

# PROGRAMLANABİLİR CMOS ANALOG VEKTÖR KUANTALAYICI TÜMDEVRESİ

## ÖZET

Kuantalama fikrinin temelleri 1898’li yıllara kadar uzanmaktadır. Kuantalayıcıların tarihsel hikayesi ve ilgili kaynakların önemli bir bölümü Robert M. GRAY ve David L. NEUHOFF’un ‘Quantization’ başlıklı makalesinde bulunabilir.

Haberleşme teorisinde, kuantalayıcılar genellikle bir işaret sıkıştırma metodu olarak algılanmışlardır. Kuantalamanın amacı şu şekilde özetlenebilir: Belli bir olasılık yoğunluk fonksiyonu ile tanımlanmış bir kaynaktan gelen işaretlerin mümkün olan en düşük bit sayısı ile kodlamak, öyle ki bu kodun çözülmesi ile elde edilen işaret mümkün olduğu kadar yüksek kalitede olsun.

Kuantalayıcıları giriş işaret vektörlerinin boyutlarına göre 2 ana kategoriye ayırmak mümkündür. Giriş işaret vektörü tek boyutlu ise kuantalayıcı ‘Skalar Kuantalayıcı’ olarak adlandırılır, aksi takdirde kuantalayıcı ‘Vektörel Kuantalayıcı’ olarak adlandırılır.

Son yıllarda Vektörel Kuantalayıcılar üzerine odaklanmış çalışmaların sayısı hızla artmaktadır. Bunun ana sebebi, Shannon’ın ‘Rate-Distortion’ teoremine göre, teorik olarak, daha iyi bir performans, her zaman, tek bir işaret yerine bir işaret bloğunun birlikte kodlanması sayesinde elde edilebilir.

Vektör Kuantalamayı,  $m$  boyutlu vektör uzayı  $R^m$ ’den onun bir alt kümesi olan  $T$  ye bir dönüşüm olarak tanımlayabiliriz.

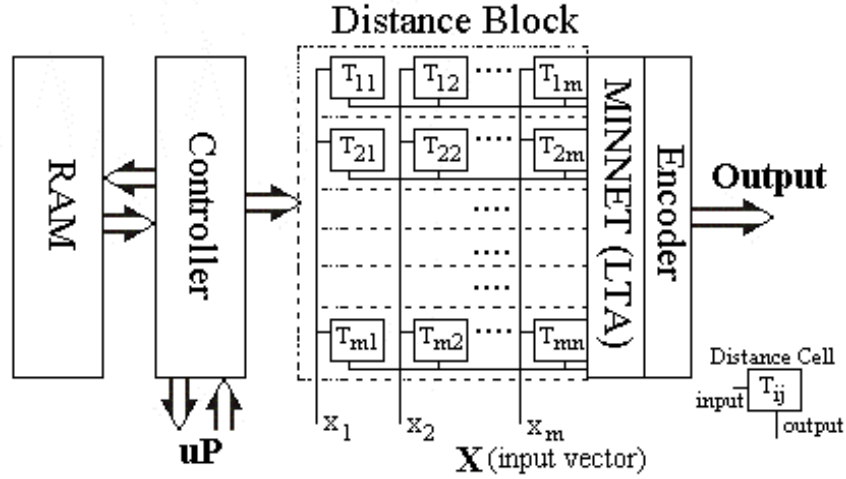
$$Q: R^m \rightarrow T$$

Kod-kitabı (code-book) olarak adlandırılan  $T$  kümesinin içinde kuantalayıcı vektörleri vardır. Kuantalayıcı, girişindeki her bir giriş vektörüne karşı düşen kuantalayıcı vektörünü seçer. Bunu şu şekilde ifade edebiliriz:

$$Q(X) = t_i$$

Daha sonra seçilen kuantalayıcı vektörünün Kod-kitabı içindeki indeksi kodlanarak çıkış işareti üretilir. Birinci işlem bir sınıflama işlemidir, ikincisi ise kodlama işlemidir.

