**KOCAELİ ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**



**PROJE BAŞLIĞI: FIR FİLTRELER İLE GERÇEK ZAMANLI SES GÖSTERİMİ**

**PROJE YÜRÜTÜCÜSÜ: Seyfullah Uysal**

**DANIŞMAN: Doç. Dr. Kerem Küçük**

**5,2 cm**

**KOCAELİ 2016**

İÇİNDEKİLER

[İÇİNDEKİLER i](#_Toc469997944)

[ŞEKİLLER DİZİNİ ii](#_Toc469997945)

[ÖZET iii](#_Toc469997946)

[ABSTRACT iv](#_Toc469997947)

[GİRİŞ 1](#_Toc469997948)

[1. FİLTRELER 2](#_Toc469997949)

[1.1. FIR FİLTRELER 2](#_Toc469997950)

[1.1.1. FIR Filtre Kullanmanın Avantajları 2](#_Toc469997951)

[1.1.2. FIR Filtre Kullanmanın Dezavantajları 3](#_Toc469997952)

[1.1.3. FIR FİLTRE ÇEŞİTLERİ 4](#_Toc469997953)

[1.1.3.1. Alçak Geçişli Filtreler 4](#_Toc469997954)

[1.1.3.2. Yüksek Geçişli Filtreler 4](#_Toc469997955)

[1.1.3.3. Band Geçişli Filtreler 5](#_Toc469997956)

[1.1.3.4. Band Durduran Filtreler 5](#_Toc469997957)

[1.2. FIR QUANTİZATİON 6](#_Toc469997958)

[1.3. ADAPTİVE FİLTRE 7](#_Toc469997959)

[2. PROJENİN GERÇEKLEŞTİRİLMESİ 8](#_Toc469997960)

[2.1. TANIM 8](#_Toc469997961)

[2.2. UYGULANMASI VE ÇALIŞMASI 9](#_Toc469997962)

[2.2.1. Proje Detayı 9](#_Toc469997963)

[2.2.1.1. Arayüz 9](#_Toc469997964)

[2.2.1.2. Ses Sinyalinin Alımı ve İşlenmesi 9](#_Toc469997965)

[2.2.1.3. FIR filtreleme 10](#_Toc469997966)

[2.2.1.4. Alçak Geçiren Filtre Katsayılarının Bulunması 10](#_Toc469997967)

[2.2.1.5. Yuksek Geçiren Filtre Katsayılarının Bulunması 11](#_Toc469997968)

[2.2.1.6. Band Geçiren Filtre Katsayılarının Bulunması 11](#_Toc469997969)

[2.2.1.7. Band Durduran Filtre Katsayılarının Bulunması 12](#_Toc469997970)

[2.2.1.8. FIR Niceleme 12](#_Toc469997971)

[2.2.1.9. Adaptive FIR Filtreleme 13](#_Toc469997972)

[SONUÇLAR 16](#_Toc469997973)

[KAYNAKLAR 17](#_Toc469997974)

# ŞEKİLLER DİZİNİ

[Şekil 1.3.Adaptive filtre sistem modeli 7](#_Toc469998273)

[Şekil 1.2.Bipolar Niceleyici 1](#_Toc400643203)

[Şekil 3.2.Niceleme Formülü 1](#_Toc400643204)

[Şekil 1.1.3.3.Band Geçişli Filtre 1](#_Toc400643205)

[Şekil 1.1.3.4.Band Durduran Filtre 1](#_Toc400643206)

[Şekil 1.1.3.1.Alçak Geçiren Filtre 1](#_Toc400643203)

[Şekil 1.1.3.2. Yüksek Geçiren Filtre 1](#_Toc400643204)

[Şekil 1.1.3.3 Band Geçişli Filtre 1](#_Toc400643205)

[Şekil 1.1.3.4. Band Durduran Filtre 1](#_Toc400643206)

