_w\vec{v})\vec{v}^T S_B\vec{v}}{\left(\vec{v}^T S_w\vec{v}\right)^2} = 0$$
(4.19)

$$\vec{v}^{T} S_{w} \vec{v} (S_{B} \vec{v}) - \vec{v}^{T} S_{B} \vec{v} (S_{w} \vec{v}) = 0$$
(4.20)

$$\frac{\vec{v}^{T} S_{w} \vec{v} (S_{B} \vec{v})}{\vec{v}^{T} S_{w} \vec{v}} - \frac{\vec{v}^{T} S_{B} \vec{v} (S_{w} \vec{v})}{\vec{v}^{T} S_{w} \vec{v}} = 0$$
(4.21)

ara işlemlerinden sonra

$$(S_B\vec{v}) - \frac{\vec{v}^T S_B \vec{v} (S_w \vec{v})}{\vec{v}^T S_w \vec{v}} = (S_B \vec{v}) - \lambda (S_w \vec{v}) = 0$$
(4.22)

eşitliği elde edilir.

(4.22) ifadesindeki denklemin çözümü ise genelleştirilmiş özdeğer probleminden başka bir şey değildir. Bu özdeğer problemi

$$S_B \vec{v} = \lambda S_w \vec{v} \tag{4.23}$$

denklemi ile verilir.

Bu denklemin çözümünden bulunacak olan maksimum değere sahip özdeğere karşılık gelen \vec{v} vektörü verilerin izdüşürüleceği doğrunun eğimini vermektedir [66]. Bu birim vektör Şekil 4.3'de gösterilen izdüşüm doğrusu ile aynı yöndedir [64].



Şekil 4.3 : Çift eşik doğrularının gösterilimi.

Şekil 4.3'de açık renkli olarak çizilmiş histogram eğrisi birinci sınıfa, koyu renkli olarak çizilmiş histogram eğrisi ikinci sınıfa ait veriler olarak alalım. Bu durumda \vec{v} vektörü bulunduktan sonra histogram eğrileri kullanılarak *a* ve *b* noktaları bulunur. Burada koordinatları (x_{a1} , x_{a2}) olan *a* noktası, izdüşüm doğrusu üzerinde ikinci sınıfa ait olabilecek son noktadır; başka bir deyişle, izdüşüm doğrusu üzerinde ikinci sınıfa ait verilerin orijine uzaklığı en küçük olan noktadır. Benzer şekilde koordinatları (x_{b1} , x_{b2}) olan *b* noktası ise birinci sınıfa ait verilerde orijine olan izdüşüm uzaklığı maksimum olan değerdir. Bu durumda verilerin ayırt edilmesini sağlayacak doğruların (hiperdüzlemlerin) denklemleri ise aşağıdaki ifadeler ile verilir:

$$\vec{v}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \vec{v}^T \begin{pmatrix} x_{a1} \\ x_{a2} \end{pmatrix} = 0$$
(4.24)

$$\vec{v}^{T}\begin{pmatrix}x_{1}\\x_{2}\end{pmatrix} - \vec{v}^{T}\begin{pmatrix}x_{b1}\\x_{b2}\end{pmatrix} = 0$$
(4.25)

Bu denklemler izdüşüm doğrusu üzerinde bulunan *a* ve *b* noktalarına dik çıkan Şekil 4.3'de kesik çizgiler ile gösterilmiş doğru denklemleridir. Bu doğru denklemleri bulunduktan sonra *a* ile *b* noktaları arasında kalan veri kümesi için aynı algoritma tekrar uygulanır ve benzer doğrular arada kalan veri kümesi için tekrar hesaplanır. Verileri kesin olarak ayrıştıran doğrular bulunduktan sonra, yöntem durdurulur. Bu (4.24) ve (4.25) ifadeleri ile bulunmuş olan denklemler çift eşik doğrularıdır.

4.1.2 Fisher tabanlı algoritma ile çift eşik doğrularının bulunmasında genelleştirilmiş hal

Çift eşik doğrularının çizilmesi için anlatım kolaylığı açısından iki boyutlu ve iki sınıftan oluşan veri kümeleri kullanıldı. Çift eşik doğrularının çizilmesinde genelleştirilmiş durum, $d \ge c$ koşulu altında d boyutlu c adet farklı sınıf olsun. c=2 olması durumu için Altbölüm 4.1.1'de anlatılan yöntem hiç bir değişikliğe uğramaksızın uygulanır; aksi halde (c-1) adet birim izdüşüm vektörü bulunur ve bunlarla

$$V = \begin{bmatrix} \vec{v}_1 | \vec{v}_2 | ... | \vec{v}_{c-1} \end{bmatrix}$$
(4.26)

matrisi oluşturulur.

Bu durumda izdüşüm vektörlerinin bulunması için öncelikle genelleştirilmiş haldeki sınıf içi saçılma matrisi (S_W) ve sınıflar arası saçılma matrisi (S_B) elde edilmelidir. n_i , *i*. sınıfa ait eğitim kümesindeki veri sayısı ise, S_W ve S_i matrisleri

$$S_w = \sum_{i=1}^c S_i \tag{4.27}$$

$$S_{i} = \sum_{x \in C_{i}} (\vec{x} - \vec{\mu}_{i}) (\vec{x} - \vec{\mu}_{i})^{T}$$
(4.28)

ifadeleri ile, $\vec{\mu}_i$ ortalaması da

$$\vec{\mu}_i = \frac{1}{n_i} \sum_{x \in C_i} \vec{x}$$
(4.29)

eşitliği ile verilir. Dolayısıyla

$$S_{w} = \sum_{i=1}^{c} \sum_{C_{i}} (\vec{x} - \vec{\mu}_{i}) (\vec{x} - \vec{\mu}_{i})^{T}$$
(4.30)

olur.

(4.28) ifadesinde bulunan $\vec{\mu}_i$, *i*. sınıfa ait verilerin ortalaması, *n* eğitim kümesindeki toplam veri sayısıdır. Tüm verilerin ortalaması olan $\vec{\mu}$ vektörü,

$$\vec{\mu} = \frac{1}{n} \sum_{\forall x} \vec{x}$$
(4.31)

ifadesi ile verilir.

Sınıflar arası saçılma matrisi S_B aşağıdaki eşitlikteki gibidir:

$$S_B = \sum_{i=1}^{c} n_i (\vec{\mu}_i - \vec{\mu}) (\vec{\mu}_i - \vec{\mu})^T$$
(4.32)

 $\vec{\mu}_{pi}$, *i*. sınıfa ait verilerin ortalama izdüşüm uzaklığından oluşan vektör, $\vec{\mu}_p$ ise $\vec{\mu}_{pi}$ 'lerin ortalamasıdır. Dolayısıyla

$$\vec{\mu}_p = \frac{1}{n} \sum_{\forall y} y \tag{4.33}$$

olarak yazılabilir.

Sınıf içi saçılma matrisi (S_{PW}) ve izdüşürülmüş sınıflar arası saçılma matrisi (S_{PB}) sırası ile

$$S_{PW} = \sum_{i=1}^{c} \sum_{y \in C_{i}} (\vec{y} - \vec{\mu}_{pi}) (\vec{y} - \vec{\mu}_{pi})^{T}$$
(4.34)

olmak üzere,

$$S_{PW} = V^T S_W V \tag{4.35}$$

ile, ve

$$S_{PB} = \sum_{i=1}^{c} n_i (\vec{\mu}_{pi} - \vec{\mu}_p) (\vec{\mu}_{pi} - \vec{\mu}_p)^T$$
(4.36)

de,

$$S_{PB} = V^T S_B V \tag{4.37}$$

ifadesiyle verilir.

Sınıflandırma işleminin optimum şekilde yapılması için, izdüşürülmüş sınıflar arası saçılma matrisi (S_{PB}) ile izdüşürülmüş sınıf içi saçılma matrisi (S_{PW}) skaler olmadıklarından, pay ve payda terimlerinin determinant olduğu

$$J(V) = \frac{\det(V^T S_B V)}{\det(V^T S_w V)}$$
(4.38)

biçiminde J(V) ölçütü ile tanımlanır. Bu ölçüt maksimize edilerek en iyi ayıran hiperdüzlemler bulunur [67]. (4.38) ifadesinin maksimum kılınması, aşağıda verilen

$$(S_B - \lambda_i S_w) \vec{v}_i = 0 \tag{4.39}$$

genelleştirilmiş özdeğer probleminin çözümüne indirgenir [68]. S_B matrisi için maksimum(rank(S_B))=c-l olmaktadır. İspatı Ek B'de verilmiştir. **(4.39)** ifadesi ile verilen özdeğer probleminin özdeğerleri ve karşı düşen özvektörleri \vec{v}_i , V izdüşüm matrisinin sütunlarını oluştururlar. Bu matrisler reel ve yarı kesin pozitif olduklarından özdeğerleri de pozitif reeldir ve maksimum değere sahip olanın özvektörü, verilerin sınıflandırılmasında en iyi ayrımı sağlamaktadır [64]. Birden fazla sayıda ve birbirine eşit maksimum özdeğer çıkması durumunda, bu maksimum özdeğerlerden elde edilen ve farklı olan özvektörlerin herhangi biri seçilip veri sınıflandırılmasında kullanılabilir (bu farklı özvektörlerin hepsi veriler arasında aynı derecede ayırım sağlamaktadır [68]). Maksimum özdeğere karşı düşen izdüşüm vektörü elde edildikten sonra sınıflandırma devresi ile kullanılması Bölüm 5.3'de ele alınmıştır.

4.2 Eğiticili Perseptron Öğrenme Algoritması ile ÇAD ve ÇD Parametrelerinin Bulunması

Çift eşik hiperdüzlemlerinin oluşturulması için kullanılabilecek bir yöntem de yapay sinir ağlarında kullanılan eğiticili perseptron öğrenme algoritmasıdır [28]. Bölüm 3'de ÇAD ve ÇD yapıları kullanılarak eğik ızgaralı verilerin, çift eşik hiperdüzlemi ile nasıl sınıflandırılacağı incelenmiştir. Çift eşik hiperdüzlemi belirleyen katsayılar tek katlı perseptron öğrenme algoritmasından faydalanılarak da elde edilebilmektedir. Dolayısıyla perseptron öğrenme algoritması da Bölüm 3'de ele alınan verilerin sınıflandırılması için kullanılabilir. Şekil 4.4'de *n* girişli tek katlı perseptron yapısı verilmiştir.



Şekil 4.4 : *n* girişli tek katlı perseptron yapısı.

Şekil 4.4 ile verilen bu yapının tanım bağıntısı da şöyledir:

$$y = \begin{cases} 1 & \sum_{i=1}^{n} x_i w_i \ge k & i \zeta i n \\ 0 & \sum_{i=1}^{n} x_i w_i < k & i \zeta i n \end{cases}$$

$$(4.40)$$

Sınıflandırmada kullanılacak katsayılar (4.40) eşitliğindeki w ve k değerleridir. Bu w ve k değerleri ile hiperdüzlem denklemleri oluşturulur. Sınıflandırma için gerekli olan çift eşik hiperdüzlem denklemleri her farklı sınıf için aşağıdaki ifadeler kullanılarak elde edilir:

$$y_{i} = \begin{cases} 1 & \vec{x} \, \vec{v}_{i}^{T} - a \ge 0 \\ 0 & \vec{x} \, \vec{v}_{i}^{T} - a < 0 \end{cases}$$
(4.41)

$$y_{i} = \begin{cases} 1 & \vec{x} \, \vec{v}_{i}^{T} - b \leq 0 \\ 0 & \vec{x} \, \vec{v}_{i}^{T} - b > 0 \end{cases}$$
(4.42)

(4.41) ve (4.42) ifadelerinde kullanılan \vec{v} vektörü, *a* ve *b* parametrelerinin değerleri perseptron öğrenme algoritmasına göre hesaplanır.

Öğrenme algoritması sonucu, hiperdüzlem denklemleri

$$\vec{x}\,\vec{v}_i^T - a = 0 \tag{4.43}$$

$$\vec{x}\,\vec{v}_i^T - b = 0 \tag{4.44}$$

biçiminde yazılabilir.

Bu yöntemde veri, sınıflara uygun sayıda çift eşik hiperdüzlemleri ile ayrıştırılarak sınıflandırılmaktadır.

Uygun sayıda çift eşik hiperdüzlem denkleminin bulunması aşağıdaki yöntemi adım adım uygulayarak yapılır:

c_i (i=1,2,..,*k*) veri sınıflarını göstersin,

- 1. Herhangi bir c_i sınıfını diğer tüm sınıflardan ayıran uygun bir adet çift eşik hiperdüzlemi olup olmadığı perseptron öğrenme algoritmasıyla kontrol edilir.
 - i. Eğer varsa, *c_i* sınıfı için ağırlık katsayıları kaydedilir. Madde 2'ye geçilir.
 - ii. Eğer bulunamıyorsa madde 3'e geçilir.
- 2. Bir sonraki sınıfa geçilir.
 - i. Eğer bir sonraki sınıf sonuncu *k* sınıfı ise sınıflandırma işlemi sonlandırılır.
 - ii. Değil ise madde 1 uygulanır.
- 3. Çift eşik hiperdüzlem sayısı bir arttırılır ve c_i sınıfının diğer sınıflardan ayrılıp ayrılmadığı aynı algoritmayla kontrol edilir.
 - Ayrılabiliyor ise çift eşik hiperdüzlemlerinin ağırlık katsayıları kaydedilir. Madde 2'ye geçilir.
 - ii. Ayrılamıyor ise madde 3'e geçilir.

Perseptron yapısında kullanılan aktivasyon fonksiyonu keskin eşik fonksiyonu biçiminde olduğu için ağırlık katsayıları perseptron öğrenme algoritmasına göre güncellenerek elde edilir [14].

Perseptron öğrenme algoritması,

$$v_i(n+1) = v_i(n) - \eta (y_d - y_o) x_i$$
(4.45)

$$a_i(n+1) = a_i(n) - \eta (y_d - y_o)$$
(4.46)

$$b_i(n+1) = b_i(n) - \eta (y_d - y_o)$$
(4.47)

biçimindedir. Buradaki y_d elde edilmek istenilen sonuç, y_o ise elde edilen sonuç değeridir. η ise öğrenme katsayısı olarak isimlendirilir; ve 0 ile 1 arasında değer alır. Güncelleme işleminde $y_d = y_o$ olması durumunda ağırlık katsayısı değişmez. Öğrenme algoritması bütün ağırlık katsayıları değişmediği durumda durur [69].

Eğitim kümesindeki verilerden c_i sınıfına ait veriler kullanılarak güncelleme işlemi bittikten sonra **(4.41)** ve **(4.42)** ifadeleri kullanılarak bulunmuş olan çift eşik hiperdüzlemi, $a_i \leq \vec{x} \vec{v}_i^T \leq b_i$ bölgesini göstermektedir ve c_i sınıfına ait veriler bu bölge içinde bulunmaktadır. Öğrenme algoritması ile bulunan \vec{v} vektörü ÇAD parametreleri olan w'lara (ağırlık katsayıları), a ve b ise çekirdek devrenin I_1 , I_2 (kontrol akımları) değerlerine karşılık gelmektedir ve ÇD'nin I_H parametresinden yararlanılarak veri sınıfı belirlenmektedir. Şayet veri kümesindeki bir sınıf, iki veya daha çok sayıda çift eşik hiperdüzlemi gerektirirse, diğer bir deyişle iki veya daha çok sayıda bölgede aynı sınıfın olması durumunda, farklı bölgelere karşılık düşen aynı verilerin ÇD'lerinin I_H parametreleri aynı seçilerek veri sınıfının doğru sınıflanması sağlanır. Dolayısıyla uygun sayıda ÇAD ve ÇD bloğu kullanılarak sınıflandırma yapısı oluşturulur. Perseptron öğrenme algoritması kullanılarak üç ve dört boyutlu iki farklı veri kümesi Bölüm 5.3'de sınıflandırılmış ve sonuçları verilmiştir.

5. SINIFLANDIRICI DEVRE UYGULAMALARI

5.1 Kuantalayıcı

Gerçekleştirilmiş olan çekirdek devre yapıları kullanılarak kuantalayıcı devre oluşturulabilmektedir.

Sekiz seviyeli kuantalayıcı devre oluşturmak için sekiz adet çekirdek devre Şekil 5.1'de gösterildiği biçimde paralel olarak bağlanmıştır. Bu çekirdek devrelerin girişlerine aynı giriş işareti (kuantalanacak işaret) uygulanmıştır.



Şekil 5.1 : Sekiz seviyeli kuantalayıcı devre blok yapısı [55].

Kuantalayıcı yapı, aynı zamanda tek boyutlu 8 farklı veri türünü ayırt etmek için kullanılabilecek bir sınıflandırıcı yapısıdır. Ancak yapının kuantalayıcı olarak çalışması için her bir çekirdek devrenin I_1 , I_2 ve I_H kontrol parametrelerinin Çizelge 5.1'de verildiği biçimde uygulanmış olması gerekir. Çizelgede verilen parametreler doğrultusunda kuantalayıcı yapının benzetimi için devrenin girişine üçgen bir dalga uygulanmıştır. Devrenin giriş-çıkış karakteristiği Şekil 5.2'de verilmiştir. Ayrıca kuantalama devresinde kuanta aralıkları ve kuanta seviyesi kontrol akımlarına bağlı olarak değiştirilebilir.



Şekil 5.2 : Sekiz seviyeli kuantalayıcı devresi *I_{in}-I_{out}* karakteristiği.

Çizelge 5.1 : Kuantalayıcı Yapısında Kullanılan Çekirdek Devre Parametreleri.

| ÇD | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|------------------|-----|----|----|----|----|----|----|----|
| I_1 (μ A) | 0 | 5 | 10 | 15 | 20 | 25 | 30 | 35 |
| $I_2(\mu A)$ | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 |
| $I_H(\mu A)$ | 0.1 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |

Gerçekleştirilen kuantalayıcı yapı ile çekirdek devrenin kontrol parametrelerinin dışarıdan değiştirilebiliyor ve ayrıca ÇD bloklarının paralel bağlanabiliyor olmasının sağladığı avantajlar, bu çalışmada ele alınan yaklaşımın üstünlüğünü göstermesi açısından da yararlı olmaktadır.