Yukarıdaki bilgilerden yararlanarak şunu belirtebiliriz ki, Vektör Kuantalayıcı tasarımı kendi içinde iki problemi barındırır. Birinci problem olasılık yoğunluk fonksiyonu belli giriş işaretleri için ‘optimal’ Kod-kitabının tasarlanması. Bu problemin çözümü Haberleşme Teorisinin konusudur. İkinci problem ise Kod-Kitaplığı belirlenmiş bir Vektör Kuantalayıcının optimal bir şekilde silikon üzerine aktarılmasıdır. Bu tezde ikinci problemin çözümüne ilişkin bir sistem yapısı gösterilmiş; ve sistem alt blokları ile birlikte gerçekleştirilerek silikon üzerine aktarılmıştır. Aşağıda, önerilen Programlanabilir CMOS Analog Vektör Kuantalayıcı sistemin mimari yapısı görülmektedir.



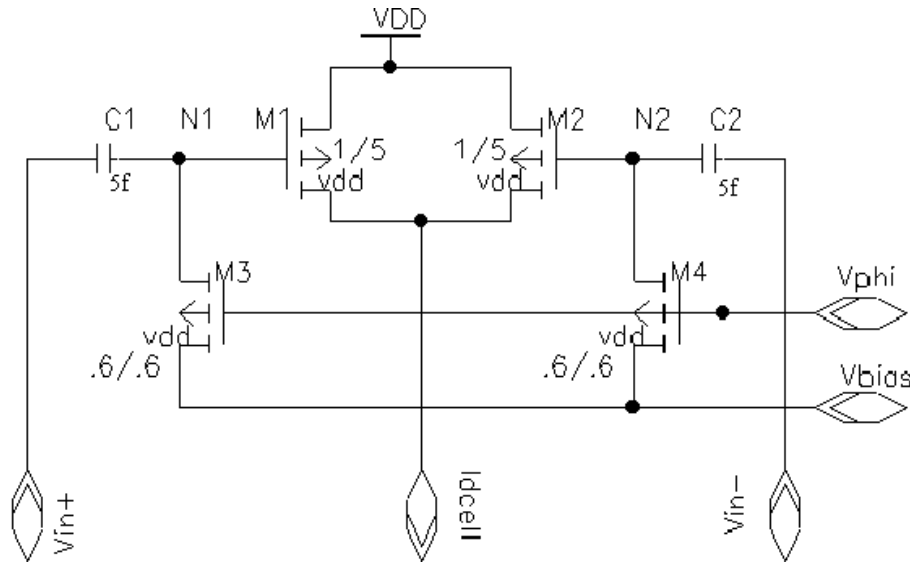
Şekil 1 :Sistem blok Şeması

Sistem kabaca iki ana fonksiyonel bölüme ayrılabilir, Distance ve MİNNET bloklarından oluşan birinci bölüm sınıflandırma işlemini, Encoder bloğundan oluşan ikinci bölüm ise kodlama işlemini yapar.

Şemadaki Distance bloğu, Distance hücre matrisinden oluşmuştur. Matristeki her bir satırda giriş vektör boyu kadar Distance hücresi vardır ve içlerinde o satıra ilişkin kuanta vektörü saklanmaktadır. Distance hücreleri girişlerine uygulanan işaret ile içlerinde saklanan kuanta vektör elemanı arasındaki mesafeyi ölçer ve çıkışlarına aktarırlar. Matristeki satır sayısı Kod-kitabındaki kuanta vektör sayısına eşittir. Distance hücresi olarak gerçekleştirilen devrenin devre şeması aşağıda görülmektedir. Bu devrenin giriş işareti gerilim, çıkış işareti akımdır. Devre kuanta vektör elemanını dinamik olarak saklamaktadır. Devre, hız, rezolüsyon ve silikon alanı açısından benzer devrelere göre çok daha avantajlı bir konumdadır. Hücre satırının çıkışında kuanta vektörü ile giriş vektörü arasındaki Euclid mesafesi ölçülür. Birim hücrenin kapladığı alan  $15.8 \mu\text{m} \times 13.6 \mu\text{m}$  dır.