**FIR GERÇEK ZAMANLI SES GÖSTERİMİ**

# ÖZET

Promizde mikrofon olarak gerçek zamanlı olarak alan ve farklı FIR filtreleriyle gösterimini gerçekleştirdik.Öncelikli olarak gerçek zamanlı ses sinyalimiz mikrofonun üzerinden alınarak ses üzerinde FIR, Adaptive FIR, FIR Quantitization uygulanarak farklı FIR filtrelerinin gösterimi amaçlanmıştır. Bir FIR filtresi, zaman alanında (örn., Eşleştirilmiş bir filtre) ve / veya frekans alanında (en yaygın olan) belirli özellikleri karşılayan katsayıları ve filtre sırasını bularak tasarlanır. Uyumlu filtreler, giriş sinyali ile bilinen darbe şekli arasında çapraz korelasyon gerçekleştirir. FIR konvolüsyonu, giriş sinyali ile impuls cevabının zamanla ters kopyası arasındaki karşılıklı korelasyondur. Bu nedenle, eşleştirilmiş filtrenin darbe tepkisi, bilinen darbe şeklini örnekleyerek ve bu örnekleri ters sırada filtrenin katsayıları olarak kullanarak "tasarlanmıştır".

**Anahtar Kelimeler:** FIR filtre, Alçak Geçiren, Yüksek Geçiren, Band Durduran, Band-Geçişli,Kovolüsyon,Frekans,Zaman-Uzayı,Niceleme,Adaptive

**fır Real time sound display**

# ABSTRACT

We realized real-time display as real-time microphone and demonstration with different FIR filters. Firstly real-time sound signal is obtained through our microphone and it is aimed to show different FIR filters by applying FIR, Adaptive FIR, FIR Quantization to the sound. An FIR filter is designed to find coefficients and filter ranges that meet certain properties in time domain (eg, a matched filter) and / or frequency domain (most common). The coherent filters perform cross correlation between the input signal and the known pulse shape. The FIR convolution is the reciprocal correlation between the input signal and the reverse copy of the impulse response over time. For this reason, the impulse response of the matched filter is "designed" by sampling the known pulse shape and using these samples in reverse order as the coefficients of the filter.

**Key Words:** FIR filter, Low Pass, High Pass, Stop Band, Pass-Band,Convulation,Frequency,Time-space,Quantization,Adaptive

# GİRİŞ

Sayısal filtre, sayısallaştırılmış analog sinyaller üzerinde çalışan, giriş sinyalini istenen çıkış sinyaline dönüştüren yöntem ya da algoritmadır. Uygulama alanları çok geniş olan filtrelerin başlıca tasarım amaçları arasında, karışmış sinyalleri birbirinden ayırmak, sinyaldeki gürültüyü azaltarak sinyal kalitesini arttırmak, ve bozulmuş sinyalin tekrar elde edilmesi olarak sayılabilir.

# filtreler

Filtreler belirli frekanslarda çalışan ya da belirli frekansları engelleyen, amaca göre belirli bant genişlikleri bulunan yapılardır.

## FIR FİLTRELER

Elektronikte sıkça kullanılan FIR filtreler geri beslemesi olmayan filtrelerin genel adıdır. Filtre çıkışı tamamen giriş sinyaline bağlıdır. Çıkışı kendi girişini beslemez.

C:\Users\Ynss\AppData\Local\Microsoft\Windows\INetCache\Content.Word\FIR FİLTRE.PNG

FIR’ın açılımı Finite Impulse Response (Sonlu Tepki Cevabı) şeklindedir.  Bu, filtrenin verilen bir impulse (tepki) sinyaline sonlu bir cevap vermesi demektir.  Sayısal bir FIR filtrenin genel denklemi aşağıdaki gibidir:

C:\Users\Ynss\AppData\Local\Microsoft\Windows\INetCache\Content.Word\FIR DENKLEM.PNG

Yani çıkış sinyali giriş sinyalinin geciktirilmiş değerlerinin belli katsayılarla çarpılıp toplanması ile elde edilir. Bu işleme “**convolution**” denir.

### FIR Filtre Kullanmanın Avantajları

IIR filtreye kıyasla;