5.2 Karakter Tanıma

Gerçekleştirilen devrenin bir başka uygulaması olarak çok kullanılmakta olan "Karakter Tanıma" problemi şablon uydurma metodu kullanılarak ele alınmıştır. Karakter tanıma çok tipik bir sınıflandırma örneği olarak gösterilmektedir [1,70]. Buradaki sınıflandırma işlemi ele alınan verinin bir şablona uyup uymadığının saptanması şeklindedir. Burada verilen örnekte şablonlar sayı karakterleri olarak düşünülmüş ve Şekil 5.3'de verilmiştir. Karakter tanıma, örüntü tanımanın özel bir şeklidir. Şablon uydurmadan büyük bir resmin içindeki küçük resimleri tanımada yararlanılır. Burada şablonun küçük resimler olduğu düşünülebilir. Hatta bu şablon herhangi bir karakter, resim ya da sayı olabilir. Her bir şablon Şekil 5.4'de gösterildiği gibi 4×5=20 hücreye bölünmüştür. Oluşabilecek bir satır hatasının düzeltilmesi için fazladan bir kolon kullanılmış ve karakterlerin yerleri özel olarak seçilmiştir. Şekil 5.5'de gösterilen karakter tanıma sınıflandırıcısı, örnek şablonun, her hücresinin diğer örnek şablonlar ile karşılaştırılabileceği şekilde tasarlanmıştır [71]. Bu şekil aynı zamanda Dört Boyutlu Sınıflandırıcı Blokları (DBSB) içermektedir. DBSB yapısı, çıkışları aynı yere bağlanmış ve çıkış fonksiyonu $f(x_1, x_2, x_3, x_4)$ ile ifade edilen dört adet çekirdek devrenin paralel bağlanmasından oluşmaktadır. Her şablonun her satırının farklı bir sonuç vermesi için DBSB yapısının çıkışı,

$$f(x_1, x_2, x_3, x_4) = x_1 2^0 + x_2 2^1 + x_3 2^2 + x_4 2^3$$
(5.1)

şeklinde seçilmiştir. Her DBSB'nin çıkışı, geçiş karakteristiği $g_i(y)$ fonksiyonu ile ifade edilen bir çekirdek devreye uygulanmıştır. Giriş örnek örüntüsünün satırları Şekil 5.5'de gösterildiği biçimde DBSB bloklarına uygulanır. Uygulanan giriş örüntüsüne göre, a_i , b_i , c_i , d_i (i=1,...,5) değerleri sayısal olarak "1" veya "0" olabilir ve DBSB yapısının çıkışı (5.1) ifadesine göre değerler alır. Bu değerler Çizelge 5.2'de verilmektedir. Bu çizelgeden görüldüğü gibi çıkışlara ilişkin satırların biri, diğeri ile çakışmamaktadır. Böylece y_i (i=1,...,5) çıkışlarını kontrol ederek giriş şablonunun ne olduğu söylenilebilir. Karakter tanıma işlemi bu aşamada gerçekleşmektedir. İstenen sadece karakter tanıma işleminin gerçekleştirilmesi ise, çıkışlar toplanarak tek bir çıkış ile elde edilebilir. Σy_i ile gösterilen bu toplam her bir giriş için farklı sonuç vermektedir.



Şekil 5.3 : Örnek örüntüler.

| <i>a</i> ₁ | b 1 | <i>c</i> ₁ | <i>d</i> ₁ |
|-----------------------|-----------------------|-----------------------|-----------------------|
| <i>a</i> ₂ | b ₂ | c_2 | <i>d</i> ₂ |
| <i>a</i> ₃ | b 3 | <i>c</i> ₃ | <i>d</i> ₃ |
| <i>a</i> ₄ | b ₄ | <i>c</i> ₄ | d_4 |
| <i>a</i> ₅ | b 5 | <i>c</i> ₅ | d_5 |

Şekil 5.4 : Şablon hücre gösterimi.

Karakter tanıma sınıflandırıcısında, $g_i(y_i)$ fonksiyonu ile gösterilen bloğun kontrol akımları, seçilmek istenen şablonu gösterecek şekilde ayarlanır. Ayırt edilmek istenen örüntü girişe uygulanması durumunda, ya da $g_i(y_i)$ çekirdek devresinin girişi uygun bölgede ise, z_i çıkışı sayısal olarak "1" değerini aksi durumda ise "0" değerini verecektir. Sonuç olarak, bulunmak istenen doğru şablon girişe uygulandığında "z" çıkışı genlik olarak "5" değerini, farklı bir şablon uygulandığında da "5" den daha küçük bir değeri verecektir.



Şekil 5.5 : Karakter tanıma sınıflandırıcısı blok diyagramı.

Ayrıca Çizelge 5.2'de, herhangi iki farklı giriş için en fazla üç adet y_i (i=1,2,..,5) çıkışının birbiri ile çakıştığı açık bir şekilde görülmektedir. Bu durumdan hata düzeltilmesinde yararlanılabilir.

| Şablonlar | \mathcal{Y}_1 | <i>Y</i> 2 | <i>y</i> ₃ | <i>Y</i> 4 | <i>Y</i> 5 | Σy_i |
|-----------|-----------------|------------|-----------------------|------------|------------|--------------|
| 0 | 7 | 5 | 5 | 5 | 7 | 29 |
| 1 | 8 | 8 | 8 | 8 | 8 | 40 |
| 2 | 14 | 8 | 14 | 2 | 14 | 52 |
| 3 | 14 | 8 | 12 | 8 | 14 | 56 |
| 4 | 10 | 10 | 14 | 8 | 8 | 50 |
| 5 | 7 | 1 | 7 | 4 | 7 | 26 |
| 6 | 1 | 1 | 7 | 5 | 7 | 21 |
| 7 | 7 | 4 | 4 | 4 | 4 | 23 |
| 8 | 14 | 10 | 14 | 10 | 14 | 62 |
| 9 | 7 | 5 | 7 | 4 | 4 | 27 |

Çizelge 5.2 : Farklı örüntüler için y_i (i=1,...,5) çıkış değerleri.

Sınıflandırıcı devresinin, şablon uydurma yöntemi ile karakter tanıma uygulaması için Şekil 5.3'de verilen şablonlar ile benzetimi yapılmıştır. Bunun için bir şablon seçilmiş, 0 ms ile 1ms arasında şablon "0", 1 ms ile 2 ms arasında şablon "1" ve diğer şablonlar 10 ms olana kadar benzer şekilde uygulanmış; benzetim 10 şablon seçilerek tekrarlanmıştır. Benzetimde "1" genliği için 10 μ A değeri seçilmiştir. Karakter tanıma uygulamasının benzetim sonuçları, ilk beş şablon için Şekil 5.6'da son beş şablon için ise Şekil 5.7' de verilmiştir.



Şekil 5.6 : İlk beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları.

Bu şekillerden de görüldüğü gibi 0 ms ile 10 ms arasında sadece 1 ms'lik bir kısmın genliği 10 µA olmakta ve devrenin istenildiği gibi çalıştığı görülmektedir.

Çizelge 5.2'deki sadece bir satırın, bir veya daha çok hücresinde hata oluştuğunda z_i çıkışı genlik olarak "4" değerini alacaktır. Oluşan hatanın düzeltilmesi için sınıflandırıcı bloğunun sonuna g(z) fonksiyonunu sağlayan bir adet çekirdek devre eklenebilir. Bu çekirdek devre oluşan girişine, "4" veya "5" genlikli bir işaret geldiğinde çıkışta genlik olarak "1" verilmesini dolaysıyla da hatanın düzeltilmesini sağlamış olacaktır.



Şekil 5.7 : Son beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları.

Karakter tanıma sınıflandırıcısının, aynı zamanda hata düzeltme işlemini de gerçekleştirdiğini göstermesi açısından, Şekil 5.8'de verilen hatalı şablonlar, 1 ms ile 2 ms arasında şablon "1", 2 ms ile 3 ms arasında şablon "2" ve diğerleri, 7 ms olana kadar benzer şekilde bloğa uygulanmıştır. Benzetim sonucu Şekil 5.9'da gösterilmiştir. Sonuçtan da görüldüğü gibi test şablonlarının tek bir satırında, tek ya da daha çok hücrede, hata olması durumunda doğru sınıflandırma işlemi gerçeklenmiştir [71].



Şekil 5.8 : Hatalı test şablonları.



Şekil 5.9 : Karakter tanıma sınıflandırıcısı hata düzeltme benzetim sonucu.10

Bu örnekle, şablon uydurma yöntemi ile karakter tanıma uygulaması için gerçekleştirilmiş olan sınıflandırıcı devre yapılarının şablon tanıma ve benzeri uygulamalarda da kullanılabileceği gösterilmiştir.

5.3 İris ve Haberman Verisinin Sınıflandırılması

5.3.1 İris verisinin Fisher tabanlı algoritma ile sınıflandırılması

Literatürde çok kullanılan veri kümelerinden biri iris veri kümesidir. Bu veri kümesi 1936 yılında Fisher tarafından oluşturulmuştur [72]. Sınıflandırma uygulamalarında kullanılan en temel veri sınıflarındandır. Veri kümesi dört adet özellik, diğer bir ifadeyle dört boyutlu veri (x_1, x_2, x_3, x_4) ve üç sınıftan (c_1, c_2, c_3) oluşmaktadır. Bu sınıflar Setosa, Versicolor ve Virginica isimli bitki tipleridir. Veri kümesinin sınıflandırılması bitkinin taç ve çanak yapraklarının uzunluk ve kalınlık değerlerine bakılarak gerçekleştirilir. İris veri kümesi 150 adet veri içermektedir. Bu veriler üç sınıfa eşit sayıda dağılmıştır. Veri kümesi detaylı olarak Ek C1'de verilmiştir. Sınıflandırma için kullanılacak veri kümesinin özellikleri santimetre birimi cinsinden verilmiş ve gerçekleştirilmiş olan sınıflandırıcı ile test edilmiştir. Bu testte sınıflandırıcının kontrol parametrelerinin, Bölüm 4.1'de geliştirilmiş olan Fisher tabanlı algoritma ile hesaplanması için, her sınıftan 40 veri ve toplamda 120 veri kullanılmıştır. Fisher tabanlı algoritma sonucunda veriler arasında en iyi ayrımı sağlayacak izdüşüm vektörü $\vec{v} = (0.57 - 0.80 \ 0.10 \ 0.14)$ olarak elde edilmiştir. Bu vektör (4.4) ifadesinde belirtildiği şekilde kullanılarak iris veri kümesinin orijine olan izdüsüm uzaklığı hesaplanarak verilerin orijine olan izdüsüm uzaklıkları Sekil 5.10'da gösterilmiştir. Bu şekil üstündeki, yıldız işaretleri c_1 , artı işaretleri c_2 ve küçük daireler de c₃ sınıfına ait verilerdir. Şekil 5.10'dan görüldüğü gibi yatay eksen, verilerin orijine olan izdüşüm uzaklıklarını ve dikey eksen ise veri numarasını göstermektedir. Dolayısıyla Şekil 5.10'daki verilerin uzaklıkları ile iris veri kümesi sınıflandırılabilmektedir. Bu sınıflandırma işlemi için ÇAD ve ÇD yapılarından faydalanılmıştır. Şekil 5.11'de iris verisi sınıflandırıcısı blok diyagramı verilmiştir. Bu yapıdaki her bir ÇAD yapısının ağırlık katsayısı w, Fisher tabanlı algoritma ile elde edilen \vec{v} vektörünün elemanlarına denk düşmektedir. ÇAD yapılarının çıkış değişkeni akım boyutunda olduğundan çıkış uçları birbirlerine bağlanarak toplama işlemi gerçeklenir. Bu toplam, veri kümesinin orijine olan izdüşüm uzaklığını gösterir. Dolayısıyla sınıflandırma işlemi için izdüşüm uzaklıkları (akım değerleri) üç çıkışlı Akım Çoğullayıcı (AÇ) devre ile çekirdek devre girişlerine uygulanmıştır. Şekil 5.10'da gösterildiği biçimde, her bir sınıfın ayrıldığı noktalar çekirdek devrenin kontrol akım değerlerine karşılık gelmektedir.



Şekil 5.10 : İzdüşürülmüş iris verilerinin orijine uzaklıkları.



Şekil 5.11 : İris verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuş).

Şekil 5.11'deki yapının gerçekleştirilmesi için kullanılan akım çoğullayıcının n çıkışlı devre şeması Şekil 5.12'de ve tanım bağıntısı da **(5.2)** ifadesi ile verilmiştir. Akım çoğullayıcı sayesinde, ÇAD akımlarının toplamı tekrar çekirdek devrelere
uygulanarak, Şekil 5.10'da gösterilen bölgeler, çıkışın üç farklı değerden birini almasıyla seçilir.

Şekil 5.11'deki devrenin benzetiminde kullanılan *n* çıkışlı akım çoğullayıcı yapı için 0.35 µm AMS CMOS teknoloji parametreleri ve besleme gerilimleri V_{DD} ve V_{SS} ± 1.65 V olarak kullanılmıştır. Kutuplama akımının 18 µA olması için V_B gerilimi 0.8 V olarak alınmıştır. Akım çoğullayıcı devresinde kullanılan tranzistor boyutları da Çizelge 5.3'de verilmiştir.



Şekil 5.12 : Akım çoğullayıcı devre şeması [65].

Cizelge 5.3 : *n* çıkışlı akım çoğullayıcı devresi MOS tranzistorların boyutları.

| MOSFET | W [µm] | L [µm] |
|--|--------|--------|
| M_1, M_3, M_4 | 21 | 1.05 |
| M_5, M_6 | 6.3 | 1.05 |
| M ₂ , M _{a-1} , M _{a-2} ,, M _{a-n} | 42.7 | 1.05 |
| M ₇ , M _{b-1} , M _{b-2} ,, M _{b-n} | 11.2 | 1.05 |

Sınıflandırıcının başarımı için her sınıftaki test kümesinden geri kalan 10, toplamda 30 adet veri kullanılarak Şekil 5.11'de gösterilen yapının SPICE benzetimi yapılmıştır. Benzetim için kullanılan test verileri Çizelge 5.4'de gösterilen aralıklarla Şekil 5.11'deki yapıya uygulanmıştır.

| Verilerin uygulanma süresi | x_1 | x_2 | x_3 | x_4 | Sınıf |
|----------------------------|-------|-------|-------|-------|-----------------------|
| 0 ms - 1 ms | 4.3 | 2.3 | 1.4 | 0.2 | c_1 |
| 1 ms - 2 ms | 5.7 | 2.7 | 3.9 | 1.1 | c_2 |
| 2 ms - 3 ms | 5.7 | 2.7 | 4.8 | 1.8 | c_2 |
| 3 ms - 4 ms | 4.9 | 2.2 | 6 | 2.5 | c_3 |
| 4 ms - 5 ms | 5.6 | 2.5 | 5.1 | 1.9 | c_3 |
| 5 ms - 6 ms | 4.6 | 3 | 1.7 | 0.4 | c_1 |
| 6 ms - 7 ms | 4.7 | 3.1 | 1.5 | 0.1 | c_1 |
| 7 ms - 8 ms | 6.1 | 2.9 | 3.8 | 1.1 | c_2 |
| 8 ms - 9 ms | 6.4 | 3 | 4 | 1.3 | c_2 |
| 9 ms - 10 ms | 4.9 | 2 | 4.7 | 1.4 | c_2 |
| 10 ms - 11 ms | 4.8 | 3.2 | 1.2 | 0.2 | c_1 |
| 11 ms - 12 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 12 ms - 13 ms | 6.3 | 2.8 | 6.5 | 1.8 | c_3 |
| 13 ms - 14 ms | 6.7 | 3 | 6.4 | 2 | c_3 |
| 14 ms - 15 ms | 7.2 | 3.2 | 5.4 | 2.1 | c_3 |
| 15 ms -16 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |
| 16 ms - 17 ms | 5.3 | 3.7 | 1.5 | 0.2 | c_1 |
| 17 ms - 18 ms | 5.4 | 3.8 | 1.3 | 0.3 | c_1 |
| 18 ms - 19 ms | 6.3 | 3.0 | 4.4 | 1.3 | c_2 |
| 19 ms - 20 ms | 6.3 | 3.0 | 4.1 | 1.3 | c_2 |
| 20 ms - 21 ms | 6.9 | 3.1 | 5.6 | 2.4 | c_3 |
| 21 ms - 22 ms | 7.6 | 3.3 | 5.7 | 2.5 | c_3 |
| 22 ms - 23 ms | 7.7 | 3.4 | 5.2 | 2.3 | c_3 |
| 23 ms - 24 ms | 6.4 | 3.0 | 4.0 | 1.3 | c_2 |
| 24 ms - 25 ms | 6.4 | 3.0 | 4.4 | 1.2 | c_2 |
| 25 ms - 26 ms | 6.5 | 3.0 | 4.0 | 1.2 | c_2 |
| 26 ms - 27 ms | 7.7 | 3.4 | 5.0 | 1.9 | c_3 |
| 27 ms - 28 ms | 7.7 | 3.6 | 5.2 | 2.0 | <i>c</i> ₃ |
| 28 ms - 29 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 29 ms - 30 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |

Çizelge 5.4 : Şekil 5.11'deki iris verisi sınıflandırıcısı test kümesi.