Şekil 1 deki MİNNET bloğu, giriş işaretleri arasında en küçük genliğe sahip olanı seçen bir devredir. Kaybeden-Hepsini-Alır (Loser-Takes-All (LTA) ) devreleri ile gerçekleştirilir. Literatürde pek çok değişik LTA yapısı önerilmiştir. Değişik devrelerin değişik avantaj ve dezavantajları vardır. Tez de bu devre ROW\_WTA bloğu ile gerçekleştirilmiştir. Şekil 3de ROW\_WTA bloğunun şeması

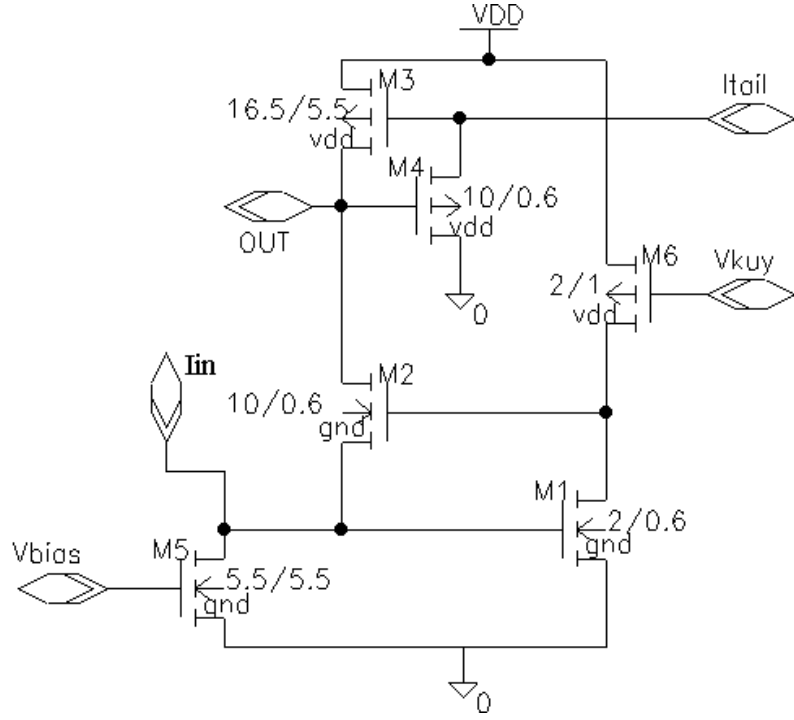
görülmektedir. Bu devre önce sabit bir akımdan giriş akımını (Distance bloğu satır çıkışı) çıkarır. Sabit akım her zaman giriş akımından büyük olacak şekilde seçilir. Böylece fark akımı vektörler arası mesafe kısaldıkça sabit akıma, arttıkça sıfıra doğru yaklaşır. Bu fark akımı J. Lazarro tarafından önerilmiş Kazanan-Hepsini-Alır (Winner-Takes-All (WTA)) devresine uygulanır ve böylece en büyük fark akımı yada girişe en küçük mesafedeki kuantta vektörü seçilmiş olur. Bu blok da küçük bir silikon alanı kaplamaktadır. Devrenin WTA kısmı sisteme hız ve rezolüsyon limitleri getirebilir. Bu sebeple yüksek performans gerektiren uygulamalarda farklı WTA topolojilerinin aranması anlamlıdır.



Şekil 2 : Distance hücresinin şema gösterimi

Encoder bloğu ROW\_WTA bloğu tarafından seçilen kuantta vektörünün Kod-kitabı içindeki indeksinin kodlanmasından sorumludur. Uygulamada bu blok kuantta vektörünün Kod-kitabındaki adresini gösteren basit bir Kodlayıcı devresi ile gerçekleştirilmiştir. Bu durumda kodlamanın optimal olması için her bir kuantta vektörünün seçilme olasılığının birbirine eşit olması gereklidir. Farklı uygulamalar için farklı encoder blokları tasarlanabilir. Devre basit bir sayısal devredir.