* Kolayca "doğrusal faz" olarak tasarlanabilirler (ve genellikle bunlar). Basitçe söylemek gerekirse, lineer faz filtreleri giriş sinyalini geciktirir, ancak fazını bozmazlar.
* Bunları uygulamak kolaydır. Çoğu DSP mikroişlemcisinde, FIR hesaplaması tek bir talimatın döngüsü ile yapılabilir.
* Çok hızlı uygulamalar için uygundur. Çoklu oranla, "kenar boşluğu giderme" (örnekleme oranını düşürme), "enterpolasyon" (örnekleme oranını artırma) ya da her ikisini birden gerçekleştiriyoruz. Kesme veya enterpolasyon olsun, FIR filtrelerinin kullanımı bazı hesaplamaların yapılmasına izin vererek önemli bir hesaplama verimliliği sağlar. Aksine, IIR filtreleri kullanılıyorsa, bu çıktı atılacak olsa bile her çıktı ayrı ayrı hesaplanmalıdır (böylece geri besleme filtreye dahil edilir).
* İstenilen sayısal özelliklere sahipler. Uygulamada, tüm DSP filtreleri sonlu hassasiyetli aritmetik, yani sınırlı sayıda bit kullanılarak uygulanmalıdır. IIR filtrelerinde sonlu hassasiyetli aritmetik kullanımı, geribildirim kullanımı nedeniyle önemli sorunlara neden olabilir, ancak geri beslemesiz FIR filtreleri genellikle daha az bit kullanılarak uygulanabilir ve tasarımcı, ideal olmayan aritmetik ile ilgili çözmek için daha az pratik sorunlara sahiptir.
* Kesirli aritmetik kullanılarak uygulanabilirler. IIR filtrelerinden farklı olarak, 1.0'dan küçük katsayılarla bir FIR filtresi uygulamak her zaman mümkündür. (İstenirse, FIR filtresinin toplam kazancı çıkışında ayarlanabilir.) Sabit nokta DSP'leri kullanırken bu önemli bir husustur, çünkü uygulama daha basit hale getirir.

### FIR Filtre Kullanmanın Dezavantajları

IIR filtresine kıyasla;

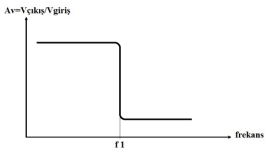
FIR filtrelerinin temel dezavantajı, özellikle düşük frekansa (örnekleme oranına göre) kesilmelere ihtiyaç duyulduğunda benzer netlik veya seçiciliğe sahip bir IIR filtresine kıyasla, genel amaçlı bir işlemcide oldukça fazla hesaplama gücünün gerekli olmasıdır. Bununla birlikte, birçok dijital sinyal işlemcisi FIR filtrelerini çoğu uygulama için IIR kadar etkili kılmak için özel donanım özellikleri sağlar.

### FIR FİLTRE ÇEŞİTLERİ

#### ALÇAK GEÇİREN FİLTRELER

Belirli kesim frekansından düşük olan frekansları geçiren,kesim frekansından yüksek olan frekansları da zayıflatarak iletimine engel olan filtrelerdir.

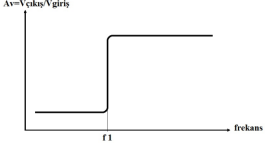
İdeal bir alçak geçiren frekans cevabı dikdörtgen şeklinde olmalıdır.



Şekil 1.1.3.1.Alçak Geçiren Filtre

#### YÜKSEK GEÇİREN FİLTRELER

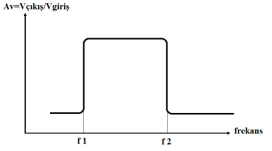
Kesim frekansının üstündeki frekansları geçirip altındaki frekansları sönümlendirerek iletimine engel olan filtrelerdir.Seri kapasiteler ve paralel bobinlerin kaskat dizilimi ile gerçeklenirler.



Şekil 1.1.3.2.Yüksek Geçiren Filtre

#### BAND GEÇİREN FİLTRELER

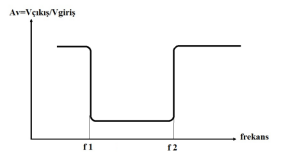
Band geçişli filtreler istenilen band aralığındaki frekansları geçirip diğer frekansları engelleyen filtre türüdür.



Şekil 1.1.3.3.Band Geçişli Filtre

#### BANT DURDURAN FİLTRELER

Belirli bir frekans aralığındaki sinyalleri zayıflatan devrelere bant durduran filtre devresi denir.

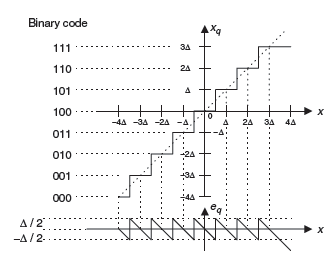


Şekil 1.1.3.4.Band Durduran Filtre

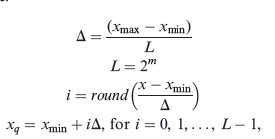
## FIR QUANTİZATİON

Sayısallaştırma, matematikte ve sayısal sinyal işlemede, büyük bir girdi kümesini (sayılabilir) daha küçük kümeye eşleme işlemidir. Yuvarlama ve kesme, nicemleme işlemlerinin tipik örnekleridir. Sayısallaştırma, bir sinyali dijital biçimde temsil etme süreci normal olarak yuvarlama işlemini içerdiğinden, neredeyse tüm dijital sinyal işlemelerinde bir dereceye kadar katlanır. Nicemleme, esas itibariyle tüm kayıplı sıkıştırma algoritmalarının çekirdeğini oluşturur.