Çizelge 5.4'de verilmiş olan iris verisinin özellikleri olan x_i (i=1,2,...4) değerleri cm boyutundadır. Ancak benzetim için bu değerler ÇAD sayesinde gerekli normalizasyon yapılarak akıma dönüştürülmüş ve çekirdek devreye uygulanmıştır. Çizelge 5.4'deki 0-1 ms arasındaki x_1 =4.3 verisinin ÇAD-1 yapısında kullanılması şöyle özetlenebilir: Şekil 3.20'deki yapıda V_y =0.43 V ve R_1 =10 k Ω alınması durumunda giriş büyüklüğü $V_y/R_1 = 43 \,\mu$ A olarak akıma çevrilir. Ayrıca Fisher tabanlı algoritma ile bulunan $\vec{v} = (0.57 - 0.80 \ 0.10 \ 0.14)$ vektörünün ilk elemanı 0.57 olduğu için $R_2/R_3 = 57$ kΩ / 100 kΩ alınarak (**3.6**) ifadesine göre çıkış akımı elde edilir. Bu çıkış değeri elde edilirken x_1 =4.3 olan giriş verisi µA boyutunda gibi düşünülüp 10 kat büyütülerek 43 µA olarak \vec{v} vektörünün elemanı ile çarpılmıştır. Benzer durum diğer giriş verileri içinde uygulandığından, orijine olan izdüşüm uzaklığı Şekil 5.10'da gösterilmiş olan uzaklığın 10 katı olarak elde edilmiştir. Bu durumda Şekil 5.10'da c_1 sınıfına ait verilerin orijine olan izdüşüm uzaklığı 0.1 ile 0.8 arasında olduğu halde verilerin ayırt edilebilmesi için Şekil 5.11'deki çekirdek devre-1'in kontrol akım değerleri I_1 =1 µA, I_2 =8 µA olacak şekilde seçilmiştir. Şekil 5.11'de kullanılan diğer ÇAD ve ÇD parametreleri benzer şekilde bulunmuş ve sırası ile Çizelge 5.5'de ve Çizelge 5.6'da verilmiştir. Ayrıca Şekil 5.11'de kullanılan ÇAD çıkışları \vec{v} vektörünün elemanlarının pozitif ya da negatif olmasına göre seçildiğinden, Şekil 3.20'den farklı olarak tek çıkışları bulunmaktadır. Diğer bir ifadeyle \vec{v} vektörünün ilk elemanı pozitif, ikinci elemanı negatiftir, dolayısıyla Şekil 5.11'de kullanılan ÇAD-1 yapısı için pozitif çıkış ucu, ÇAD-2 yapısı için negatif çıkış ucu kullanılmıştır.

Çizelge 5.5 : İris sınıflandırıcısı ÇAD direnç değerleri.

| Dirençler | ÇAD-1 | ÇAD-2 | ÇAD-3 | ÇAD-4 |
|-----------|-----------------------|--------|-----------------------|-----------------------|
| R_1 | 10 kΩ | 10 kΩ | 10 kΩ | 10 kΩ |
| R_2 | 57 kΩ | 80 kΩ | 10 kΩ | 14 kΩ |
| R_3 | $100 \text{ k}\Omega$ | 100 kΩ | $100 \text{ k}\Omega$ | $100 \text{ k}\Omega$ |

Şekil 5.11'deki yapının çıkış akımı çekirdek devrelerin I_H akımları ile kontrol edildiğinden her çekirdek devrenin I_H akımı farklı seçilerek veri kümesinin sınıflandırılması sağlanır. Ayrıca iris kümesindeki veriler sınıflandırıcıya uygulanmadan önce sınıflandırıcının uygun çalışma bölgesi göz önüne alınarak 10 kat büyütülmüştür. Böylece ÇAD yapısı ile sınıflandırıcı devrenin çalışma bölgesi dışındaki veri kümelerinin de ayırt edilebildiği gösterilmiştir.

Çizelge 5.6 : İris sınıflandırıcısı çekirdek devre kontrol akımları.

| Kontrol akımları | ÇD-1 | ÇD-2 | ÇD-3 |
|------------------|-------|-------|-------|
| I_1 | 1 μΑ | 16 µA | 21 µA |
| I_2 | 8 μΑ | 19 µA | 27 μΑ |
| I_H | 10 µA | 20 µA | 30 µA |

Çizelge 5.4'de sınıflandırıcı sonuçlarının ne çıkması gerektiği gözükmektedir. Şekil 5.11'deki yapının benzetim sonucu Şekil 5.13'de verilmiştir; buna göre benzetim sonucunda çıkış değerinin 10 μ A olması durumunda sonuç c_1 , 20 μ A olması durumunda sonuç c_2 , 30 μ A olması durumunda sonuç c_3 olmaktadır.



Şekil 5.13 : Şekil 5.11'deki İris verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.13'de benzetim sonucundan görüldüğü gibi 30 adet iris test verisi doğru olarak sınıflandırılmıştır ve bu veri için Bölüm 4.1'de verilen Fisher tabanlı algoritma kullanılarak veri kümesinin sınıflandırılabilmesi sağlanmıştır.

5.3.2 Haberman verisinin Fisher tabanlı algoritma ile sınıflandırılması

Literatürde sınıflandırmada çok kullanılan diğer bir veri kümeside Haberman veri kümesidir [73]. Bu veri kümesi üç boyutlu (x_1 , x_2 , x_3 ,) olup iki sınıftan (c_1 , c_2) oluşmaktadır. Bu küme Haberman tarafından 1958-1970 yılları arasında Şikago Üniversitesi Billing Hastanesinde göğüs kanseri teşhisi ile ameliyat olmuş hastalardan yararlanılarak oluşturulmuştur. Küme ameliyat sonrasında beş yıldan daha uzun yaşayan ya da beş yıl içinde ölen hastaları sınıflandırmaktadır. Sınıflandırma işlemi için hastanın yaşı, ameliyat tarihi ve pozitif netice veren göğüs kanserli hasta sayısı kullanılmıştır; veri kümesi toplam 306 adet veri içermekte ve bu verilerin 255 adeti c_1 sınıfına, 81 adeti ise c_2 sınıfına ait bulunmaktadır. Sınıflandırıcıların başarımlarının karşılaştırılmasında kolaylık sağlaması açısından iki sınıfa ait rastgele 104 veri alınmış ve sınıflara eşit dağıtılmıştır. Bu verilerden her sınıftan 42 ve toplamda 84 veri kullanılarak, Bölüm 4.1'de ele alınan Fisher tabanlı algoritma Haberman veri kümesine uygulanmıştır. Fisher tabanlı algoritma sonucunda izdüşüm vektörü $\vec{v} = (0.23 \ 0.50 \ 0.80)$ olarak elde edilmiştir. Böylece bu izdüşüm vektörü (4.4) ifadesinde gösterildiği şekilde kullanılarak Haberman veri kümesinin orijine olan izdüşüm uzaklıkları bulunmuş ve Şekil 5.14'deki şekilde çizdirilmiştir. Bu şekilde, küçük yuvarlaklar c_1 ve artı işaretleri c_2 sınıfına ait verilerdir.



Şekil 5.14 : İzdüşürülmüş haberman verilerinin orijine uzaklıkları.

Şekil 5.14'den görüldüğü gibi yatay eksen verilerin orijinden uzaklıklarını dikey eksen ise veri numarasını göstermektedir. Dolayısıyla iris veri kümesinde olduğu

gibi, aynı yöntemle Şekil 5.14'deki verilerin uzaklıkları ile de Haberman veri kümesi de sınıflandırılabilmektedir. Bu sınıflandırma işlemi için ÇAD ve ÇD yapılarından faydalanılmıştır. Şekil 5.15'de Haberman verisi sınıflandırıcısı blok diyagramı verilmiştir. ÇAD çıkışlarındaki akımların toplamı, girişe uygulanan verinin orijine olan izdüşüm uzaklığına karşılık gelmektedir. Dolayısıyla sınıflandırma işlemi için, izdüşüm uzaklığı (akım değeri) iki çıkışlı akım çoğullayıcı devre ile iki çekirdek devre girişine uygulanmıştır. Bu sayede çekirdek devreler ile Şekil 5.14'de gösterilen bölgelerin seçilmesi sağlanmaktadır.



Şekil 5.15 : Haberman verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuştur).

Çizelge 5.7'de verilmiş olan Haberman verisinin özellikleri olan x_i (i=1,2,3) değerleri µA akım boyutunda düşünülmüştür. Benzetim için Çizelge 5.7'deki 0-1 ms arasında x_1 =34 verisinin ÇAD-1 yapısında kullanılması şu şekilde özetlenebilir: Şekil 3.20'deki yapıda V_y =34 mV ve R_1 =10 k Ω alınması durumunda giriş büyüklüğü akımı V_y/R_1 = 34 µA olarak seçilmiştir. Ayrıca Fisher tabanlı algoritmanın uygulanmasıyla bulunan $\vec{v} = (0.23 \ 0.50 \ 0.80)$ vektörünün ilk elemanı 0.23 olduğu için, $R_2/R_3 = 23 k\Omega/100 k\Omega$ alınarak (3.6) ifadesine göre çıkış akımı elde edilir. Diğer taraftan Şekil 5.14'de c_1 sınıfına ait verilerin orijine olan izdüşüm uzaklığı 38 ile 45 arasında olduğu için Şekil 5.15'deki çekirdek devre-1'in kontrol akımı değerleri I_1 =38 µA, I_2 =45 µA olacak şekilde seçilmiştir. Benzer şekilde Şekil 5.15'de kullanılan ÇAD ve ÇD parametreleri bulunmuş ve sırası ile Çizelge 5.8'de ve Çizelge 5.9'da verilmiştir. Ayrıca her çekirdek devrenin $I_{\rm H}$ akımı farklı seçilmiştir. Bu sayede Şekil 5.15'deki yapının çıkış akımına bakılarak veri sınıfının bulunması sağlanmıştır.

| Verilerin uygulanma süresi | x_1 | x_2 | x_3 | Sınıf |
|----------------------------|-------|-------|-------|-----------------------|
| 0 ms - 1 ms | 34 | 60 | 1 | c_1 |
| 1 ms - 2 ms | 61 | 68 | 1 | c_2 |
| 2 ms - 3 ms | 51 | 59 | 3 | c_2 |
| 3 ms - 4 ms | 37 | 59 | 6 | c_1 |
| 4 ms - 5 ms | 54 | 58 | 1 | c_1 |
| 5 ms - 6 ms | 61 | 62 | 5 | c_2 |
| 6 ms - 7 ms | 42 | 63 | 1 | c_1 |
| 7 ms - 8 ms | 53 | 61 | 1 | c_1 |
| 8 ms - 9 ms | 48 | 67 | 7 | c_2 |
| 9 ms - 10 ms | 65 | 66 | 15 | c_2 |
| 10 ms - 11 ms | 60 | 59 | 17 | c_2 |
| 11 ms - 12 ms | 42 | 59 | 2 | c_1 |
| 12 ms - 13 ms | 30 | 62 | 3 | c_1 |
| 13 ms - 14 ms | 65 | 62 | 22 | c_2 |
| 14 ms - 15 ms | 41 | 60 | 23 | c_2 |
| 15 ms - 16 ms | 46 | 58 | 3 | c_1 |
| 16 ms - 17 ms | 42 | 61 | 4 | c_1 |
| 17 ms - 18 ms | 72 | 67 | 3 | c_1 |
| 18 ms - 19 ms | 47 | 63 | 23 | c_2 |
| 19 ms - 20 ms | 43 | 58 | 52 | <i>c</i> ₂ |

Çizelge 5.7 : Şekil 5.15'deki Haberman verisi sınıflandırıcısı test kümesi.

Çizelge 5.8 : Haberman sınıflandırıcısı ÇAD direnç değerleri.

| Dirençler | ÇAD-1 | ÇAD-2 | ÇAD -3 |
|-----------|--------|-------|--------|
| R_1 | 10 kΩ | 10 kΩ | 10 kΩ |
| R_2 | 2.3 kΩ | 5 kΩ | 8 kΩ |
| R_3 | 10 kΩ | 10 kΩ | 10 kΩ |

Çizelge 5.9 : Haberman sınıflandırıcısı çekirdek devre kontrol akımları.

| Kontrol akımları | ÇD-1 | ÇD-2 |
|------------------|-------|-------|
| I_1 | 38 µA | 45 μΑ |
| I_2 | 46 µA | 82 μΑ |
| I_H | 10 µA | 20 µA |

Sınıflandırıcının başarımını sınamak için her sınıftan geri kalan 10 adet, toplamda 20 adet veri kullanılarak Şekil 5.15'deki sınıflandırıcı yapının SPICE benzetimi yapılmıştır. Benzetim için kullanılan test verileri Çizelge 5.7'de gösterilen aralıklar ile Şekil 5.15'deki yapıya uygulanmıştır. Çizelge 5.7'de sınıflandırıcı sonuçlarının ne olması gerektiği gözükmektedir. Şekil 5.16'daki benzetim sonucu, çıkış değeri 10 μ A ise verinin c_1 , 20 μ A ise verinin c_2 sınıfına ait olduğunu göstermektedir.



Şekil 5.16 : Şekil 5.15' deki haberman verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.16'da benzetim sonucundan görüldüğü gibi 20 adet Haberman test verisi Fisher tabanlı algoritma ile doğru olarak sınıflandırılmıştır.

5.3.3 İris verisinin perseptron öğrenme algoritması ile sınıflandırılması

İris veri kümesi, Bölüm 4.2'de ele alınan CAD ve CD parametrelerinin perseptron öğrenme algoritması kullanılarak bulunmasından faydalanılarak sınıflandırılmıştır. Öğrenme algoritması yardımı ile ÇAD ve ÇD parametrelerinin hesaplanması için her sınıftan 40 veri, toplamda 120 veri kullanılmıştır. Veri kümesi dört boyutlu ve sınıf sayısı üç olduğu için, Bölüm 4.2'de incelenen öğrenme algoritmasına göre 3 adet çift eşik hiperdüzlemi elde edilmiş olup, bu çift eşik hiperdüzlemlerinden biri c_1 sınıfını diğer ikisi ise c_2 sınıfının bulunduğu bölgeleri göstermektedir. Üç adet çift eşik hiperdüzleminin belirlediği bölgelerin hiçbirinde bulunmayan veriler ise üçüncü sınıfa ait veri olarak sınıflandırılmıştır. Bu algoritma sonucu, her sınıf için gereken \vec{v} vektörü (ÇAD parametreleri w_i 'ler) ve a,b sayıları (ÇD parametreleri, I_1 ve I_2) bulunmuştur. Böylece c_1 sınıfına ait veriler için $\vec{v}_{c1} = (0.25 \quad 0.96 \quad -1.50 \quad -0.63)$ vektörü, a=1 ve b=3.5 olarak elde edilmiştir. Benzer şekilde c_2 sınıfına ait veriler için bulunan ilk $\vec{v}_{c2-1} = (0.3 \ 1.0 \ -0.2 \ 3.7)$ vektörü, $a_1 = 7.8$ ve $b_1 = 9.3$ ve ikinci $\vec{v}_{c2-2} = (0.4 \quad 2.0 \quad -0.1 \quad 3.5)$ vektörü, $a_2=14$ ve $b_2=15$ olarak bulunmuştur. Elde edilen bu sonuçlar için oluşturulmuş olan iris sınıflandırıcısı bloğu Şekil 5.17'de verilmiştir.

Sınıflandırıcının başarımını sınamak için her sınıftan geri kalan 10, toplamda 30 adet veri kullanılarak Şekil 5.17'de blok diyagramı verilen devrenin SPICE benzetimi yapılmıştır. Benzetim için kullanılan test verileri Çizelge 5.10'da gösterilen aralıklar ile Şekil 5.17'deki devreye uygulanmıştır.

| Verilerin Uygulanma Süresi | x_1 | x_2 | x_3 | x_4 | Sınıf |
|----------------------------|-------|-------|-------|-------|-------|
| 0 ms - 1 ms | 4.3 | 2.3 | 1.4 | 0.2 | c_1 |
| 1 ms - 2 ms | 5.7 | 2.7 | 3.9 | 1.1 | c_2 |
| 2 ms - 3 ms | 5.7 | 2.7 | 4.8 | 1.8 | c_2 |
| 3 ms - 4 ms | 4.9 | 2.2 | 6 | 2.5 | c_3 |
| 4 ms - 5 ms | 5.6 | 2.5 | 5.1 | 1.9 | c_3 |
| 5 ms - 6 ms | 4.6 | 3 | 1.7 | 0.4 | c_1 |
| 6 ms - 7 ms | 4.7 | 3.1 | 1.5 | 0.1 | c_1 |
| 7 ms - 8 ms | 6.1 | 2.9 | 3.8 | 1.1 | c_2 |
| 8 ms - 9 ms | 6.4 | 3 | 4 | 1.3 | c_2 |
| 9 ms - 10 ms | 4.9 | 2 | 4.7 | 1.4 | c_2 |
| 10 ms - 11 ms | 4.8 | 3.2 | 1.2 | 0.2 | c_1 |
| 11 ms - 12 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 12 ms - 13 ms | 6.3 | 2.8 | 6.5 | 1.8 | c_3 |
| 13 ms - 14 ms | 6.7 | 3 | 6.4 | 2 | c_3 |
| 14 ms - 15 ms | 7.2 | 3.2 | 5.4 | 2.1 | c_3 |
| 15 ms -16 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |
| 16 ms - 17 ms | 5.3 | 3.7 | 1.5 | 0.2 | c_1 |
| 17 ms - 18 ms | 5.4 | 3.8 | 1.3 | 0.3 | c_1 |
| 18 ms - 19 ms | 6.3 | 3.0 | 4.4 | 1.3 | c_2 |
| 19 ms - 20 ms | 6.3 | 3.0 | 4.1 | 1.3 | c_2 |
| 20 ms - 21 ms | 6.9 | 3.1 | 5.6 | 2.4 | c_3 |
| 21 ms - 22 ms | 7.6 | 3.3 | 5.7 | 2.5 | c_3 |
| 22 ms - 23 ms | 7.7 | 3.4 | 5.2 | 2.3 | c_3 |
| 23 ms - 24 ms | 6.4 | 3.0 | 4.0 | 1.3 | c_2 |
| 24 ms - 25 ms | 6.4 | 3.0 | 4.4 | 1.2 | c_2 |
| 25 ms - 26 ms | 6.5 | 3.0 | 4.0 | 1.2 | c_2 |
| 26 ms - 27 ms | 7.7 | 3.4 | 5.0 | 1.9 | c_3 |
| 27 ms - 28 ms | 7.7 | 3.6 | 5.2 | 2.0 | c_3 |
| 28 ms - 29 ms | 5.4 | 3.8 | 1.6 | 0.6 | c_1 |
| 29 ms - 30 ms | 5.4 | 3.9 | 1.9 | 0.4 | c_1 |

Çizelge 5.10 : Şekil 5.17'deki iris verisi sınıflandırıcısı test kümesi.