Sistem şemasında gözükten RAM ve Controller bloklarına gelince; Ram bloğu içinde Kod-kitabı saklanmaktadır. Distance bloğu içindeki değerlerin (Kod kitabının) yenilenmesi gerektiğinde bu bloktan yararlanır. Sistem Analog bir Vektör Kuantalayıcı olduğu için ihtiyaç duyulan RAM hücre sayısı Distance bloğundaki Distance hücre sayısı çarpı bir Distance hücresinin rezolüsyonuna eşittir. Sistemin seriminden de rahatlıkla görülebileceği üzere sistemin RAM ihtiyacı aktif alanın en büyük kısmını kaplamaktadır. Controller bloğu ise Distance bloğuna ve RAM bloğuna yazma (okuma) işlemlerinden sorumlu sayısal bloktur. Vektör kuantalayıcıyı programlamak için kullanacağımız mikroişlemci bu blok ile iletişim kurar. Basit bir sayısal kontrol devresidir ve ciddi bir silikon alanı kaplamaz.



Şekil3 : ROW\_WTA bloğunun şema gösterimi

Vektör kuantalayıcı sisteminin içinde sistemin analog olmasından kaynaklanan ve şemada gösterilmeyen iki blok daha vardır Bunlar, Distance bloğu içindeki hücrelerin giriş işaretlerini üreten giriş kuvvetlendiricisi (IN\_AMP) bloğu ve Sayısal Analog dönüştürücü (DAC) bloğudur. Bu bloklar RAM bloğu gibi çok alan kullanan bloklardır. Ve sistemin hız limitini belirlerler. Sistemin rezolüsyon sınırı DCELL, ROW\_WTA ve IN\_AMP blokları tarafından belirlenir.

Sistem içindeki her bir bloğun tasarım prosedürleri detaylı olarak tez içinde anlatılmıştır. Şimdi sistemin yerleşimini ve çalışma prensibini kısaca anlatalım.

Sistemde Kod-Kitabının yazıldığı bir RAM bloğu bulunmaktadır. RAM bloğunun kelime sayısı Kod-Kitabındaki kuanta vektörlerinin sayısı kadar, kelimelerin uzunluğu ise vektör uzunluğu çarpı bir vektör elemanının rezolüsyonu kadardır. Sistemdeki DAC bloklarının rezolüsyonu da gene bir vektör elemanının rezolüsyonu (m) kadardır. Analog işaretler üzerinden çalıştığımız için RAM bloğunun data bus'ı (m) bitlik gruplar halinde kuanta vektör elemanlarının her birini oluşturmak üzere DAC bloklarının giriş kelime hattına bağlanır.

Sistemde bulunan Distance hücre matrisindeki satır sayısı da Kod-Kitabındaki kuanta vektörlerinin sayısı kadardır. Matrisin her satırında Kod-Kitabındaki farklı bir kuanta vektörü her bir vektör elemanı bir distance hücresine gelecek şekilde saklanır. Böylece matristeki kolon sayısı vektör eleman sayısına eşit olur. Giriş vektör elemanları Distance hücre matrisine kolonlardan uygulanır. Böylece aynı kolondaki iki hücre aynı giriş vektör elemanı ile uyarılmış olur. Doğal olarak farklı iki satırda bulunan aynı kolona ait 2 ayrı Distance hücresi, içinde farklı kuanta vektör elemanı saklanmışsa, hücre çıkışlarının ait oldukları satır çıkışlarına etkileri de farklı olacaktır. Daha önce belirtildiği üzere, kuanta vektör elemanları Distance

hücrelerinde dinamik olarak saklanmaktadırlar. Bu sebeple periyodik olarak yenilenmeleri (refresh) gerekmektedir. Kuanta vektör elemanları yenilenirken de işaretler gene matris kolonlarından uygulanırlar. Bu sebeple yenileme işlemi boyunca matris satırları ardışık olarak aktif hale getirilmelidirler.