Bir giriş değeri ile nicelenmiş değeri arasındaki fark (yuvarlama hatası gibi), nicemleme hatası olarak adlandırılır. Kuantizasyonu gerçekleştiren bir cihaz veya algoritmik fonksiyona bir nicemleyici denir. Bir analog-dijital dönüştürücü bir nicemleyici örneğidir.



Şekil 1.2.Bipolar Niceleyici

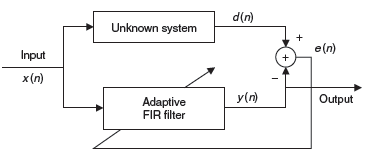


Şekil 3.2.Niceleme Formülü

Yukarıdaki formülde xmax ve xmin sırasıyla maksimum ve minimum değerlerdir. Analog giriş sinyali x. L sembolü, nicemleme seviyelerinin sayısını,burada m, kullanılan bit sayısıdır.ADC D sembolü nicemleyici veya ADC çözünürlüğünün basamak boyutudır.Son olarak, xq, nicemleme seviyesini, ve i, ikili kod.

## ADAPTİVE FİLTRE

Filtreler, işaret/gürültü oranını iyileştirecek şekilde işleme sokulur. Adaptif filtreler karakteristikleri, işaretin ve gürültünün zaman içinde değişen özelliklerine uydurulabilen, esnek yapıya sahip olan filtrelerdir. Ayrıca verilen performans indeksini en uygun duruma getirecek şekilde, parametrelerini gelen işarete göre adapte ederek filtreleme işlemini yerine getirirler. Ancak bu filtreler, öğrenme ve adaptasyon için bir başlangıç periyodu gerektirirler. Bu periyot içinde performansı tatmin edici değildir. Adaptif filtrede kullanılan performans indeksi, en az ortalama karesel hata (LMS) algoritmasıdır.

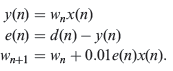


Şekil 1.3.Adaptive filtre sistem modeli



Burada wn şu an kullanılan katsayı, wn+1 elde edilen katsayısıdır LMS algoritmasından alınır ve bir sonraki giriş örneği için kullanılacaktır. 0.01 değeri katsayı değişim hızını kontrol eder. Göstermek için uyarlanabilir filtrenin konsepti olan LMS algoritması, katsayısı w0=0.3 olacak şekilde ayarlanır ve aşağıdakilere yol açar:

Buradaki w0 değerleri tahmini olarak verilmiştir.



Y(n) istenmeyen x(n) yani gürültü sinyalimizin LMS katsayısıyla çarpılarak FIR filtreden alınan çıktıdır.e(n) ise d(n) ile (gürültü ve alınan sinyalin toplamı) FIR filtreden alınan sonuç olan x(n) in farkı ile bulunan hata sinyalimizdir.Daha sonra bu sinyali kullanarak formülden de görüldüğü gibi bir sonraki LMS katsayımız bulunur.Bu şekilde adaptive filtremiz gerçekleştiriliyor.

# Projenin gerçekleştirilmesi

## Tanım

Promizde mikrofon olarak gerçek zamanlı olarak alan ve farklı FIR filtreleriyle gösterimini gerçekleştirdik.Öncelikli olarak gerçek zamanlı ses sinyalimiz mikrofonun üzerinden alınarak ses üzerinde FIR, Adaptive FIR, FIR Quantitization uygulanarak farklı FIR filtrelerinin gösterimi amaçlanmıştır