Çizelge 5.10'da verilmiş olan iris verisinin özellikleri olan x_i (*i*=1-4) değerleri cm boyutundadır. Ancak benzetimde bu değerler ÇAD sayesinde gerekli normalizasyon yapılarak akıma dönüştürülmüş ve çekirdek devreye uygulanmıştır. Benzetim yapılırken, Çizelge 5.10'daki 0-1 ms arasında x_1 =4.3 verisinin, ÇAD-1 yapısında kullanılması şu şekilde özetlenebilir: Şekil 3.20'deki devrede V_y =0.43 V ve R_1 =10 k Ω alınarak giriş büyüklüğü $V_y/R_1 = 43 \,\mu\text{A}$ olacak şekilde akım boyutuna dönüştürülür. Ayrıca öğrenme algoritması ile c_1 sınıfı için bulunan $\vec{v}_{c1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$ vektörünün ilk elemanı 0.25 olduğu için, $R_2/R_3 = 25 \,\mu\Omega$ / 100 $\mu\Omega$ alınarak (3.6) ifadesine göre çıkış akımı elde edilir. Bu çıkış değeri elde edilirken, x_1 =4.3 olan giriş verisi μ A boyutunda gibi düşünülüp 10 kat büyütülmüş ve 43 μ A olarak \vec{v}_{c1} vektörünün ilk elemanı ile çarpılmıştır. Bu durumda c_1 sınıfı için bulunmuş olan a=1 ve b=3.5 değerleri (çekirdek devre I_1 , I_2 parametresi) 10 kat büyük olacaktır. Diğer bir ifade ile Şekil 5.17'deki çekirdek devre-1'in kontrol akım değerleri $I_1=10 \ \mu$ A, $I_2=35 \ \mu$ A olarak seçilmiştir. Benzer durum göz önünde alınarak Şekil 5.17'de kullanılan ÇAD ve ÇD parametreleri, c_1 ve c_2 sınıfı için bulunarak sırası ile Çizelge 5.11'de ve Çizelge 5.12'de verilmiştir. Ayrıca Şekil 5.17'deki ÇAD yapılarının çıkışları \vec{v}_{ci} vektörünün elemanlarının pozitif ya da negatif olmasına göre seçilmiştir. Bu nedenle Şekil 3.20'den farklı olarak tek çıkış bulunmaktadır. Diğer bir ifadeyle $\vec{v}_{c1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$ vektörünün ilk ve ikinci elemanı pozitif, üçüncü ve dördüncü elemanı negatiftir. Dolayısıyla Şekil 5.17'de kullanılan ÇAD-1 ve ÇAD-2 yapısı için pozitif çıkış ucu, ÇAD-3 ve ÇAD-4 yapısı için negatif çıkış ucu kullanılmıştır.



Şekil 5.17 : İris verisi sınıflandırıcı bloğu (öğrenme algoritması sonucu oluşturulmuş).

| ÇAD-1 | ÇAD -2 | ÇAD -3 | ÇAD -4 | ÇD-1 |
|---------------------------|---------------------------|--------------------------|---------------------------|--------------------|
| $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $R_1=10 \text{ k}\Omega$ | $I_1 = 10 \ \mu A$ |
| $R_2=2.5 \text{ k}\Omega$ | $R_2=9.6 \text{ k}\Omega$ | $R_2=15 \text{ k}\Omega$ | $R_2=6.3 \text{ k}\Omega$ | $I_2 = 35 \mu A$ |
| $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $R_3=10 \text{ k}\Omega$ | $I_H = 10 \ \mu A$ |

Çizelge 5.11 : Şekil 5.17'de c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

Çizelge 5.12 : Şekil 5.17'de c_2 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

| ÇAD-5 | ÇAD-6 | ÇAD-7 | ÇAD-8 | ÇD-2 | ÇAD-9 | ÇAD-10 | ÇAD-11 | ÇAD-12 | ÇD-3 |
|-----------------------------|-----------------------------|-----------------------------|-----------------------------|------------------------------|-----------------------------|-----------------------------|-----------------------------|-----------------------------|-------------------|
| $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | <i>I</i> ₁ =78 μA | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $R_1 = 10 \mathrm{k}\Omega$ | $I_1 = 140 \mu A$ |
| $R_2=3 \text{ k}\Omega$ | $R_2=10$ k Ω | $R_2=2$ k Ω | $R_2=37$ k Ω | <i>I</i> ₂ =93 μA | $R_2=4$ k Ω | $R_2=20$ k Ω | $R_2=1 \text{ k}\Omega$ | $R_2=35$ k Ω | $I_2 = 150 \mu A$ |
| $R_3=10k\Omega$ | $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $I_H = 20 \mu A$ | $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $R_3=10$ k Ω | $I_H=20 \ \mu A$ |

Şekil 5.17'deki yapının çıkış akımı çekirdek devrelerin I_H akımları ile kontrol edilmektedir. Dolayısıyla her çekirdek devrenin I_H akımı farklı seçilerek veri kümesinin sınıflandırılması sağlanmıştır. Benzetim için kullanılan test verileri, Çizelge 5.10'da gösterilen aralıklarla Şekil 5.17'deki yapıya uygulanmıştır. Çizelge 5.10'da sınıflandırıcı sonuçlarının ne çıkması gerektiği gözükmektedir; benzetim sonucu Şekil 5.18'de verilmiştir. Bu şekildeki benzetim sonucunda çıkış değerinin 10 μ A olması durumunda sonuç c_1 , 20 μ A olması durumunda ise c_2 , 0 μ A olması durumunda da c_3 olarak değerlendirilmiştir.



Şekil 5.18 : Şekil 5.17'deki iris verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.18'deki benzetim sonucundan görüldüğü 30 adet test verisi doğru olarak sınıflandırılmıştır.

5.3.4 Haberman verisinin perseptron öğrenme algoritması ile sınıflandırılması

Haberman veri kümesi, Bölüm 4.2'de ele alınan ÇAD ve ÇD parametrelerinin perseptron öğrenme algoritması kullanılarak bulunmasından faydalanılarak sınıflandırılmıştır. Öğrenme algoritması yardımı ile ÇAD ve ÇD parametrelerinin hesaplanması için her sınıftan 42, toplamda 84 adet veri kullanılmıştır. Veri kümesi üç boyutlu ve sınıf sayısı iki olduğu için, Bölüm 4.2'de incelenen öğrenme algoritmasına göre 2 adet çift eşik hiperdüzlemi elde edilmiştir. Bu çift eşik hiperdüzlemlerinin belirlediği bölge içinde kalan veri kümesi c_1 sınıfına ait verileri, kalmayanlar ise c_2 sınıfı veri kümesi olarak değerlendirilerek sınıflandırılmıştır. Öğrenme algoritması sonucu \vec{v} vektörleri (ÇAD parametreleri, w_i 'ler) ve a, bsayıları (ÇD parametreleri I_1 ve I_2) bulunmuştur. Bu durumda c_1 sınıfına ait veriler için ilk $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörü, $a_1=11$ ve $b_1=69$ ve ikinci olarak $\vec{v}_2 = (-1.3 \ 2.2 \ -10)$ vektörü, $a_2=20$ ve $b_2=24$ değerleri elde edilmiştir. Bu değerler için oluşturulmuş Haberman sınıflandırıcısı bloğu Şekil 5.19'da verilmiştir.



Şekil 5.19 : Haberman verisi sınıflandırıcı bloğu (perseptron öğrenme algoritması sonucu oluşturulmuş).

Sınıflandırıcının başarımı için her sınıftan geri kalan 10 adet, toplamda 20 adet veri kullanılarak Şekil 5.19'daki yapıya karşılık düşen devre ile SPICE benzetimi yapılmış, kullanılan test verileri Çizelge 5.13'de gösterilen aralıklarla Şekil 5.19'daki yapıya uygulanmıştır.

| Verilerin Uygulanma Süresi | x_1 | <i>x</i> ₂ | <i>x</i> ₃ | Sınıf |
|----------------------------|-------|-----------------------|-----------------------|-----------------------|
| 0 ms - 1 ms | 34 | 60 | 1 | c_1 |
| 1 ms - 2 ms | 61 | 68 | 1 | c_2 |
| 2 ms - 3 ms | 51 | 59 | 3 | c_2 |
| 3 ms - 4 ms | 37 | 59 | 6 | c_1 |
| 4 ms - 5 ms | 54 | 58 | 1 | c_1 |
| 5 ms - 6 ms | 61 | 62 | 5 | c_2 |
| 6 ms - 7 ms | 42 | 63 | 1 | c_1 |
| 7 ms - 8 ms | 53 | 61 | 1 | c_1 |
| 8 ms - 9 ms | 48 | 67 | 7 | c_2 |
| 9 ms - 10 ms | 65 | 66 | 15 | c_2 |
| 10 ms - 11 ms | 60 | 59 | 17 | c_2 |
| 11 ms - 12 ms | 42 | 59 | 2 | c_1 |
| 12 ms - 13 ms | 30 | 62 | 3 | c_1 |
| 13 ms - 14 ms | 65 | 62 | 22 | c_2 |
| 14 ms - 15 ms | 41 | 60 | 23 | c_2 |
| 15 ms - 16 ms | 46 | 58 | 3 | c_1 |
| 16 ms - 17 ms | 42 | 61 | 4 | c_1 |
| 17 ms - 18 ms | 72 | 67 | 3 | c_1 |
| 18 ms - 19 ms | 47 | 63 | 23 | c_2 |
| 19 ms - 20 ms | 43 | 58 | 52 | <i>c</i> ₂ |

Cizelge 5.13 : Şekil 5.19'daki haberman verisi sınıflandırıcısı test kümesi.

Çizelge 5.13'de verilmiş olan Haberman verisinin özellikleri olan x_i (*i*=1,2,3) değerleri µA boyutunda olarak düşünülmüştür. Benzetim için Çizelge 5.13'deki 0-1 ms arasındaki x_1 =34 verisinin, ÇAD-1 yapısında kullanılması şu şekilde özetlenir: Şekil 3.20'deki yapıda V_y =34 mV ve R_1 =10 k Ω alınması durumunda giriş büyüklüğü V_y/R_1 = 34 µA olarak akım boyutuna dönüştürülür. Ayrıca öğrenme algoritması ile c_1 sınıfı için bulunan $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörünün ilk elemanı -1.4 olduğu için R_2/R_3 = 14 k Ω / 10 k Ω alınarak (**3.7**) ifadesine göre çıkış akımı elde edilir. Bu durumda c_1 sınıfı için bulunmuş olan a=11 ve b=69 değerleri (çekirdek devre I_1 , I_2 parametresi) kullanılarak Şekil 5.19'daki çekirdek devre-1'in kontrol akım değerleri I_1 =11 µA, I_2 =69 µA olarak alınmıştır. Ayrıca Şekil 5.19'da kullanılan ÇAD yapılarının çıkışları \vec{v} vektörünün elemanlarının pozitif ya da negatif olmasına göre seçilmiştir. Diğer bir ifadeyle $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörünün ikinci elemanı pozitif, ilk ve üçüncü elemanı negatiftir. Dolayısıyla Şekil 5.19'da kullanılan ÇAD-2 bloğu için pozitif çıkış ucu, ÇAD-1 ve ÇAD-3 bloğu için ise negatif çıkış ucu kullanılmıştır. Şekil 5.19'daki yapının çıkış akımı çekirdek devrelerinin I_H akımı ile

kontrol edilmektedir. Dolayısıyla çekirdek devrelerinin I_H akımı, uygulanan veri tipinin c_1 sınıfı olup olmadığını belirlemektedir. Şekil 5.19'da kullanılan ÇAD ve ÇD devrelerinin parametreleri c_1 sınıfı için Çizelge 5.14'de verilmiştir.

ÇAD-4 CAD-5 CAD-1 ÇAD-2 CAD-3 CD-1 CAD-6 CD-2 $R_1 = 10 \text{ k}\Omega$ $R_1 = 10 \text{ k}\Omega$ $R_1 = 10 \text{ k}\Omega$ $R_1=10 \text{ k}\Omega$ $R_1 = 10 \text{ k}\Omega$ $I_1 = 11 \ \mu A$ $R_1=10 \text{ k}\Omega$ $I_1 = 20 \, \mu A$ *I*₂=69 μA $R_2=14 \text{ k}\Omega$ $R_2=22 \text{ k}\Omega$ $R_2=160 \text{ k}\Omega$ $R_2=13 \text{ k}\Omega$ $R_2=22 \text{ k}\Omega$ $R_2=100 \text{ k}\Omega$ *I*₂=24 μA

 $R_3=10 \text{ k}\Omega$

 $R_3=10 \text{ k}\Omega$

 $R_3=10 \text{ k}\Omega$

 $I_{H}=10 \ \mu A$

 $I_H = 10 \ \mu A$

 $R_3=10 \text{ k}\Omega$

 $R_3=10 \text{ k}\Omega$

 $R_3=10 \text{ k}\Omega$

Çizelge 5.14 : Şekil 5.19'da c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

| Benzetim için kullanılan test verileri Çizelge 5.13'de gösterilen aralıklar ile Şekil |
|--|
| 5.19'daki sınıflandırıcıya uygulanmıştır. Çizelge 5.13'de sonuçların ne çıkması |
| gerektiği gözükmektedir. Şekil 5.19'daki yapının benzetim sonucu da Şekil 5.20'de |
| verilmiştir. Buna göre çıkış değerinin 10 μ A olması durumunda sonuç c_1 , 0 μ A |
| olması durumundaysa sonuç c_2 olarak değerlendirilmiştir. |



Şekil 5.20 : Şekil 5.19'daki Haberman verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.20'deki benzetim sonucundan görüldüğü 20 adet Haberman test verisi doğru olarak sınıflandırılmıştır. İris ve Haberman verileri önerilen iki yöntem ile sınıflandırılmıştır. Sınıflandırıcı başarımları Çizelge 5.15'de verilmiştir. Önerilen algoritmalar ile sınıflandırıcı parametreleri Pentium 3 GHz işlemcili ve 1 GB belleği olan bir bilgisayar yardımı ile gerçekleştirilmiştir.

| Algoritma | Sınıflandırıcı | hata başarımı [%] | Algoritmanın sınıflandırıcı parametrelerini bulma süresi [sn] | | | | |
|-----------------------|----------------|-------------------|--|----------|--|--|--|
| _ | İris | Haberman | İris | Haberman | | | |
| Fisher tabanlı | 100 | 100 | 0.4 | 0.3 | | | |
| Perseptron öğrenme | 100 | 100 | 220 | 190 | | | |

Çizelge 5.15 : Sınıflandırıcı başarımları karşılaştırması.

Çizelge 5.15'den görüldüğü gibi her iki yöntem ile de veri sınıflandırılabilmektedir. Sınıflandırıcı algoritmasının kontrol parametrelerini bulması konusunda, Fisher tabanlı algoritmanın daha iyi olduğu gözlenmektedir. Ancak iki yöntem içinde kullanılan ÇD sayısının aynı olduğu görülmüştür.

Dolayısıyla gerçekleştirilmiş olan sınıflandırıcı devreleri ile literatürdeki veri kümelerinin başarılı bir şekilde sınıflandırıldığı, ayrıca sınıflandırma işlemi dışındaki diğer çeşitli uygulamalar için de kullanılabilecekleri ve yeni olanaklar ortaya çıkardıkları gösterilmiştir.

6. SONUÇLAR VE ÖNERİLER

Bu çalışmada, kuantalama, görüntü işleme, istatistik, tıbbi tanı, otomatik hedef belirleme, yapay zekâ, YSA, analog-sayısal dönüştürücüler gibi uygulama alanı bulan sınıflandırıcı devreler ve yeni olanakları ele alınmıştır. Literatürde rastlanan devrelerinin gerek güç tüketimi, gerek dışarıdan ayarlanabilme yeteneğine sahip olma gibi bazı sınıflandırma konularında istenildiği biçimde gereksinimleri karşılayamadığı saptanmıştır. Çalışmada, bu eksikliklerin bir kısmını giderecek ve mevcut literatüre katkı sağlayacak yeni devreler geliştirilmiş, bu devreler bazı gerçek uygulama yapıları ve benzetimler ile test edilip özgün sonuçlara ulaşılmıştır.

Bu çalışma kapsamında yapılanları şöyle sıralayabiliriz:

Gerek dışarıdan ayarlanabilme yeteneği, gerekse uygulama alanlarının esnek bir şekilde kullanılmalarını sağlamak amacıyla önce temel bir sınıflandırıcı devre bloğuna işlevsel olarak karar verilmiş ve bu işlevi gerçekleyen iki adet CMOS devre önerilmiştir. Temel bir yapı niteliğinde olan bu sınıflandırıcı işlevsel bloğa çekirdek devre ismi verilmiştir. Bu çekirdek devrenin önemi temel bir yapı taşı gibi çeşitli topolojiler de kullanılabilir olmasıdır. Ayrıca, bu çekirdek devre literatürdeki mevcut yapılara göre çok daha esnek ve daha az güç tüketecek şekilde tasarlanmıştır [54]. Çekirdek devre blokları, akım modlu tasarlanmış olmalarından dolayı, bloklar uygun şekilde paralel ya da ard arda bağlanabilmektedir. Dolayısıyla tek boyutlu sınıflandırıcı yapılarından çok boyutlu sınıflandırıcı yapıların kolayca oluşturulmasına olanak tanınmıştır. Bu çok boyutlu sınıflandırıcı devreler ile dikdörtgen ızgara veri kümelerinin sınıflandırılacağı da gösterilmiştir ki bu tür veriler en yoğun kullanılanların başında gelmektedir. Özellikle, bazı veri kümeleri için çekirdek devrenin esnek yapıda tasarlanmasının getirdiği kullanım kolaylıkları da avantajları arasındadır. Ayrıca bahsedilen temel çekirdek yapısının, sınıflandırma bölgelerinin incelendiği ve literatürdeki mevcut sınıflandırma yapılarıyla karşılaştırılarak oldukça daha az güç harcadığı gösterilmiştir [58].

95

Diğer taraftan geliştirilen çok boyutlu sınıflandırıcı sadece dikdörtgen ızgara dağılımına sahip verilere uygulanabildiğinden, uygulanma alanlarını genişletebilmek amacıyla çekirdek devrelerin önüne girişin lineer kombinasyonunu alan yeni bir kat eklenmiş ve çarpan devre denilmiştir. Söz konusu ön kat, çok yoğun kullanılan bazı veri dağılımlarının (örneğin CMOS tranzistorların standart bölgeleri) sınıflandırılmasını da sağlamıştır [63]. Çarpan devrenin kontrol parametrelerini de kapsayacak şekilde sınıflandırma işlemi için gerekli parametrelerinin bulunmasını sağlayan algoritmalar da geliştirilmiş, üstünlükleri ve sakıncaları incelenmiştir.

