

**İSTANBUL TEKNİK ÜNİVERSİTE**  
**ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

**DSP İLE FMSD RADAR İŞARETİ İŞLEYEREK MESAFE BULMA**

**Volkan MERİÇ**

**Departman: Elektronik ve Haberleşme Bölümü**

**Bölüm: Elektronik Mühendisliği**

**Danışman: Dr. Bülent YAĞCI**

Mayıs 2010

## ÖNSÖZ

Kıymetli zamanı ve önerileri Dr. Bülent Yağcı ve Osman Ceylan'a sonsuz minnetlerimi sunarım. Ayrıca maddi ve manevi destekleri için aileme çok teşekkür ederim.

24 May 2005

Volkan MERİÇ

## **DSP İLE FMSD RADAR İŞARETİ İŞLEYEREK MESAFE BULMA (ÖZET)**

Bu çalışmada gerçek zamanlı sayısal işaret işlemci(DSP) ile frekans modülasyonlu sürekli dalga(FMSD) radarı kısa mesafe radarı işareti işleyerek mesafe ölçme işlemi gerçekleştirilmiştir. Önerilen frekansı modülasyonlu sürekli dalga radarı kısa mesafe, düşük güç tüketimli, ucuz ve portatif küçük radardır.

Çalışmada FMSD radarlarının teorik temeli ve formülleri verilmiştir. FMSD radarları frekans değişimini üçgen dalgaya göre yaparlar. Frekansı modüle edilmiş sinyal sürekli olarak antenden iletilir. Bu sinyal bir engele çarptığında geri yansır ve yansıyan sinyal anten yardımı ile alınır. Yansıyan sinyal ile iletilen sinyal karıştırıcıdan geçirilerek çarpılır ve çıkış sinyali elde edilir.

Ayrıca sayısal işaret işlemcileri hakkında genel bilgiler verilmiştir, kullanılan DSP'si ve geliştirme kartının mimarisi kısaca anlatılmıştır. DSP sayısal işaret işleme işleri için hızlı işaret işleme kabiliyetine sahip özel olarak tasarlanmış uygun mimaride mikroişlemcilerdir. Kullanılan DSP'sinin genel mimarisi, özellikleri, aritmetik işlem ünitesi, hafıza organizasyonundan da bahsedilmiştir.

FMSD radarını modellemek için MATLAB programı kullanılmıştır. Uygun radar parametreleri programa girilmiş ve matematik modeli gerçekleştirilmiştir. Uzaklık bilgisi Matlab'a verildiğinde, bilgisayar ses kartı çıkışında FMSD radar çıkışını üretmektedir. Bu radar çıkışı DSP'nin analog ses girişine verilmiştir.

FMSD radar çıkışında mesafe bilgisinin elde edilmesi için gerçek zamanlı sayısal işaret işleme metodu kullanılmıştır. DSP analog ses girişinden bilgisayarın ses kartından gelen radar çıkışını alıp, bu analog sinyali dijitale çeviriyor. Dijitale çevrilen sinyale Hızlı Fourier Dönüşümü(FFT) ve maksimum genlikli işaret tespit algoritması uygulanıyor. Maksimum genlikli işaretin frekansını tespitinden sonra mesafe bilgisi FMSD radar formüllerine göre elde ediliyor. Mesafe bilgisi bilgisayara RS 232 portu aracılığıyla gönderiliyor. Bilgisayar bu veriyi seri portundan alıp HyperTerminal programı aracılığıyla gösteriyor.

Sonuç olarak, Matlab modeli FMSD radar işaretini modelleyip üretmeyi başarmıştır. DSP'de bu sinyali gerçek zaman ölçütlerine uygun bir şekilde işlemiş göz ardı edilebilir bir hata ile sonucu yani mesafeyi elde etmiştir.

## 1. GİRİŞ

Mesafe ölçümü için kullanılan diğer sistemlerden ultrasonik dalga ile ölçümler ve kızılötesi ölçümler kısa mesafe için, lazer ile yapılan sistemlerden pahalı olmaktadır. Bu çalışmada, FMSD radar ile taşınabilir, ucuz, düşük hatalı, 0-50 metre gibi orta mesafe uzaklık ölçüm sistemi hedeflenmiştir. İşlemci olarak Analog Devices firmasının BF533 modeli Sayısal İşaret İşlemcisi(DSP) kullanılmıştır.