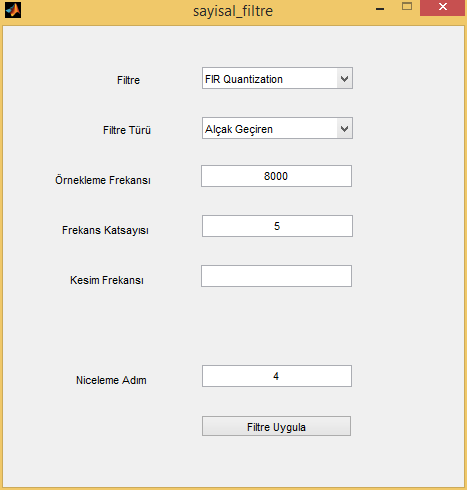
## UYGULANMASI VE ÇALIŞMASI

Projemizi matlab ile gerçekledik.Kullanıcı projeyi çalıştırıdığı zaman filtre seçimi yapabileceği bir ekran gelmektedir.Bu ekranda kullanıcı filtreyi filtre türünü seçtikten sonra FIR filtrenin yapacağı frekans filtrelemesi için gerekli olan frekans aralığını ve örnekleme frekansı ile FIR filtre katsayısını eğer FIR niceleme seçtiyse niceleme derecesini de seçtikten sonra buton ile mikrofondan gelen ana giriş sinyalini bu sinyalin frekans uzayını ve filtre sonucu ile bu sonucun frekans uzayını görebileceği bir ekranla karşılaşır.

### Proje Detayı

#### Arayüz

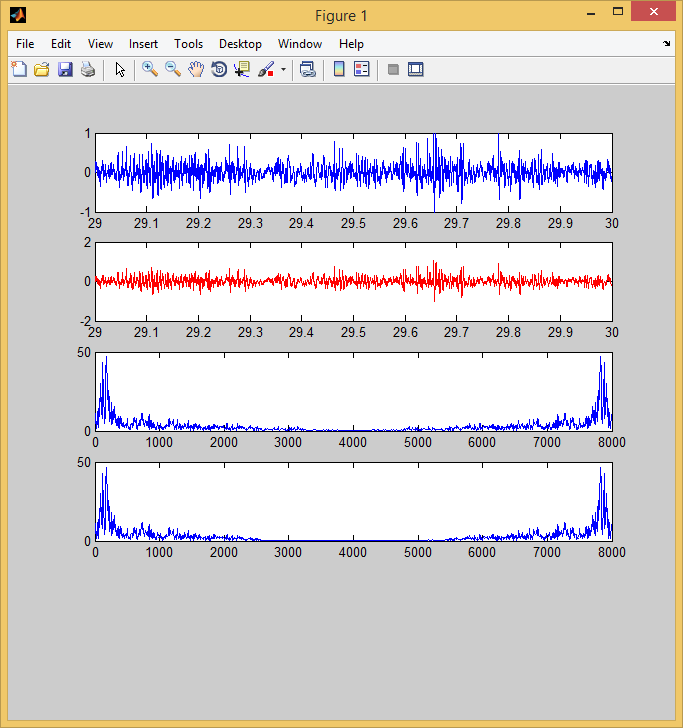
Arayüz matlab GUI ile gerçekleştirilmiştir. Arayüze ile kullanıcıdan bilgileri almaktayız.Arayüzümüzle filtre\_secim ,filtre\_tur\_secim olmak üzere iki tane pop-up menü bulunmaktadır.Bu menüler ile filtre türünü belirlemekteyiz.Ayrıca örnekleme\_frekans ile örnekleme frekansı frekans\_katsayi ile FIR katsayısı alt\_kesim\_frekans ile alt kesim frekasını ust\_kesim\_frekansı ile üst kesim frekansını ve niceleme\_adim ile FIR niceleme adımını bulmaktayız.Daha sonra alınan bilgiler buton click ile filtreleme işlemlerini yaptığımız myAudioRecording fonksiyonuna gönderilir.



#### Ses Sinyalinin Alımı ve İşlenmesi

Ses sinyalinin alımı ve işlenmesi myAudioRecording fonksiyonu ile yapılır.Bu fonksiyona arayüzden gelen örnekleme frekansı duraklama zamanı ve benzeri bilgiler geldikten sonra arayüzden seçilen filtre türüne göre bir switch case yardımıyla filtreleme işlemi gerçekleştirilip filtrelenen çıktı plot fonksiyonuyla gösterilip çıktının frekans uzayı bulunup daha buda gösterilmiştir.Ses sinyalinin gerçek zamanlı olarak alımı için bir timer kullanmaktayız.Ana fonksiyon ile mikrofondan ses alınırken bu timer sesin işlenmesi gerçekleştirilir.