Gerçekleştirilmiş olan sınıflandırıcı yapıların başarımlarına, gerçek dünyadaki uygulama alanlarına ve sağladığı yeni olanaklara örnek olarak kuantalayıcı devresi incelenmiştir [55]. Bunun yanı sıra, şablon uydurma yönteminin kullanıldığı ve bünyesinde oluşacak hataların düzeltilmesine de olanak tanıyan karakter tanıma uygulaması da gerçekleştirilmiştir [71].

Ayrıca sınıflandırıcı devrelerin, iki farklı öğrenme algoritması kullanılarak iki farklı veri kümesini sınıflandıran devrelerin parametreleri bulunmuş; benzetimlerle başarımları gösterilmiştir. Sonuçlar devrelerin ve algoritmaların etkinliğini ve başarımını göstermesi için karşılaştırmalı olarak verilmiştir.

Sonuç olarak, bu çalışmada gerçekleştirilmiş olan özgün sınıflandırıcı devrelerinin literatüre ve uygulamalara yeni olanaklar tanıdığı gösterilmiştir.

Bu tezin devamı niteliğinde dört farklı alanda yapılabilecek çalışmalardan ilki, daha kısa sürede eğitimin sağlanabileceği yeni öğrenme algoritmalarını geliştirmektir. İkincisi ise donanımsal olarak gerçekleştirilen devrelerde güç tüketimi ve kullanım kolaylığı açısından gelişme sağlanmasıdır. Üçüncü bir geliştirme, verileri ayıran daha genel bölgelerin sınıflandırılmasına olanak tanıyan bir ön kat ya da çekirdek hücrenin tasarımının yapılmasıdır. Dördüncüsü ve belki de en önemlisi, yazılımsal olarak geliştirilen algoritmaların donanımın içine dahil edilmesi, başka bir deyişle öğrenme sürecinin de donanımın içinde yapılmasıdır.

KAYNAKLAR

- [1] Liu, B., Chen, C., and Tsao, J., 2000: A Modular Current-Mode Classifier Circuit for Template Matching Application, *IEEE Trans. On Circuit* and Systems-II, Analog and Digital Sig. Process., Vol. 47, pp. 145-151.
- [2] **Tzanakou, E. M.,** 2000: *Supervised and unsupervised Pattern Recognition*, CRC Press, Inc., pp. 7-75, Boca Raton, FL, USA.
- [3] Cover, T. M., and Hart, P. E., 1967: Nearest neighbor pattern classification, *IEEE Trans. on Inform. Theory*, Vol. 13, pp. 21–27.
- [4] **Bishop, C. M.,** 1996: *Neural Networks for Pattern Recognition*, Oxford University Press, Inc., pp. 116-191, Newyork, USA.
- [5] **Beiu, V.,** 1996: *Neural Networks and Their Application*, John Wiley & Sons Ltd., Baffins Lane, Chichester, pp. 1-98, England.
- [6] Duda, R. O., Stork, D. G., and Hart, P. E., 2000: Pattern Classification and Scene Analysis, John Wiley & Sons Ltd., Inc., 2nd ed., pp. 1-57, England.
- [7] Jain, A. K., Duin, R. P. W., and Mao, J., 2000: Statistical Pattern Recognition: A Review, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 22, pp. 4-37.
- [8] Chenyz, Y., Hungyz, Y., and Fuhz, C., 2001: Fast Algorithm for Nearest Neighbor Search Based on a Lower Bound Tree, *Proceedings of the* 8th International Conference on Computer Vision, Vancouver, Canada, July 7-14, pp. 446-453.
- [9] Abdel-Aty-Zohdy, H. S., and Al-Nsour, M., 1999: Reinforcement learning neural network circuits for electronic nose, *IEEE International Symposium on Circuits and Systems*, Florida, USA, May 30- July 2, pp. 379- 382.
- [10] **Ripley, B. D.,** 1996: *Pattern Recognition and Neural Networks*, Cambridge University Press, pp. 143-177, Cambridge, UK.
- [11] Yamasaki, T., Yamamoto, K., and Shibata, T., 2001: Analog pattern classifier with flexible matching circuitry based on principal-axisprojection vector representation, *Proceedings of the 27th European Solid-State Circuits Conference*, Villach, Austria, September 18-20, pp. 197-200.

- [12] Yamasaki, T., and Shibata, T., 2003: Analog soft-pattern-matching classifier using floating-gate MOS technology, *IEEE Trans. on Neural Networks*, Vol. 14, pp. 1257-1265.
- [13] Reyes, C., and Adjouadi, M., 1995: A Clustering Technique for Random Data Classification, *IEEE International Conference on Systems, Man and Cybernetics*, Vancouver, Canada, December 22-25, pp. 316-321.
- [14] Zhao, Y., Deng, B., and Wang, Z., 2002: Analysis and Study of Perceptron to Solve XOR Problem, *Proceeding of the 2th International Workshop* on Autonomous Decentralized System, China, November 6-7, pp. 168-173.
- [15] McCulloch, W. S., and Pitts, W., 1943: A logical calculus of the immanent in nervous activity, *Bulletion of Mathematical Biophysics*, Vol. 5, pp. 115-133.
- [16] Galushkin, A.I., and Sudarikov, V.A., 1992: Adaptive neural network algorithm for solving linear algebra problems, *IEEE Symposium on Neuroinformatics and Neurocomputers*, Rostov, Russia, October 7-10, pp. 128 – 138.
- [17] Moody. J., and Darken C., 1989: Fast Learning in Networks of Locally-Tuned Processing Units, *Neural Computation*, Vol. 1, pp. 281-294.
- [18] Xiaoqin, Z., Yajuan, H., and Yeung, D. S., 2003: Determining the relevance of input features for multilayer perceptrons, *IEEE International Conference on Systems Man ans Cybernetics*, October 5-8, pp. 874-879.
- [19] Sun, P., and Makro K., 1997: A Neural Network Classifier For Conflicting Information Environments, *International Conference on Neural Networks*, Stocholm, Sweden, June 9-12, pp. 1617-1622.
- [20] Hung, Y. C., and Liu B. D., 1999: A CMOS Analog Vector Quantizer For Pattern Recognition, *The First IEEE Asia Pacific Conference on* ASICs, Korea, August 23-25, pp. 112-115.
- [21] Zhou, Z., Chen, S., and Chen, Z., 2000: FANNC: A Fast Adaptive Neural Network Classifier, *Knowledge and Informaton Systems*, Vol. 2, pp. 115-129.
- [22] Camboni, F., and Valle, M., 2001: A Mixed Mode Perceptron Cell for VLSI Neural Networks, *International Conference on Electronics*, *Circuits and Systems*, Malta, September 2-5, pp. 377-380.
- [23] Jing, Y., 2006: An Improved Cascade SVM Training Algorithm with Crossed Feedbacks, *First International Multi-Symposiums Computer and Computational Sciences*, Istanbul, Turkey, June 20-24, pp. 735-738.

- [24] Oyang, Y. J., Hwang S. C., Ou, Y.Y., Chen, C. Y., and Chen, Z. W., 2005: Data Classification with Radial Basis Function Networks based on a Novel Kernel Density Estimation Algorithm, *IEEE Transactions on Neural Networks*, Vol. 16, pp. 225-236.
- [25] Lubkin, J., and Cauwenberghs G., 1998: A Micropower Learning Vector Quantizer for Parallel Analog-to-Digital Data Compression, *Proceedings of the International Symposium on Circuits and Systems*, Vol 3, May 31-June 3, pp. 58-61.
- [26] Kachare, M., Carvajel, J. R. A., and Lopez, M. A. J., 2005: New Low-Voltage Fully Programmable CMOS Triangular/Trapezoidal Function Generator Circuits, *IEEE Trans. On Circuit and Systems-I*, Vol. 52, pp. 2033-2040.
- [27] Bilgili, E., Göknar, İ. C., and Uçan, O. N., 2005: Cellular Neural Networks with Trapezoidal Activation Function, *Int. Journal of Circuit Theory* and Applications, Vol. 33, pp. 393-417.
- [28] Aksın, D. Y., Aras, S., and Göknar, İ. C., 2000: CMOS Realization of User Programmable, Single-Level, Double-Threshold Generalized Perceptron, Proceedings of Turkish Artificial Intelligence and Neural Networks Conference, İzmir, Turkey, July 21-23, pp. 117-125.
- [29] Bilgili, E., Göknar, C., Uçan, O. N., and Albora, M., 2006: Stability of CNN with Trapezoidal Activation Function, Complex Computing-Networks: Brain-like and Wave-oriented Electrodynamic Algorithms Springer Proceedings in Physics, Vol. 104, pp. 267-274.
- [30] Gray, R. M., and Neuhoff, D. L., 1998: Quantization, *IEEE Trans. on Information Theory*, Vol. 44, pp. 2325-2383.
- [31] Leong, P. H. W., and Jabri, M. A., 1995: A Low Power VLSI Arrhytmia Classifier, *IEEE Transaction on Neural Networks*, Vol. 6, pp. 1435-1445.
- [32] Aybay, I., Çetinkaya, S., and Halıcı, U., 1996: Classification of Neural Networks Hardware, *Neural Networks World*, Vol. 6, pp. 11-29.
- [33] Erkmen, B., 2007: Genel Amaçlı Bir Yapay Sinir Ağının Karma Bir Donanım ile Gerçeklenmesi, *PhD Thesis*, Yıldız Teknik Üniversitesi, İstanbul.
- [34] Avcı, M., 2005: Hibrit Bir Donanımla Yapay Sinir Ağı Gerçekleme AY Tümdevresi, *PhD Thesis*, Yıldız Teknik Üniversitesi, İstanbul.
- [35] Cilingiroglu, U., 1993: A charge based neural Hamming Classifier, *IEEE Journal of Solid-State Circuits*, Vol. 28, pp. 59-67.
- [36] Aksin, D. Y., and Aras, S., 2005: A compact Distance Cell for Analog Classifiers, Proceedings of the IEEE International Symposium on Circuits and Systems, Kobe, Japan, May 23-26, pp. 3627-3630.

- [37] Donckers, N., Dualibe, C., and Verleysen, M., 1999: Design of Complementary Low-Power CMOS Architectures for Looser-take all and Winner-take all, *Proceedings of the Seventh International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems*, Granada, Spain, April 7-9, pp. 360-365.
- [38] Cilingiroglu, U., and Aksın, D. Y., 1998: A 4 transistor Euclidean Distance Cell for Analog Classifiers, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Monterey, USA, May 31- June 3, pp. 84-87.
- [39] Grant, D., Taylor, J., and Houselander, P., 1994: A High-Speed Integrated Hamming Neural Classifier, *Proceedings of the IEEE International Symposium on Circuits and Systems*, London, England, May 30 –June 2, pp. 479-482.
- [40] Kumar, N., Pouliquen, P. O., and Andreou, A. G., 1993: Device Mismatch Limitations on Performance of a Hamming Distance Classifier, *The IEEE International Workshop on Defect and Fault Tolerance in VLSI System*, Venice, Italy, October 27-29, pp. 327-334.
- [41] Onat, B. M., McNeil, J. A., and Çilingiroğlu, U., 1997: Implementation of a charge based Neural Euclidian Classifier for a 3 bit flash Analog to digital converter, *IEEE Transactions on Instrumentation and Measurement*, Vol. 46, pp. 672-677.
- [42] Draghici, S., and Miller, D. A., 1999: A VLSI Neural Network Classifier Based on Integer-Valued Weights, *International Joint Conference on Neural Networks*, Washington, USA, July 10-16, pp. 2419-2424.
- [43] Almeida, A. P., and Franca, J. E., 1993: A mixed-mode architecture for implementation of analog neural networks with digital programmability, *Proceedings of the International Joint Conference* on Neural Networks, Nagoya, Japan, October 25-29, pp. 887-890.
- [44] Genov, R., and Cauwenberghs, G., 2003: Kerneltron: Support Vector Machine in Silicon, *IEEE Transactions on Neural Networks*, Vol. 14, pp. 1426-1434.
- [45] Chakbarrtty, S., and Cauwenberghs, G., 2007: Sub-Microwatt Analog VLSI Trainable Pattern Classifier, *IEEE Journal of Solid-State Circuits*, Vol. 42, pp. 1169-1178.
- [46] Peng, S. Y., Minch, B. A., and Hasler, P., 2008: Analog VLSI Implementation of Support Vector Machine Learning and Classification, *Proceedings* of the IEEE International Symposium on Circuits and System, Washington, USA, May 18-21, pp. 860-863.
- [47] Peng, S. Y., Hasler, P. E., and Anderson, D., 2007: An Analog Programmable Multi-Diemensional Radial Basis Function Based Classifier, *International Conference on Very Large Scale Integration*, Atlanta, USA, October 15-17, pp. 13-18.

- [48] Narayan, R., Honbo, D., Memik, G., Choudhary, A., and Zambreno, J., 2007: An FPGA Implementation of Decision Tree Classification, Design, Automation & Test in Europe Conference & Exhibition, Yokohama, Japan, April 16-20, pp. 1-6.
- [49] Reaz, M. B. I., Islam, S. Z., Ali, M. A. M., and Sulaiman, M. S., 2002: FPGA Realization of Backpropagation for Stock Market Prediction, *Proceedings of the 9th International Conference on Neural Information*, Sofia, Bulgaria, Vol. 2, November 18-22, pp. 960-964.
- [50] Wang, Y., and Salam, F. M. A., 1991: Experiments Using CMOS Neural Networks Chips as Pattern/Character Recognizers, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Singapore, June 11-14, pp. 1196-1199.
- [51] Bridges, S., Holleman, J., and Figueroa. M., 2006: A Random Projection Imager for Visual Pattern Classification in Analog VLSI, *Proceedings* of the 32nd European Solid-State Circuits Conference, Montreux, Switzerland, September 18-22, pp. 428-431.
- [52] Bermak, A., and Martinez, D., 2003: A Compact 3-D VLSI Classifier Using Bagging Threshold Network Ensembles, *IEEE Transaction on Neural Networks*, Vol. 14, pp. 1097-1109.
- [53] Djahanshahi, H., Jullien, G. A., Miller, W.C., and Ahmadi, M., 1996: Neural-Based Smart CMOS Sensors for On-Line Pattern Classification Application, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Atlanta, USA, May 12-15, pp. 384-387.
- [54] Yıldız, M., Minaei, S., and Göknar, C., 2007: A CMOS Classifier Circuit using Neural Networks with Novel Architecture, *IEEE Transaction on Neural Networks*, Vol. 18, pp. 1845-1849.
- [55] Yıldız, M., Minaei, S., and Göknar, C., 2006: CMOS Realization of a Quantized-Output Classifier Circuit, 13th IEEE International Conference on Electronics, Circuits and Systems, Nice, France, December 10-13, pp. 292-295.
- [56] Morgül, A., and Temel, T., 2005: Current-mode level restoration circuit for multi-valued logic, *Electronics Letters*, Vol. 41, pp. 230-231.
- [57] Hodges D., Jackson H., and Saleh R., 2004: Analysis and Design of Digital Integrated Circuits, Third edition, McGraw Hill, pp. 66-69, Newyork, USA.
- [58] Yıldız, M., Minaei, S., and Göknar, C., 2007: A Low-Power Multilevel-Output Classifier Circuit, *European Conference on Circuit Theory* and Design, Seville, Spain, August 26-30, pp. 747-750.

- [59] Leong, P. H. W., 1993: Kakadu-A Low Power Analogue Neural Network Classifier, *International Journal of Neural Systems*, Vol. 4, pp. 381-394.
- [60] Lu, C., Shi, B., and Chen, L., 2002: A General Purpose Neural Network with on Chip BP Learning, *International Symposium on Circuits and System*, Arizona, USA, May 16-29, pp. 520-523.
- [61] Gatt, E., and Micallef, J., 2002: Analogue Radial Basis Function Networks for Phoneme Recognition, *International Conference on Electronics, Circuits, and Systems*, Dubrovnik, Croatia, September 15-18, pp. 583-586.
- [62] Zeng, H., Ying, H., and Xingbo, S., 2000: A fuzzy central cluster neural classifier, *Proceedings of the 3rd World Congress on Intelligent Control and Automation*, Hefei, China, June 28–July 2, pp. 1747-1750.
- [63] Yıldız, M., Minaei, S., and Özoğuz, S., 2009: Linearly Weighted Classifier Circuit, *Northeast Workshop on Circuits and Systems*, konferans kitabında basılmak üzere kabul edilmiştir.
- [64] Qian, D., 2007: Modified Fisher's Linear Discriminant Analysis for Hyperspectral Imagery, *IEEE Geoscience and Remote Sensing Letters*, Vol. 4, pp. 503-507.
- [65] Ferri, G., and Guerrini, N., 2001: High valued passive element simulation using low-voltage low- power current conveyor for fully integrated applications, *IEEE Transactions on Circuit and Systems II*, Vol. 48, pp. 405-409
- [66] Qi, L., and Donald, W. T., 1997: Principal Feature Classification, *IEEE Transaction on Neural Networks*, Vol. 8, pp. 155-160.
- [67] Jiang, G., Xiaoqing D., and Youshou W., 1999: On improvement of Multiple Discriminant Analysis Method for Discriminative Feature Extraction, *Systems, Man, and Cybernetics, Conference Proceedings*, October 12-15, pp. 915 – 920
- [68] Çevikalp, H., 2006: Theoretical Analysis of Linear Discriminant Analysis Criteria, *IEEE 14th Signal Processing and Communications Applications*, Antalya, Turkey, April 17-19, pp. 1-4.
- [69] Genç, İ., and Güzeliş, C., 1998: Threshold Class CNNs with Input-Dependent Initial State, IEEE International Workshop on Cellular Neural Networks and their Applications, London, England, April 14-17, pp. 130-135
- [70] Chou, C. H., and Chen, Y. C., 1989: A VLSI Architecture for real-time and flexible image template matching, *IEEE Trans. on Circuit and Systems*, Vol. **36**, pp. 1336-1342.