Matris kolon işaretleri IN\_AMP bloğu tarafından oluşturulur. Bunun sebebi, DCELL bloğu normal çalışabilmek için giriş işaretinin hem kendisine , hem de evriğine aynı anda ihtiyaç duymaktadır. Doğal olarak Distance hücre matrisindeki her bir kolon için bir IN\_AMP bloğunun kullanılması gerekmektedir. Bloğun iki ayrı girişi vardır birinci giriş yenileme işlemi sırasında IN\_AMP bloğunun sürdüğü kolona bağlanması gereken DAC bloğunun çıkışına bağlıdır; ikinci giriş ise sürülen bloğa ilişkin giriş vektör elemanına bağlıdır. Blok yenileme süresi boyunca DAC çıkışlarını, operasyon süresi boyunca da giriş vektör elemanını kullanarak Kolon işaretlerini üretir.

Yenileme süreci şu şekilde özetlenebilir: Microprocessör tarafından belirlenen bir zamanın sonunda Controller bloğu RAM bloğuna gerekli kontrol işaretlerini göndererek sırayla matristeki her bir Analog Kuanta vektörünün DAC çıkışlarında oluşmasını sağlar. Bu süreç boyunca IN\_AMP bloklarına gönderilen kontrol işareti ile bu blokların giriş işareti olarak DAC çıkışlarını alması sağlanır. Matrisin her bir satırı içindeki Distance hücreleri, kendilerine ilişkin Kuanta vektör elemanları girişlerinde belirttiği anda aktive edilerek, ilgili vektör elemanlarının hücre içine yazılması sağlanır. Böylece bütün satırlar yazıldıktan sonra normal işleme geçilir. Normal işlem süresince IN\_AMP blokları giriş vektör elemanlarını işlerler.

Normal operasyon boyunca, giriş işaretleri değiştikçe, matris kolon işaretleri değişir. Dolayısıyla ilgili hücrelerin çıkış işaretleri (giriş ve kuanta vektörleri arasındaki mesafeler) değişir. Ve bu işaretlere göre de sınıflama ve kodlama işlemi sürekli olarak yapılır.

Sistemin yapısı, onun Kohonen Self-Organizing Feature Map haline getirilmesine uygundur. Yapılması gereken sadece çıkış işaretlerinin bir microprocessör tarafından değerlendirilip yeni kuanta vektörlerinin Controller bloğu vasıtası ile Kod-Kitabına (RAM) yazılmasıdır.

Vektör Kuantalayıcı sisteminin serimi tez içinde bulunabilir. Sistemde 79 adet PAD bulunmaktadır. Bunların dağılımı şu şekildedir :

- 6 tane sayısal çıkış padi Encoder block çıkışı için,
- 20 tane analog pad giriş vektörünün uygulanması için,
- 8 tane analog pad Analog kutuplama işaretleri için
- 8 tane bi-directional sayısal pad microprocessör data busı için,
- 10 tane sayısal giriş padi microprocessör adres busı için,
- 3 tane sayısal giriş padi microprocessör kontrol işaretleri için,
- 2 tane sayısal giriş padi sistem saat ve reset işaretleri için,

4 çifti analog 7 çifti sayısal olmak üzere besleme padleri.

Bonding padler ile birlikte tüm sistem alanı 4.1mm x 3.7mm (15.17 mm<sup>2</sup>) kadardır. Kod-kitabının büyüklüğü göz önüne alındığında bu gerçekten küçük bir alandır. Üstelik kuantalama işlemi yapan Distance blok, MINNET (ROW\_WTA bloklarının oluşturdukları kolon) ve Encoder bloklarının kapladığı alan sistem alanının %3'den azdır. Toplam aktif alanın %70 kadarı RAM bloğu tarafından kapanır. Geri kalan %27 ise DAC, IN\_AMP ve sayısal bloklar tarafından paylaşılır.