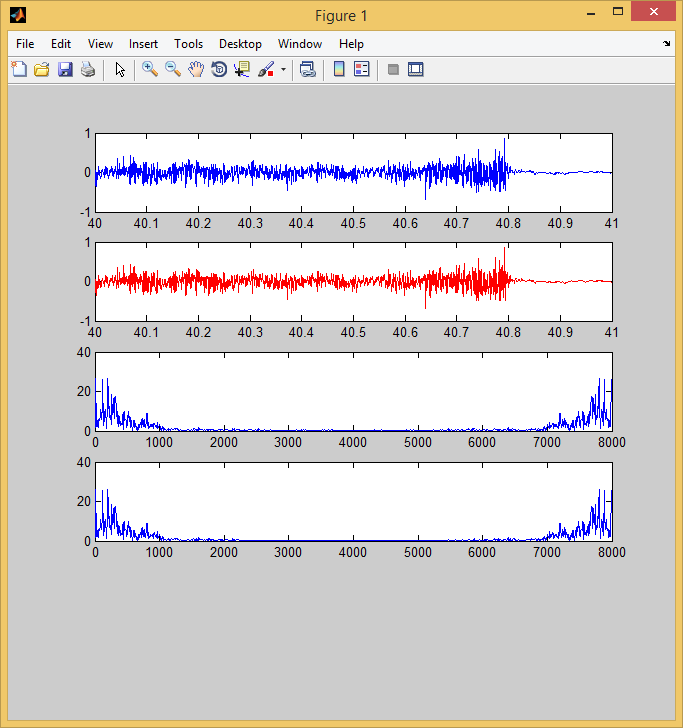
Sinyal işlenirken seçilen filtre türünü karşılayan filtre işlemini yapan fonksiyon çağırılır.



#### FIR filtreleme



Tasarlanan FIR fonksiyonu ile FIR filtreleme gerçekleştirilmiştir. Fonksiyon iki tane parametre almaktadır. Giriş sinyali ve FIR filtre katsayıları bu katsayılar diğer fonksiyonlar ile bulunmaktadır. FIR fonksiyonu yukarıdaki matematiksel işlemi gerçekleştiren algoritmayı gerçeklemektedir.



#### Alçak Geçiren Filtre Katsayılarının Bulunması

Alçak geçiren filtre katsayılarını alçak\_geciren fonksiyonu ile bulmaktayız.

C:\Users\Ynss\AppData\Local\Microsoft\Windows\INetCache\Content.Word\lp.png

Filtre katsayını fonksiyon şu şekilde bulmaktadır. Yukarıdaki denklem ile kullanıcı tarafından girilen örnekleme ve kesim frekansı ile normalize kesim frekansı aşağıdaki gibi hesaplanır.

normalize\_kesim\_frekansi=2\*pi\*(kesim\_frekansi/örnekleme\_frekansi)

normalize kesim frekansı bulunduktan sonra kullanıcı tarafından girilen filtre katsayisi nın 1 eksiği alınarak formüldeki M değeri bulunur ve M değerine göre bir for döngüsü yardımıyla her bir iterasyona karşılık gelen katsayı bulunarak bir dizi olarak return işlemi gerçekleştirilir.

#### Yuksek Geçiren Filtre Katsayılarının Bulunması

Yuksek geçiren filtre katsayılarını yuksek\_geciren fonksiyonu ile bulmaktayız.



Filtre katsayını fonksiyon şu şekilde bulmaktadır. Yukarıdaki denklem ile kullanıcı tarafından girilen örnekleme ve kesim frekansı ile normalize kesim frekansı aşağıdaki gibi hesaplanır.

normalize\_kesim\_frekansi=2\*pi\*(kesim\_frekansi/örnekleme\_frekansi)

normalize kesim frekansı bulunduktan sonra kullanıcı tarafından girilen filtre katsayisi nın 1 eksiği alınarak formüldeki M değeri bulunur ve M değerine göre bir for döngüsü yardımıyla her bir iterasyona karşılık gelen katsayı bulunarak bir dizi olarak return işlemi gerçekleştirilir.

#### Band Geçiren Filtre Katsayılarının Bulunması

Band geçiren filtre katsayılarını band\_geciren fonksiyonu ile bulmaktayız.