- [71] Yıldız, M., Minaei, S., and Göknar, C., 2008: Realization and Template Matching Application of a CMOS Classifier Circuit, *IEEE Applied Electronics*, Pilsen, Czech Republic, September 10-11, pp. 231-234.
- [72] Fisher, R.A., 1936: The use of multiple measurements in taxonomic problems, *Annual Eugenics*, Vol. 7, pp. 179-188.
- [73] Haberman, S.J., 1976: Generalized Residuals for Log-Linear Models, Proceedings of the 9th International Biometrics Conference, Boston, USA, pp. 104-122.

EKLER

EK A : AMS 0.35 μ m MOSIS SPICE teknoloji parametreleri EK B : Maksimum(rank(S_B)=c-1'in ispatı EK C.1 : İris veri kümesi EK C.2 : Haberman veri kümesi

EK A

Bu ekte benzetimlerde kullanılan NMOS ve PMOS tranzistorların 0.35 µm AMS CMOS SPICE teknoloji parametreleri verilmiştir.

NMOS:

(LEVEL=7 MOBMOD =1.000e+00 CAPMOD =2.000e+00 NLEV=0 K1=6.044e-01 K2=2.945e-03 K3=-1.72e K3B=6.325e-01 NCH=2.310e+17 VTH0=4.655e-01 VOFF=-5.72e-02 DVT0=2.227e+01 DVT1=1.051e DVT2=3.393e-03 KETA=-6.21e-04 PSCBE1=2.756e+08 PSCBE2=9.645e-06 DVT0W=0.000 DVT1W=0.000 DVT2W=0.000 UA=1.000e-12 UB=1.723e-18 UC=5.756e-11 U0=4.035e+02 DSUB=5.000e-01 ETA0=3.085e-02 ETAB=-3.95e-02 NFACTOR=1.119e-01 EM=4.100e+07 PCLM=6.831e-01 PDIBLC1=1.076e-01 PDIBLC2=1.453e-03 DROUT=5.000e-01 A0=2.208e+00 A1=0.000 A2=1.000 PVAG=0.000 VSAT=1.178e+05 AGS=2.490e-01 B0=-1.76e-08 B1=0.000 DELTA=1.000e-02 PDIBLCB=2.583e-01 W0=1.184e-07 DLC=8.285e-09 DWC=2.676e-08 DWB=0.000 DWG=0.000 LL=0.000 LW=0.000 LWL=0.000 LLN=1.000 LWN=1.000e WL=0.000 WW=0.000 WWL=0.000 WLN=1.000e WWN=1.000 AT=3.300e+04 UTE=-1.80e KT1=-3.30e-01 KT2=2.200e-02 KT1L=0.000 UA1=0.000 UB1=0.000 UC1=0.000 PRT=0.000 CGDO=2.100e-10 CGSO=2.100e-10 CGBO=1.100e-10 CGDL=0.000 CGSL=0.000 CKAPPA =6.000e-01 CF=0.000 ELM=5.000e XPART=1.000e CLC=1.000e-15 CLE=6.000e-01 RDSW=6.043e+02 CDSC=0.000 CDSCB=0.000 CDSCD=8.448e-05 PRWB=0.000 PRWG=0.000 CIT=1.000e-03 TOX=7.700e-09 NGATE=0.000e NLX=1.918e-07 ALPHA0=0.000 BETA0=3.000e+01 AF=1.400 KF=2.810e-27 EF=1.000 NOIA=1.000e+20 NOIB=5.000e+04 NOIC=-1.40e-12 LINT=-1.67e-08 WINT=2.676e-08 XJ=3.000e-07 RSH=8.200e+01 JS=2.000e-05 CJ=9.300e-04 CJSW=2.800e-10 MJ=3.100e-01 MJSW=1.900e-01 PB=6.900e-01 TT=0.000e+00 PBSW=9.400e-01)

PMOS:

(LEVEL=7 MOBMOD =1.000e CAPMOD=2.000e NLEV=0 K1=5.675e-01 K2=-4.39e-02 K3=4.540e K3B=-8.52e-01 NCH=1.032e+17 VTH0=-6.17e-01 VOFF=-

1.13e-01 DVT0=1.482e+00 DVT1=3.884e-01 DVT2=-1.15e-02 KETA=-2.56e-02 PSCBE1=1.000e+09 PSCBE2=1.000e-08 DVT0W=0.000 DVT1W=0.000 DVT2W=0.000 UA=2.120e-10 UB=8.290e-19 UC=-5.28e-11 U0=1.296e+02 DSUB=5.000e-01 ETA0=2.293e-01 ETAB=-3.92e-03 NFACTOR=8.237e-01 EM=4.100e+07 PCLM=2.979e+00 PDIBLC1=3.310e-02 PDIBLC2=1.000e-09 DROUT =5.000e-01A0=1.423e A1 = 0.000A2=1.000 PVAG=0.000 VSAT=2.000e+05 AGS=3.482e-01 B0=2.719e-07 B1=0.000 DELTA=1.000e-02 PDIBLCB=-1.78e-02 W0=4.894e-08 DLC=-5.64e-08 DWC=3.845e-08 DWB=0.000 DWG=0.000 LL=0.000 LW=0.000 LWL=0.000 LLN=1.000 LWN=1.000 WL=0.000 WW=0.000 WWL=0.000 WLN=1.000 WWN=1.000 AT=3.300e+04 UTE=-1.35e+00 KT1=-5.70e-01 KT2=2.200e-02 KT1L=0.000 UA1=0.000 UB1=0.000 UC1=0.000 PRT=0.000 CGDO=2.100e-10 CGSO=2.100e-10 CGBO=1.100e-10 CGDL=0.000 CGSL=0.000 CKAPPA =6.000e-01 CF=0.000 ELM=5.000 XPART=1.000 CLC=1.000e-15 CLE=6.000e-01 RDSW=1.853e+03 CDSC=6 994e-04 CDSCB=2 943e-04 CDSCD=1 970e-04 PRWB=0.000 PRWG=0.000 CIT=1.173e-04 TOX=7.700e-09 NGATE=0.000 NLX=1.770e-07 ALPHA0=0.000 BETA0=3.000e+01 AF=1.290e KF=1.090e-27 EF=1.000e NOIA=1.000e+20 NOIB=5.000e+04 NOIC=-1.40e-12LINT=-8.14e-08 WINT=3.845e-08 XJ=3.000e-07 RSH=1.560e+02 JS=2.000e-05 CJ=1.420e-03 CJSW=3.800e-10 MJ=5.500e-01 MJSW=3.900e-01 PB=1.020e+00 TT=0.000e PBSW=9.400e-01)

EK B

Bu ekte, maksimum(rank(S_B)=c-1' olduğu ispatlanmıştır.

$$S_{B} = n_{1} \left(\vec{\mu}_{1} - \frac{1}{c} \left(\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c} \right) \right) \left(\vec{\mu}_{1} - \frac{1}{c} \left(\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c} \right) \right)^{T} +$$

$$n_{2} \left(\vec{\mu}_{2} - \frac{1}{c} \left(\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c} \right) \right) \left(\vec{\mu}_{2} - \frac{1}{c} \left(\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c} \right) \right)^{T} + ... +$$

$$n_{c} \left(\vec{\mu}_{c} - \frac{1}{c} \left(\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c} \right) \right) \left(\vec{\mu}_{c} - \frac{1}{c} \left(\vec{\mu}_{1} + \vec{\mu}_{2} + ... + \vec{\mu}_{c} \right) \right)^{T}$$

$$S_{B} = \frac{n_{1}}{c^{2}} \left((c-1)\vec{\mu}_{1} - \vec{\mu}_{2} - ... - \vec{\mu}_{c} \right) ((c-1)\vec{\mu}_{1} - \vec{\mu}_{2} - ... - \vec{\mu}_{c})^{T} +$$

$$\frac{n_{2}}{c^{2}} \left(-\vec{\mu}_{1} + (c-1)\vec{\mu}_{2} - ... - \vec{\mu}_{c} \right) \left(-\vec{\mu}_{1} + (c-1)\vec{\mu}_{2} - ... - \vec{\mu}_{c} \right)^{T} + ... +$$

$$(B.2)$$

 S_B matrisinin ifadesinde $\Im = \left[\vec{\mu}_1 \ \vec{\mu}_2 \dots \vec{\mu}_c\right]$ tanımı kullanılırsa, toplam üç matrisin çarpımı biçiminde aşağıda gösterildiği şekilde ifade edilebilir.

(B.3) ifadesinde ortadaki toplamda yer alan matrislerin hepsinin satırları aynı lineer bağımlılığı (satırların toplamları sıfır ediyor) sağladığından toplam matrisin dolayısıyla da S_B matrisinin rankı *c*-*l* sayısından büyük olamaz.

EK C.1

| Canak Yanrağı | Canak Yaprağı | Tac Yanrağı | Tac Yaprağı | |
|---------------|----------------|---------------|----------------|--------|
| Uzunluğu [cm] | Genisliği [cm] | Uzunluğu [cm] | Genisliği [cm] | Sinif |
| 4.30 | 2.30 | 1.40 | 0.20 | Setosa |
| 4.40 | 2.90 | 1.40 | 0.20 | Setosa |
| 4.40 | 3.00 | 1.30 | 0.20 | Setosa |
| 4.40 | 3.00 | 1.50 | 0.20 | Setosa |
| 4.50 | 3.00 | 1.40 | 0.20 | Setosa |
| 4.60 | 3.00 | 1.70 | 0.40 | Setosa |
| 4.60 | 3.00 | 1.40 | 0.30 | Setosa |
| 4.60 | 3.00 | 1.50 | 0.20 | Setosa |
| 4.60 | 3.10 | 1.40 | 0.20 | Setosa |
| 4.70 | 3.10 | 1.50 | 0.10 | Setosa |
| 4.70 | 3.10 | 1.50 | 0.20 | Setosa |
| 4.80 | 3.10 | 1.60 | 0.20 | Setosa |
| 4.80 | 3.10 | 1.40 | 0.10 | Setosa |
| 4.80 | 3.20 | 1.10 | 0.10 | Setosa |
| 4.80 | 3.20 | 1.20 | 0.20 | Setosa |
| 4.80 | 3.20 | 1.50 | 0.40 | Setosa |
| 4.90 | 3.20 | 1.30 | 0.40 | Setosa |
| 4.90 | 3.20 | 1.40 | 0.30 | Setosa |
| 4.90 | 3.30 | 1.70 | 0.30 | Setosa |
| 4.90 | 3.30 | 1.50 | 0.30 | Setosa |
| 5.00 | 3.40 | 1.70 | 0.20 | Setosa |
| 5.00 | 3.40 | 1.50 | 0.40 | Setosa |
| 5.00 | 3.40 | 1.00 | 0.20 | Setosa |
| 5.00 | 3.40 | 1.70 | 0.50 | Setosa |
| 5.00 | 3.40 | 1.90 | 0.20 | Setosa |
| 5.00 | 3.40 | 1.60 | 0.20 | Setosa |
| 5.00 | 3.40 | 1.60 | 0.40 | Setosa |
| 5.00 | 3.40 | 1.50 | 0.20 | Setosa |
| 5.10 | 3.40 | 1.40 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.60 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.60 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.50 | 0.40 | Setosa |
| 5.10 | 3.50 | 1.50 | 0.10 | Setosa |
| 5.10 | 3.50 | 1.40 | 0.20 | Setosa |
| 5.10 | 3.50 | 1.50 | 0.10 | Setosa |
| 5.10 | 3.60 | 1.20 | 0.20 | Setosa |
| 5.20 | 3.60 | 1.30 | 0.20 | Setosa |
| 5.20 | 3.70 | 1.50 | 0.10 | Setosa |
| 5.20 | 3.70 | 1.30 | 0.20 | Setosa |
| 5.30 | 3.70 | 1.50 | 0.20 | Setosa |
| 5.40 | 3.80 | 1.30 | 0.30 | Setosa |
| 5.40 | 3.80 | 1.30 | 0.30 | Setosa |
| 5.40 | 3.80 | 1.30 | 0.20 | Setosa |
| 5.40 | 3.80 | 1.60 | 0.60 | Setosa |
| 5.40 | 3.90 | 1.90 | 0.40 | Setosa |
| 5.50 | 3.90 | 1.40 | 0.30 | Setosa |
| 5.50 | 4.00 | 1.60 | 0.20 | Setosa |
| 5.70 | 4.10 | 1.40 | 0.20 | Setosa |
| 5.70 | 4.20 | 1.50 | 0.20 | Setosa |

Çizelge C.1 : İris veri kümesi.

| Çanak Yaprağı | Çanak Yaprağı | Taç Yaprağı | Taç Yaprağı | 6£ |
|---------------|----------------|---------------|----------------|-------------|
| Uzunluğu [cm] | Genişliği [cm] | Uzunluğu [cm] | Genişliği [cm] | Sinii |
| 5.80 | 4.40 | 1.40 | 0.20 | Setosa |
| 4.90 | 2.00 | 4.70 | 1.40 | Versicolour |
| 5.00 | 2.20 | 4.50 | 1.50 | Versicolour |
| 5.00 | 2.20 | 4.90 | 1.50 | Versicolour |
| 5.10 | 2.30 | 4.00 | 1.30 | Versicolour |
| 5.20 | 2.30 | 4.60 | 1.50 | Versicolour |
| 5.40 | 2.30 | 4.50 | 1.30 | Versicolour |
| 5.50 | 2.40 | 4.70 | 1.60 | Versicolour |
| 5.50 | 2.40 | 3.30 | 1.00 | Versicolour |
| 5.50 | 2.40 | 4.60 | 1.30 | Versicolour |
| 5.50 | 2.50 | 3.90 | 1.40 | Versicolour |
| 5.50 | 2.50 | 3.50 | 1.00 | Versicolour |
| 5.60 | 2.50 | 4.20 | 1.50 | Versicolour |
| 5.60 | 2.50 | 4.00 | 1.00 | Versicolour |
| 5.60 | 2.60 | 4.70 | 1.40 | Versicolour |
| 5.60 | 2.60 | 3.60 | 1.30 | Versicolour |
| 5.60 | 2.60 | 4.40 | 1.40 | Versicolour |
| 5.70 | 2.70 | 4.50 | 1.50 | Versicolour |
| 5 70 | 2 70 | 4 10 | 1 00 | Versicolour |
| 5 70 | 2 70 | 4 50 | 1.50 | Versicolour |
| 5 70 | 2 70 | 3 90 | 1 10 | Versicolour |
| 5 70 | 2.70 | 4 80 | 1.80 | Versicolour |
| 5.80 | 2.80 | 4 00 | 1 30 | Versicolour |
| 5.80 | 2.80 | 4 90 | 1.50 | Versicolour |
| 5.80 | 2.80 | 4 70 | 1.20 | Versicolour |
| 5 90 | 2.80 | 4 30 | 1 30 | Versicolour |
| 5 90 | 2.80 | 4 40 | 1 40 | Versicolour |
| 6.00 | 2.80 | 4 80 | 1 40 | Versicolour |
| 6.00 | 2.00 | 5.00 | 1.10 | Versicolour |
| 6.00 | 2.90 | 4 50 | 1.50 | Versicolour |
| 6.00 | 2.90 | 3 50 | 1.00 | Versicolour |
| 6.10 | 2.90 | 3.80 | 1.00 | Versicolour |
| 6.10 | 2.90 | 3.70 | 1.00 | Versicolour |
| 6.10 | 2.90 | 3.90 | 1.00 | Versicolour |
| 6.10 | 2.90 | 5.10 | 1.20 | Versicolour |
| 6.20 | 3.00 | 4 50 | 1.50 | Versicolour |
| 6.20 | 3.00 | 4.50 | 1.50 | Versicolour |
| 6.30 | 3.00 | 4.30 | 1.00 | Versicolour |
| 6.30 | 3.00 | 4.70 | 1.30 | Versicolour |
| 6 30 | 3.00 | 4 10 | 1 30 | Versicolour |
| 6.40 | 3.00 | 4.00 | 1 30 | Versicolour |
| 6.40 | 3.00 | 4.00 | 1.30 | Versicolour |
| 6 50 | 3.00 | 4.40 | 1.20 | Versicolour |
| 6.50 | 3.00 | 4.00 | 1.40 | Versicolour |
| 6.60 | 3.10 | 3 20 | 1.20 | Versicolour |
| 6 70 | 3.10 | 5.50 A 20 | 1.00 | Versicolour |
| 670 | 2 20 | 4.20 | 1.30 | Versioolour |
| 6.70 | 3.20 | 4.20 | 1.20 | Vorsicolour |
| 0.70 | 3.20 | 4.20 | 1.30 | Vorsicolour |
| 0.80 | 3.20 | 4.30 | 1.30 | Versicelour |
| 0.90 | 5.50 2.40 | 5.00 | 1.10 | Versieeleur |
| /.00 | 3.40 | 4.10 | 1.30 | Versicolour |
| 4.90 | 2.20 | 0.00 | 2.50 | virginica |

Çizelge C.1 : İris veri kümesi (devam).

| Çanak Yaprağı | Çanak Yaprağı | Taç Yaprağı | Taç Yaprağı | Sinif |
|---------------|----------------|---------------|----------------|-----------|
| Uzunluğu [cm] | Genişliği [cm] | Uzunluğu [cm] | Genişliği [cm] | Simi |
| 5.60 | 2.50 | 5.10 | 1.90 | Virginica |
| 5.70 | 2.50 | 5.90 | 2.10 | Virginica |
| 5.80 | 2.50 | 5.60 | 1.80 | Virginica |
| 5.80 | 2.50 | 5.80 | 2.20 | Virginica |
| 5.80 | 2.60 | 6.60 | 2.10 | Virginica |
| 5.90 | 2.60 | 4.50 | 1.70 | Virginica |
| 6.00 | 2.70 | 6.30 | 1.80 | Virginica |
| 6.00 | 2.70 | 5.80 | 1.80 | Virginica |
| 6.10 | 2.70 | 6.10 | 2.50 | Virginica |
| 6.10 | 2.70 | 5.10 | 2.00 | Virginica |
| 6.20 | 2.80 | 5.30 | 1.90 | Virginica |
| 6.20 | 2.80 | 5.50 | 2.10 | Virginica |
| 6.30 | 2.80 | 5.00 | 2.00 | Virginica |
| 6.30 | 2.80 | 5.10 | 2.40 | Virginica |
| 6.30 | 2.80 | 5.30 | 2.30 | Virginica |
| 6 30 | 2 80 | 5 50 | 1.80 | Virginica |
| 6 30 | 2.80 | 6 70 | 2 20 | Virginica |
| 6 30 | 2.80 | 6 90 | 2.30 | Virginica |
| 6 40 | 2.90 | 5.00 | 1.50 | Virginica |
| 6.40 | 2.90 | 5 70 | 2 30 | Virginica |
| 6.40 | 3.00 | 4 90 | 2.00 | Virginica |
| 6 40 | 3.00 | 6 70 | 2.00 | Virginica |
| 6.40 | 3.00 | 4 90 | 2.00 | Virginica |
| 6 50 | 3.00 | 5 70 | 2 10 | Virginica |
| 6.50 | 3.00 | 5.70 | 2.10 | Virginica |
| 6.50 | 3.00 | 0.00 | 1.80 | Virginica |
| 6.50 | 3.00 | 4.80 | 1.80 | Virginica |
| 6.30 | 3.00 | 4.90 | 2.10 | Virginica |
| 0.70 6.70 | 3.00 | 5.00 | 2.10 | Virginica |
| 0.70 | 3.00 | 5.80 | 1.00 | Virginica |
| 6.70 | 3.00 | 6.10 | 1.90 | Virginica |
| 6.70 | 3.00 | 6.40 | 2.00 | Virginica |
| 6.70 | 3.00 | 5.60 | 2.20 | Virginica |
| 6.80 | 3.10 | 5.10 | 1.50 | Virginica |
| 6.80 | 3.10 | 5.60 | 1.40 | Virginica |
| 6.90 | 3.10 | 6.10 | 2.30 | Virginica |
| 6.90 | 3.10 | 5.60 | 2.40 | Virginica |
| 6.90 | 3.20 | 5.50 | 1.80 | Virginica |
| 7.10 | 3.20 | 4.80 | 1.80 | Virginica |
| 7.20 | 3.20 | 5.40 | 2.10 | Virginica |
| 7.20 | 3.20 | 5.60 | 2.40 | Virginica |
| 7.20 | 3.20 | 5.10 | 2.30 | Virginica |
| 7.30 | 3.30 | 5.10 | 1.90 | Virginica |
| 7.40 | 3.30 | 5.90 | 2.30 | Virginica |
| 7.60 | 3.30 | 5.70 | 2.50 | Virginica |
| 7.70 | 3.40 | 5.20 | 2.30 | Virginica |
| 7.70 | 3.40 | 5.00 | 1.90 | Virginica |
| 7.70 | 3.60 | 5.20 | 2.00 | Virginica |
| 7.70 | 3.80 | 5.40 | 2.30 | Virginica |
| 7.90 | 3.80 | 5.10 | 1.80 | Virginica |

Çizelge C.1 : İris veri kümesi (devam).