Filtre katsayını fonksiyon şu şekilde bulmaktadır. Yukarıdaki denklem ile kullanıcı tarafından girilen örnekleme frekansı alçak kesim frekansı ve yüksek kesim frekansı ile bunların normalize kesim frekansı aşağıdaki gibi hesaplanır.

dusuk\_normalize\_kesim\_frekans=2\*pi\*(dusuk\_kesim\_frekans/ornekleme\_frekansi);  
yuksek\_normalize\_kesim\_frekans=2\*pi\*(yuksek\_kesim\_frekans/ornekleme\_frekansi);

normalize kesim frekansları bulunduktan sonra kullanıcı tarafından girilen filtre katsayisi nın 1 eksiği alınarak formüldeki M değeri bulunur ve M değerine göre bir for döngüsü yardımıyla her bir iterasyona karşılık gelen katsayı bulunarak bir dizi olarak return işlemi gerçekleştirilir.

#### Band Durduran Filtre Katsayılarının Bulunması

Band geçiren filtre katsayılarını band\_durduran fonksiyonu ile bulmaktayız.



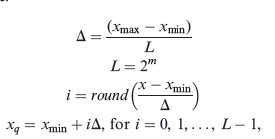
Filtre katsayını fonksiyon şu şekilde bulmaktadır. Yukarıdaki denklem ile kullanıcı tarafından girilen örnekleme frekansı alçak kesim frekansı ve yüksek kesim frekansı ile bunların normalize kesim frekansı aşağıdaki gibi hesaplanır.

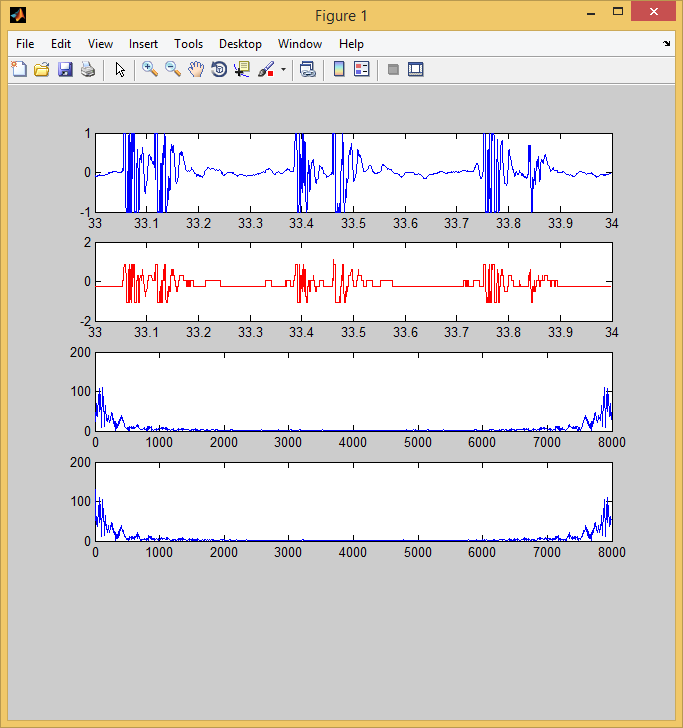
dusuk\_normalize\_kesim\_frekans=2\*pi\*(dusuk\_kesim\_frekans/ornekleme\_frekansi);  
yuksek\_normalize\_kesim\_frekans=2\*pi\*(yuksek\_kesim\_frekans/ornekleme\_frekansi);

normalize kesim frekansları bulunduktan sonra kullanıcı tarafından girilen filtre katsayisi nın 1 eksiği alınarak formüldeki M değeri bulunur ve M değerine göre bir for döngüsü yardımıyla her bir iterasyona karşılık gelen katsayı bulunarak bir dizi olarak return işlemi gerçekleştirilir.

#### FIR Niceleme

Niceleme işlemini niceleme fonksiyonu ile yapmaktayız microfon sinyali alınıp FIR filtre uygulandıktan sonra bu fonksiyon fir sinyali ve niceleme seviyesi ile çağrılır daha sonra aşağıdaki formüle göre niceleme seviyesi L=2bit formülü ile hesaplanmaktadır daha bulunan L dereri ile nicele adımı sinyalin maximum ve minimum noktaları baz alınp bulunduktan sonra niceleme adımı ile sinyal değeri çarpılarak normalize edilmemiş olan çıkış sinyal değeri elde edilir bu değerin yuvarlanması ile normalize edilmiş çıkış sinyali alınır bu işlemler bir for döngüsüyle her bir iterasyonda hesaplanarak nicelenmi sinyal fonksiyon tarafından döndürülür.

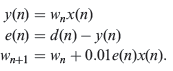




#### Adaptive FIR Filtreleme

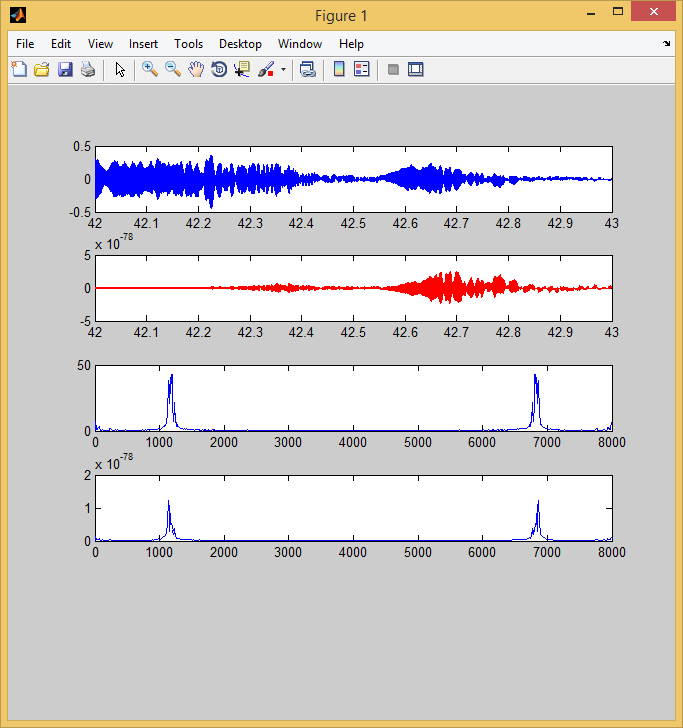
Adaptive FIR filtreleme işleme adaptive\_firlter fonksiyonu ile yapılmıştırfonksiyon giriş sinyali filtre katsayısı ve başlangıç tahmini alınmaktadır.

FIR filtre uygulaması sonucu elde edilen çıktı w oranımız ile aşağıdaki formüle uygun olarak çarpılıp her bir iterasyonda bu işlem tekrarlanır.



B sonraki w(n) değerimiz aşağıdaki formül ile hesaplandıktan sonra global olarak tanımlanmış w değerini güncellemek için bu değer fonsiyona bir return argümanı olarak verilir.

ap1



# SONUÇLAR

FIR filtre, FIR niceleme ve adaptive FIR uygulamasında belirtilen kaynaklardan yararlanılmıştır. Kaynaklardan anlaşıldığı kadarıyla uygulanmaya çalışılmıştır.Ve elde edilen sonuçlar gayet iyi bir düzeydedir.

# 

# KAYNAKLAR

1. <https://en.wikipedia.org/wiki/Infinite_impulse_response>
2. [*http://dspguru.com/dsp/faqs/fir/basics*](http://dspguru.com/dsp/faqs/fir/basics)
3. *“Lİ\_TAN ”DİGİTAL SİGNAL PROCESSİNG “Fundamentals and Applications”*
4. https://www.codeproject.com/Articles/1000084/Least-Mean-Square-Algorithm-using-Cplusplus
5. http://www.mcu-turkey.com/dijital-fir-filtre-uygulamasi/
6. http://shulgadim.blogspot.com.tr/2012/01/fir-filter-implementation.html
7. https://www.mathworks.com/help/fixedpoint/ug/convert-fir-filter-to-fixed-point-with-types-separate-from-code.html
8. https://sestevenson.wordpress.com/implementation-of-fir-filtering-in-c-part-1/
9. http://www.picproje.org/index.php?topic=55741.0
10. http://ptolemy.eecs.berkeley.edu/eecs20/week12/implementation.html
11. https://www.mathworks.com/matlabcentral/answers/164435-plotting-the-recorded-sound-while-recording
12. https://www.mathworks.com/help/matlab/import\_export/record-and-play-audio.html
13. http://homepages.udayton.edu/~hardierc/ece203/sound.htm
14. http://stackoverflow.com/questions/30101695/matlab-real-time-audio-processing
15. http://stackoverflow.com/questions/6681063/programming-in-matlab-how-to-process-in-real-time
16. http://www.fpganedir.com/ornek/fir\_filtre/index.php
17. http://learn.mikroe.com/ebooks/digitalfilterdesign/chapter/finite-impulse-response-fir-filter-design-methods/
18. http://docplayer.biz.tr/3693645-Deney-4-sayisal-filtreler.html
19. https://www.mathworks.com/help/signal/ref/fir1.htm