EK C.2

Bu ekte veri sınıflandırıcılarının uygulamalarında eğitim ve test amacı ile kullanılan Haberman verisi verilmiştir. x_1 hastanın yaşını, x_2 ameliyat tarihini ve x_3 pozitif netice veren göğüs kanserli hasta sayısını göstermektedir. c_1 beş yıldan daha uzun yaşayan, c_2 beş yıldan daha az yaşayan hasta sınıfını göstermektedir.

| x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 30 | 64 | 1 | c_1 | 37 | 63 | 0 | c_1 | 39 | 59 | 2 | c_1 | 42 | 62 | 20 | c_1 |
| 30 | 62 | 3 | c_1 | 37 | 58 | 0 | c_1 | 39 | 63 | 4 | c_1 | 42 | 65 | 0 | c_1 |
| 30 | 65 | 0 | c_1 | 37 | 59 | 6 | c_1 | 40 | 58 | 2 | c_1 | 42 | 63 | 1 | c_1 |
| 31 | 59 | 2 | c_1 | 37 | 60 | 15 | c_1 | 40 | 58 | 0 | c_1 | 43 | 63 | 14 | c_1 |
| 31 | 65 | 4 | c_1 | 37 | 63 | 0 | c_1 | 40 | 65 | 0 | c_1 | 43 | 64 | 2 | c_1 |
| 33 | 58 | 10 | c_1 | 38 | 59 | 2 | c_1 | 41 | 58 | 0 | c_1 | 43 | 64 | 3 | c_1 |
| 33 | 60 | 0 | c_1 | 38 | 60 | 0 | c_1 | 41 | 59 | 8 | c_1 | 43 | 60 | 0 | c_1 |
| 34 | 60 | 1 | c_1 | 38 | 62 | 3 | c_1 | 41 | 59 | 0 | c_1 | 43 | 65 | 0 | c_1 |
| 34 | 61 | 10 | c_1 | 38 | 64 | 1 | c_1 | 41 | 64 | 0 | c_1 | 43 | 66 | 4 | c_1 |
| 34 | 67 | 7 | c_1 | 38 | 66 | 0 | c_1 | 41 | 69 | 8 | c_1 | 44 | 61 | 0 | c_1 |
| 34 | 60 | 0 | c_1 | 38 | 66 | 11 | c_1 | 41 | 65 | 0 | c_1 | 44 | 63 | 1 | c_1 |
| 35 | 64 | 13 | c_1 | 38 | 60 | 1 | c_1 | 41 | 65 | 0 | c_1 | 44 | 61 | 0 | c_1 |
| 35 | 63 | 0 | c_1 | 38 | 67 | 5 | c_1 | 42 | 58 | 0 | c_1 | 44 | 67 | 16 | c_1 |
| 36 | 60 | 1 | c_1 | 39 | 63 | 0 | c_1 | 42 | 60 | 1 | c_1 | 45 | 60 | 0 | c_1 |
| 36 | 69 | 0 | c_1 | 39 | 67 | 0 | c_1 | 42 | 59 | 2 | c_1 | 45 | 67 | 0 | c_1 |
| 45 | 59 | 14 | c_1 | 49 | 62 | 0 | c_1 | 52 | 61 | 0 | c_1 | 54 | 69 | 7 | c_1 |
| 45 | 64 | 0 | c_1 | 49 | 66 | 0 | c_1 | 52 | 63 | 4 | c_1 | 54 | 63 | 19 | c_1 |
| 45 | 68 | 0 | c_1 | 49 | 60 | 1 | c_1 | 52 | 69 | 0 | c_1 | 54 | 58 | 1 | c_1 |
| 45 | 67 | 1 | c_1 | 49 | 62 | 1 | c_1 | 52 | 60 | 4 | c_1 | 54 | 62 | 0 | c_1 |
| 46 | 62 | 0 | c_1 | 49 | 63 | 3 | c_1 | 52 | 60 | 5 | c_1 | 55 | 58 | 1 | c_1 |
| 46 | 58 | 3 | c_1 | 49 | 61 | 0 | c_1 | 52 | 62 | 0 | c_1 | 55 | 58 | 0 | c_1 |
| 46 | 63 | 0 | c_1 | 49 | 67 | 1 | c_1 | 52 | 62 | 1 | c_1 | 55 | 58 | 1 | c_1 |
| 47 | 61 | 0 | c_1 | 50 | 59 | 0 | c_1 | 52 | 64 | 0 | c_1 | 55 | 66 | 18 | c_1 |
| 47 | 63 | 6 | c_1 | 50 | 61 | 6 | c_1 | 52 | 65 | 0 | c_1 | 55 | 66 | 0 | c_1 |
| 47 | 66 | 0 | c_1 | 50 | 61 | 0 | c_1 | 52 | 68 | 0 | c_1 | 55 | 69 | 3 | c_1 |
| 47 | 67 | 0 | c_1 | 50 | 63 | 1 | c_1 | 53 | 58 | 1 | c_1 | 55 | 69 | 22 | c_1 |
| 47 | 58 | 3 | c_1 | 50 | 58 | 1 | c_1 | 53 | 60 | 1 | c_1 | 55 | 67 | 1 | c_1 |
| 47 | 60 | 4 | c_1 | 50 | 59 | 2 | c_1 | 53 | 60 | 2 | c_1 | 56 | 60 | 0 | c_1 |
| 47 | 68 | 4 | c_1 | 50 | 61 | 0 | c_1 | 53 | 61 | 1 | c_1 | 56 | 66 | 2 | c_1 |
| 47 | 66 | 12 | c_1 | 50 | 64 | 0 | c_1 | 53 | 63 | 0 | c_1 | 56 | 66 | 1 | c_1 |
| 48 | 61 | 8 | c_1 | 50 | 65 | 4 | c_1 | 54 | 59 | 7 | c_1 | 56 | 67 | 0 | c_1 |

Çizelge C.2 : Haberman veri kümesi.

| x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf | x_1 | x_2 | x_3 | Sınıf |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 58 | 60 | 3 | c_1 | 59 | 67 | 3 | c_1 | 61 | 64 | 0 | c_1 | 76 | 67 | 0 | c_1 |
| 58 | 61 | 1 | c_1 | 60 | 61 | 1 | c_1 | 61 | 65 | 8 | c_1 | 77 | 65 | 3 | c_1 |
| 58 | 67 | 0 | c_1 | 59 | 64 | 7 | c_1 | 61 | 68 | 0 | c_1 | 34 | 59 | 0 | c_2 |
| 58 | 58 | 0 | c_1 | 63 | 63 | 0 | c_1 | 61 | 59 | 0 | c_1 | 34 | 66 | 9 | c_2 |
| 58 | 58 | 3 | c_1 | 63 | 66 | 0 | c_1 | 62 | 62 | 6 | c_1 | 38 | 69 | 21 | c_2 |
| 58 | 61 | 2 | c_1 | 63 | 61 | 9 | c_1 | 62 | 66 | 0 | c_1 | 39 | 66 | 0 | c_2 |
| 59 | 60 | 0 | c_1 | 63 | 61 | 28 | c_1 | 62 | 66 | 0 | c_1 | 41 | 60 | 23 | c_2 |
| 65 | 58 | 0 | c_1 | 64 | 58 | 0 | c_1 | 62 | 58 | 0 | c_1 | 41 | 64 | 0 | c_2 |
| 65 | 64 | 0 | c_1 | 64 | 65 | 22 | c_1 | 63 | 61 | 0 | c_1 | 41 | 67 | 0 | c_2 |
| 65 | 67 | 0 | c_1 | 64 | 66 | 0 | c_1 | 63 | 62 | 0 | c_1 | 42 | 69 | 1 | c_2 |
| 65 | 59 | 2 | c_1 | 64 | 61 | 0 | c_1 | 63 | 63 | 0 | c_1 | 42 | 59 | 0 | c_2 |
| 65 | 64 | 0 | c_1 | 64 | 68 | 0 | c_1 | 67 | 66 | 0 | c_1 | 43 | 58 | 52 | c_2 |
| 71 | 68 | 2 | c_1 | 69 | 65 | 0 | c_1 | 67 | 61 | 0 | c_1 | 43 | 59 | 2 | c_2 |
| 70 | 63 | 0 | c_1 | 70 | 66 | 14 | c_1 | 68 | 67 | 0 | c_1 | 43 | 64 | 0 | c_2 |
| 72 | 58 | 0 | c_1 | 70 | 67 | 0 | c_1 | 68 | 68 | 0 | c_1 | 43 | 64 | 0 | c_2 |
| 72 | 64 | 0 | c_1 | 70 | 68 | 0 | c_1 | 69 | 60 | 0 | c_1 | 44 | 64 | 6 | c_2 |
| 72 | 67 | 3 | c_1 | 65 | 67 | 1 | c_1 | 74 | 63 | 0 | c_1 | 44 | 58 | 9 | c_2 |
| 73 | 62 | 0 | c_1 | 73 | 68 | 0 | c_1 | 75 | 62 | 1 | c_1 | 44 | 63 | 19 | c_2 |
| 46 | 62 | 5 | c_2 | 48 | 67 | 7 | c_2 | 52 | 59 | 2 | c_2 | 45 | 65 | 6 | c_2 |
| 46 | 65 | 20 | c_2 | 49 | 63 | 0 | c_2 | 52 | 62 | 3 | c_2 | 45 | 66 | 0 | c_2 |
| 47 | 63 | 23 | c_2 | 49 | 64 | 10 | c_2 | 52 | 66 | 4 | c_2 | 45 | 67 | 1 | c_2 |
| 47 | 62 | 0 | c_2 | 50 | 63 | 13 | c_2 | 53 | 58 | 4 | c_2 | 46 | 58 | 2 | c_2 |
| 47 | 65 | 0 | c_2 | 50 | 64 | 0 | c_2 | 53 | 65 | 1 | c_2 | 46 | 69 | 3 | c_2 |
| 48 | 58 | 11 | c_2 | 51 | 59 | 13 | c_2 | 53 | 59 | 3 | c_2 | 56 | 65 | 9 | c_2 |
| 48 | 58 | 11 | c_2 | 51 | 59 | 3 | c_2 | 53 | 60 | 9 | c_2 | 56 | 66 | 3 | c_2 |
| 54 | 65 | 5 | c_2 | 52 | 69 | 3 | c_2 | 53 | 63 | 24 | c_2 | 57 | 61 | 5 | c_2 |
| 54 | 68 | 7 | c_2 | 56 | 65 | 9 | c_2 | 53 | 65 | 12 | c_2 | 57 | 62 | 14 | c_2 |
| 55 | 63 | 6 | c_2 | 56 | 66 | 3 | c_2 | 54 | 60 | 11 | c_2 | 57 | 64 | 1 | c_2 |
| 55 | 68 | 15 | c_2 | 57 | 61 | 5 | c_2 | 54 | 65 | 23 | c_2 | 59 | 62 | 35 | c_2 |
| 62 | 59 | 13 | c_2 | 57 | 64 | 1 | c_2 | 59 | 62 | 35 | c_2 | 60 | 65 | 0 | c_2 |
| 62 | 58 | 0 | c_2 | 61 | 62 | 5 | c_2 | 60 | 59 | 17 | c_2 | 61 | 62 | 5 | c_2 |
| 62 | 65 | 19 | c_2 | 61 | 65 | 0 | c_2 | 60 | 65 | 0 | c_2 | 61 | 65 | 0 | c_2 |
| 63 | 60 | 1 | c_2 | 65 | 66 | 15 | c_2 | 67 | 63 | 1 | c_2 | 70 | 58 | 0 | c_2 |
| 65 | 58 | 0 | c_2 | 66 | 58 | 0 | c_2 | 69 | 67 | 8 | c_2 | 70 | 58 | 4 | c_2 |
| 65 | 61 | 2 | c_2 | 66 | 61 | 13 | c_2 | 74 | 65 | 3 | c_2 | 72 | 63 | 0 | c_2 |
| 65 | 62 | 22 | c_2 | 67 | 64 | 8 | c_2 | 78 | 65 | 1 | c_2 | 83 | 58 | 2 | c_2 |
| 57 | 63 | 0 | c_1 | 59 | 63 | 0 | c_1 | 60 | 67 | 2 | c_1 | 66 | 58 | 0 | c_1 |
| 57 | 64 | 0 | c_1 | 59 | 64 | 1 | c_1 | 60 | 61 | 25 | c_1 | 66 | 58 | 1 | c_1 |
| 57 | 67 | 0 | c_1 | 59 | 64 | 0 | c_1 | 61 | 59 | 0 | c_1 | 67 | 66 | 0 | c_1 |

Çizelge C.2 : Haberman veri kümesi (devam).

ÖZGEÇMİŞ



| Ad Soyad: | Merih YILDIZ |
|---------------------------|--|
| Doğum Yeri ve Tarihi: | Zonguldak, 11-08-1978 |
| Adres: | Acarlar Sitesi C-4 Blok d:11 İstanbul. |
| Lisans Üniversite: | İstanbul Teknik Üniversitesi, Elektronik ve Haberleşme Mühendisliği Bölümü, 2000. |
| Yüksek Lisans Üniversite: | İstanbul Teknik Üniversitesi, Elektronik ve Haberleşme Mühendisliği Bölümü, 2003. |

Yayın Listesi:

• Yıldız, M., Minaei, S., and Özoğuz, S., 2009: Linearly Weighted Classifier Circuit, *Northeast Workshop on Circuits and Systems*, konferans kitabında basılmak üzere kabul edilmiştir.

• Yıldız, M., Minaei, S., and Göknar, C., 2008: Realization and Template Matching Application of a CMOS Classifier Circuit, *IEEE Applied Electronics*, Pilsen, Czech Republic, September 10-11, pp. 231-234.

• Yıldız, M., Minaei, S., and Göknar, C., 2007: A CMOS Classifier Circuit using Neural Networks with Novel Architecture, *IEEE Transaction on Neural Networks*, Vol. 18, pp. 1845-1849.

• Yıldız, M., Minaei, S., and Göknar, C., 2007: A Low-Power Multilevel-Output Classifier Circuit, *European Conference on Circuit Theory and Design*, August 26-30 Ağustos, Seville, Spain, pp. 747-750.

• Yıldız, M., Minaei, S., and Göknar, C., 2006: Current Mode Double Threshold Neuron Activation Function, *Complex Computing Networks, Springer in Physics Series*, Vol. 104, pp. 267-274.
• Minaei S., Yıldız, M., Türköz, S., and Kuntman, H., 2003: High Swing CMOS Realization for Third Generation Current Conveyor (CCIII), *Istanbul University Journal of Electrical and Electronics*, Vol **3**, pp. 819-826.

• Yıldız, M., Minaei, S., and Göknar, C., 2006: CMOS Realization of a Quantized-Output Classifier Circuit, 13th IEEE International Conference on Electronics, Circuits and Systems, December 10-13, Nice, France, pp. 292-295.

• Yıldız, M., Minaei, S., and Göknar, C., 2005: Current Mode Double Threshold Neuron Activation Function, *Complex Computing-Networks: Brain-like and Wave-oriented Electrodynamic Algorithms Springer Proceedings in Physics*, Vol. 104, pp. 267-274.

• Minaei, S., Yıldız, M., and Kuntman, H., 2004: New Realization of Floating Lossless Inductance and R-L Impedance Simulators, *Northeast Workshop on Circuits and Systems*, June 20-23, Montreal, Canada, pp. 313-316.

• Minaei, S., Yıldız, M., and Kuntman, H., 2004: New Realization of Voltage-Mode Multifunction Filters without External Passive Elements, 2th IEEE Mediterranean Electrotechnical Conference, Dubrovnik, Croatia, May 12-15, pp. 99-102.

• Minaei S., Yıldız, M., Metin, B., and Çicekoğlu, O., 2004: İki Ucu Serbest Kayıpsız Endüktans ve R-L Empedans Benzetimlerinin Aktif Devreler İle Gerçeklenmesi, *Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyomu*, December 8-12, Bursa, Turkey, pp. 96-99.

 Minaei, S., Yıldız, M., and Kuntman, H., 2004: Sadece Aktif Eleman Kullanılarak Süzgeç Tasarımı ve Gerçekleştirilmesi, *IEEE 12. Sinyal İşleme ve İletişim Uygulamaları Kurultayı*, Kuşadası, Turkey, April 28-30, pp. 630-633.

• Minaei, S., Yıldız, M., Kuntman, H., and Turkoz, S., 2002: Yeni Yüksek Başarımlı CMOS Üçüncü Kuşak Akım Taşıyıcı (CCIII), *Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyomu*, Bursa, Turkey, December 18-22, pp. 108-112.

• Minaei, S., Yıldız, M., Kuntman, H., and Turkoz, S., 2002: High Performance CMOS Realization of The Third Generation Current conveyor (CCIIIs), 45th IEEE International Midwest Symposium on Circuit and Systems, August 4-7, Oklahoma, USA, pp. 307-310.

Realization and Template Matching Application of a CMOS Classifier Circuit

Merih Yıldız, Shahram Minaei, İzzet Cem Göknar

Dept. of Electronics and Communications Engineering, Dogus University, Acibadem 34722, Istanbul, Turkey. E-mail: myildiz@dogus.edu.tr, sminaei@dogus.edu.tr, cgoknar@dogus.edu.tr

Abstract— In this paper a new CMOS classifier circuit is presented, and its application to character recognition and template matching is developed. The proposed CMOS circuit operates in current-mode and can classify several types of data. The architecture is designed using two threshold circuits and a subtraction circuit. Using 0.35 µm TSMC technology parameters, SPICE simulations for the classifier and application circuits are included.

INTRODUCTION

Classification is an important topic in many applications such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks, statistics and template matching [1-5]. Character recognition and pattern classification using template matching techniques are powerful tools in many systems. However, the processing is computationally very expensive, consuming a lot of CPU time when implemented as software for general purpose computers. Therefore, software approaches are not practical for real-time applications [6]. Whereas the literature abounds with soft algorithms, hard classifiers are seldom encountered.

Template matching is a frequently used technique in digital image processing for finding small parts of an image which match a template image [7]. It can be used in manufacturing as part of quality control, in robot vision, as a way to navigate a mobile robot, as a way to detect edges in images etc. There are quite a few different approaches for accomplishing template matching. Some perform better than others, and some find better matches. The basic method in template matching is to loop through all the pixels in the search image and compare them to the pattern. While this method is simple to implement and understand, it is one of the slowest methods. Another way to make the matching faster is to divide the image into smaller images, and then search the smaller subimages. After finding matches in the smaller images, the obtained information will be used in the larger image. However, conventional template matching methods using a template image consume a large amount of computational time. A number of techniques have been investigated with the intent of speeding up the process [8–9]. Whereas template matching tools

recognize a single input character, Character Recognition tools recognize many from a given alphabet; that is the only distinction. The purpose of this paper is twofold: i) to design a simpler and better performing circuit, ii) to use the classifier circuits in character recognition and template matching applications.

CMOS CORE CIRCUIT (CC)

The block diagram of a CC and its transfer characteristic are shown in Fig. 1a and Fig. 1.b, respectively.



Fig. 1.a. CC block diagram. Fig. 1.b. Transfer characteristic of the CC.

The input-output transfer characteristic shown in Fig. 1.b can be expressed as:

$$I_{out} = \begin{cases} I_{H} & \text{if} & I_{1} < I_{in} < I_{2} \\ 0 & \text{otherwise} \end{cases}$$
(1)

As shown in Fig. 2, the CC can be constructed using two threshold circuits and a current mirror acting as a subtractor.

$$I_{1} \downarrow I_{2} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{H} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

$$I_{1} \downarrow$$

Fig. 2. The block diagram of the CC

The current I_H determines the level of the output current, the currents I_1 and I_2 are used to shift the horizontal position of the output current. There are many techniques in the literature for the CMOS realization of current comparator and threshold circuits [10-11].

The current sources I_{in} , I_1 and I_H in Fig. 3 are applied to the circuit through simple current mirrors. All of the transistors M_1 - M_6 are identical. The current I_1 is the threshold value of the circuit. If the input current I_{in} is smaller than the threshold current I_1 , then the drain voltage of the transistor M_2 becomes approximately V_{DD} , and the current I_H flows through M_4 , so the output current is zero.

This work is part of project 106E139 supported by the Scientific & Technological Research Council of Turkey (TÜBİTAK).



Fig. 3. CMOS implementation of the CC

When the current I_{in} exceeds the threshold current I_1 , the drain voltage of the transistor M_3 becomes approximately V_{SS} , the current flowing through M_4 is zero and the current I_H flows through M_5 , so the output current becomes equal to I_H .

To realize a CC, two threshold circuits and a subtractor circuit are used as shown in Fig. 3. Thus the classification region is obtained by differencing the output currents of the threshold circuits yielding the transfer characteristic as shown in Fig. 1b. Note that the current I_H of the threshold circuits should be equal and the following constraint must be satisfied

$$I_2 > I_1 \tag{2}$$

In Fig. 3 the transistors M_1 - M_6 and M_9 - M_{14} constitute two threshold circuits respectively.

The basic current mirror constructed with the transistors M_7 and M_8 performs the desired operation of subtraction. The transistors M_{15} , M_{16} , M_{17} are used to provide currents equal to I_H (adjusting the output level) for the threshold circuits. Similarly, the same approach is used with M_{18} , M_{19} and M_{20} to apply the input current I_{in} to both of the threshold circuits.

By interconnecting several core circuits and adding the output currents a multi-input classifier (*n*dimensional) can be obtained as shown in Fig. 4 [12]. The current I_{in-k} (*k*=1, 2,...,n) is the input current of the *k*-th CC and I_{out} is the output of the multi-output classifier.

CHARACTER RECOGNITION AND TEMPLATE MATCHING APPLICATIONS

Character recognition is a special case of pattern classification and template matching, which is a classical technique of classifying subimages inside a larger image, a special case of character recognition; in fact a template is the subimage of a larger image. The subimage can be a character, a special picture or a number. Here, the application is constructed for multiple subimages assuming that the templates are chosen as decimal numbers as shown in Fig. 5. Each subimage in the template is divided into $4 \times 5=20$ cells as illustrated in Fig. 6.



Fig. 4. Block diagram of an n-dimensional classifier.



Fig. 5. Decimal number template examples.

| \mathbf{a}_1 | b ₁ | c ₁ | \mathbf{d}_1 |
|----------------|-----------------------|-----------------------|----------------|
| \mathbf{a}_2 | b ₂ | c ₂ | \mathbf{d}_2 |
| a3 | b ₃ | c ₃ | d ₃ |
| a_4 | \mathbf{b}_4 | c ₄ | \mathbf{d}_4 |
| \mathbf{a}_5 | \mathbf{b}_5 | c ₅ | \mathbf{d}_5 |

Fig. 6. Cell arrangement of the template.

The template matching classifier topology is designed to compare the sample template cells with other template cells as shown in Fig. 7. The block diagram shown in Fig. 7 is composed of fourdimensional multilevel classifiers (FDMC). Each FDMC with same output function $f(x_1,x_2,x_3,x_4)$ is composed of four core cells connected in parallel in the form shown in Fig. 4 with output currents connected to the same node. The output of each FDMC is applied to a single CC with function "g(y)". The values of a_i, b_i, c_i, d_i (i=1,...,5) can be digitally "1" or "0" depending on the applied input template. The output of FDMC can be expressed as:

$$f(x_1, x_2, x_3, x_4) = x_1 2^0 + x_2 2^1 + x_3 2^2 + x_4 2^3 \quad (3)$$

Each row of the sample template is applied to the input of the corresponding FDMC following the order given in Fig.7. Depending on the applied template, the output of each FDMC block, which changes according to (4), is given in Table 1 and it can be observed that none of the five dimensional outputs coincides with the other. So by checking outputs y_i (i=1,...,5) according to Table 1, the input pattern can be determined and hence **character recognition** achieved at this level; if a single output is desired for recognition, these outputs can be added (a single node

will suffice in case they are currents) as the sum Σ is different for each input template.

To further realize **template matching**, single CCs with transfer characteristic $g_i(y_i)$ (i=1,...,5) are introduced as shown in Fig. 7; the control currents of the CCs in Fig. 1.b are chosen appropriately to indicate the selected template. If the input of the core cell block $g_i(y_i)$ is in the correct region the outputs z_i will be digitally "1" otherwise "0". So if a template to be matched is applied to the inputs and the applied template is correct then the output *z* of the classifier will be equal to "5" in magnitude; if it is not correct it will be less than "5".

It should be observed from Table 1 that for any two different inputs the outputs y_i are the same at most for three values of i=1,2,...,5 and not four, thus providing the basis for **fault diagnosis**. In fact if there is an error in one line at one or more pixels, only four of the outputs z_i will be "1" and z will be "4" in magnitude. For **error correction** purposes, a final CC with transfer characteristic g(z) having properly chosen threshold values (for example 3.5 μ A for I₁ and 5.5 μ A for I₂) can be added. This CC will then produce an output equal to "1" in magnitude, correctly identifying the error containing template to be matched.



Fig. 7. Template matching topology constructed with classifiers and core cells.

| Templates | y 1 | y ₂ | y ₃ | y 4 | y 5 | Σ |
|-----------|------------|-----------------------|-----------------------|------------|------------|----|
| 0 | 7 | 5 | 5 | 5 | 7 | 29 |
| 1 | 8 | 8 | 8 | 8 | 8 | 40 |
| 2 | 14 | 8 | 14 | 2 | 14 | 52 |
| 3 | 14 | 8 | 12 | 8 | 14 | 56 |
| 4 | 10 | 10 | 14 | 8 | 8 | 50 |
| 5 | 7 | 1 | 7 | 4 | 7 | 26 |
| 6 | 1 | 1 | 7 | 5 | 7 | 21 |
| 7 | 7 | 4 | 4 | 4 | 4 | 23 |
| 8 | 14 | 10 | 14 | 10 | 14 | 62 |
| 9 | 7 | 5 | 7 | 4 | 4 | 27 |

Table 1. The outputs yi (i=1,...,5) of different FDMCs and their sum. (Currents in μA)

SIMULATION RESULTS

The CC in Fig. 3 has been simulated with 0.35 μ m TSMC CMOS parameters using SPICE. Supply voltages are selected as ± 1 V. The transistor dimensions are given in Table 2.

| MOSFET | $\begin{array}{c} M_4,M_{5,} \\ M_6,M_{12}, \\ M_{13},M_{14} \end{array}$ | $\begin{array}{c} M_{7},M_{8},M_{15},M_{16},\\ M_{17},M_{18},M_{19},M_{20},\\ M_{25},M_{26},M_{27},M_{28} \end{array}$ | M ₁ , M ₂ , M ₉ , M ₁₀ |
|--------|---|--|---|
| W [µm] | 10.5 | 21 | 42 |
| L [µm] | 1.05 | 1.05 | 1.05 |

Table 2. Dimensions of the MOS transistors in the circuit of Fig. 3.

The classifier circuit has been tested for the template matching application of Fig. 5. In the simulations, magnitude corresponding to "1" is chosen as 10 µA. To simulate the circuit, from 0 ms to 1ms template "0", 1 ms to 2 ms template "1", and other templates are applied one by one until 10 ms. The simulation results of the template matching application are shown in Fig. 8 where F(i) means the control currents are set to classify the template i. From Fig. 8 it can be seen that from 0 ms to 10 ms only one section (1 ms period) according to the applied template has the value 10 µA thus confirming correct operation of the classifier circuit. To test error correction, one row has been changed at one or more pixels and an output current of 8 µA has been observed for several examples.

CONCLUSION

In this paper, a new kind of current-mode core cell has been proposed and its use illustrated in character recognition and template matching applications. The circuit is based on sub-circuits performing the threshold and subtraction operations. Although the template used in applications requires only binary values, the classifier circuit can be used as well with other values. The proposed custom tunable circuit can also be used in other applications such as classification, matching, quantization etc. SPICE simulations have also verified the correct operation of the classifier circuit.

REFERENCES

- B. Liu, C. Chen, and J. Tsao, "A Modular Current-Mode Classifier Circuit for Template Matching Application", IEEE Trans. on CAS. II, Analog and Digital Sig. Process., vol. 47, no. 2, pp. 145-151, 2000.
- [2] E. Hunt, Artificial Intelligence. New York: Academic, 1975.
- [3] H.S. Abdel-Aty-Zohdy and M. Al-Nsour, "Reinforcement learning neural network circuits for electronic nose" IEEE International Symposium on Circuits and Systems, Orlando, FL, vol. 5, 30 May-2 June, 1999, pp. 379 – 382.
- [4] G. Lin and B. Shi, "A current-mode sorting circuit for pattern recognition" Intelligent Processing and Manufacturing of Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 1003 – 1007.
- [5] G. Lin and B. Shi, "A multi-input current-mode fuzzy integrated circuit for pattern Recognition" Second International Conference on Intelligent Processing and Manufacturing Materials, Honolulu, Hawaii, July 10-15, 1999, pp. 687-693.
- [6] T. Yamasaki and T. Shibata, "Analog soft-pattern-matching classifier using floating-gate MOS technology" IEEE Trans. on NN, vol. 14, no. 5, pp.1257-1265, 2003.
- [7] C.-H. Chou and Y.-C. Chen, "A VLSI Architecture for realtime and flexible image template matching", IEEE Trans. on CAS, vol. 36, no. 10, pp. 1336-1342, 1989.
- [8] J. P. Lewis, "Fast template matching," Proc. Vision Interface, 1995, pp. 120–123.
- [9] G. J. VanderBrug and A. Rosenfeld, "Two-stage template matching," IEEE Trans. Comput., vol. C-26, no. 4, pp. 384– 393, 1977.
- [10] D.A. Freitasa and K.W. Current, "CMOS Current Comparator Circuit", Electronics Letters, vol.19, no.17, pp. 695-697, August 1983.
- [11] A. Morgül and T. Temel, "Current-mode level restoration: circuit for multi-valued logic", Electron. Lett., vol.41, no. 5, pp. 230-231, March 2005.
- [12] M. Yıldız, S. Minaei, C. Göknar, "A CMOS Classifier Circuit using Neural Networks with Novel Architecture", IEEE Transaction on NN, vol. 18, vo.6, pp.1845-1849, 2007.



Fig. 8. Simulation results of classifier circuit.

TÜBİTAK PROJE ÖZET BİLGİ FORMU

Proje No:106E139

Proje Başlığı: ASD: Çok Amaçlı Ayarlanabilir Sınıflandırıcı Devreler

Proje Yürütücüsü ve Araştırmacılar: Prof. Dr. Cem GÖKNAR, Prof. Dr. Shahram MINAEI

Dr. Merih YILDIZ(Bursiyer), Dr. Engin DENİZ (Bursiyer)

Projenin Yürütüldüğü Kuruluş ve Adresi:

Doğuş Üniversitesi, Elektronik ve Haberleşme Bölümü, Acıbadem, Kadıköy 34722 İstanbul.

Destekleyen Kuruluş(ların) Adı ve Adresi:

TÜBİTAK EEEAG

Atatürk Bulvarı No:221 06100 Kavaklıdere-Ankara.

Projenin Başlangıç ve Bitiş Tarihleri:1 Şubat 2007-1 Ağustos 2010

Öz (en çok 70 kelime) Çalışmada, ayarlanabilir sınıflandırıcı devreleri ve uygulama alanları incelenmiştir. AMS 0.35 µm CMOS prosesi ile, tasarlanan sınıflandırıcı bir tümdevrenin üretimi de yapılmıştır. Bu sınıflandırıcı devresinin kontrol parametrelerinin bulunmasını sağlayan öğrenme algoritmaları çeşitli uygulamalar için geliştirilmiştir. Sınıflandırma işlemleri geliştirilen algoritmalar ve üretilen devre ile İris ve Haberman veri kümelerine uygulanarak sonuçların uyum içinde olduğu gösterilmiştir.

Anahtar Kelimeler: Sınıflandırıcı devreler, CMOS, Akım Taşıyıcı, Karakter Tanıma

<u>Fikri Ürün Bildirim Formu</u> Sunuldu mu? Evet 🗌 Gerekli Değil 🗌

Fikri Ürün Bildirim Formu'nun tesliminden sonra 3 ay içerisinde patent başvurusu yapılmalıdır.

Projeden Yapılan Yayınlar:

YILDIZ, M., Minaei, S., and Göknar, C., CMOS Realization of a Quantized- Output Classifier Circuit, 13th IEEE International Conference on Electronics, Circuits and Systems, Nice, France, December 10-13, (2006) pp: 292-295.

YILDIZ, M., Minaei, S., and Göknar, C., A CMOS Classifier Circuit using Neural Networks with Novel Architecture, *IEEE Transaction on Neural Networks*, Vol. 18, 1845-1849, (2007).

YILDIZ, M., Minaei, S., and Göknar, C., A Low-Power Multilevel-Output Classifier Circuit, European Conference on Circuit Theory and Design, Seville, Spain, August 26-30, (2007) pp: 747-750.

YILDIZ, M., Minaei, S., and Göknar, C., Realization and Template Matching Application of a CMOS Classifier Circuit, *IEEE Applied Electronics*, Pilsen, Czech Republic, September 10-11, (2008) pp: 231-234.

YILDIZ, M., Minaei, S., and Özoğuz, S., Linearly Weighted Classifier Circuit, Northeast Workshop on Circuits and Systems, 28 Haziran -1 Temmuz, (2009) pp: 99-102.

YILDIZ, M., Özoğuz, S., Minaei, S., Göknar, C., A Low-power Multilevel CMOS classifier circuit, *İTÜ Dergisi/d*, cilt. 9, sayı. 1, 57-64, (2010a).

YILDIZ, M, Minaei, S, Göknar, C, A flexible current-mode classifier circuit and its applications, *International Journal of Circuit Theory and Applications*, DOI: 10.1002/cta.677, (2010b)