FMSD radarlar günümüzde geniş bir uygulama alanında kullanılmaktadır. Uçaklar için uçağın altına radar takılarak özellikle insansız hava araçları gibi araçlar için irtifa tespiti. Sıvıların, suyun, karın ve buzun manyetik geçirgenliğini farklı olmasından yararlanarak sıvı doluluk oranı ölçümü, kar ve buz kalınlığı tespiti yapılabilmektedir. Özellikle son yıllarda otomotiv endüstrisinde araç güvenliği arttırmak için takip mesafesinin korunması ve sürüş yardımı amaçları ile kullanılmaktadırlar.

Radar işaretinden mesafe bilgisinin elde edilmesi için Hızlı Fourier Dönüşümü(FFT) gibi işlemler yapılmaktadır. Bunun tür işaret işleme fonksiyonları yoğun miktarda çarpma toplama ve kaydırma içerdiğinden DSP bu işlem için en ideal çözümdür. DSP'ler aritmetik lojik üniteleri, MAC(multiply and accumulate), DMA(direct memory access) gibi özel üniteleri sayesinde tek saat darbesinde çarpma, toplama ve kaydırma, işlemciyi yormadan hızlı veri aktarımı gibi işlem kabiliyetlerine sahiptirler. Normal işlemciler ile bu yoğunlukta işlemleri gerçek zamanda yapmak olanaksızdır. Aynı işlemler FPGA'da da gerçekleştirilebilir fakat taşınabilir sistem için düşük güç tüketimi gereksinimi DSP'leri bu iş için daha uygun kılmaktadır.

Hali hazırda bulunan radarın istenen özelliklerde çalışmaması dolayısıyla MATLAB'ta FMSD radar modeli yazılmış ve başarılı bir şekilde gerçekleştirilmiştir. Ses kartı çıkışından DSP verilmek üzere radar çıkışı alınmıştır.

DSP üzerinde radar işareti gerçek zamanlı olarak işlenerek mesafe başarılı bir şekilde tespit edilmiştir. Mesafe bilgisi RS-232 bağlantısı ile bilgisayarın seri portuna gönderilerek mesafe bilgisi gözlemlenmiştir.

## 2. TEORİK BİLGİLER

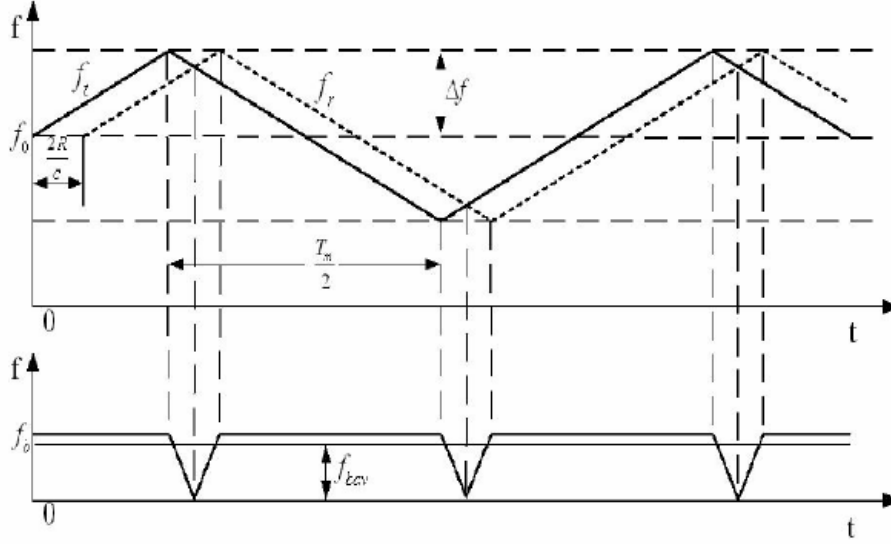
Raporun bu kısmında FMSD radarlarının çalışma prensipleri, formülleri ve mesafe tayini için gerekli işaret işleme yöntemi anlatılmış. Kullanılan DSP'nin mimarisi, hafıza organizasyonu ve geliştirme ortamına kısaca değinilmiştir.

### 2.2 FMSD RADAR

FMSD radarlar frekansı üçgen dalga ile değişen anteni ile sinyal yaymaktadır, bu sinyal ile anteninden algıladığı yansıyan sinyalin karıştırıcıdan(mixer) geçirilmesi ile radar mesafesi bilgisini içeren sinyal elde edilmektedir. Mesafe bilgisi baskın olan sinyalin frekansı ile saptanmaktadır[2].

#### 2.2.1 FMSD RADAR TEMELLERİ

Chirp olarak adlandırılan üçgen dalga şeklindeki frekans değişimli işaret sürekli olarak antenden iletilir. Yansıyan sinyal belirli bir zaman farkı ile gelir. Bu iki işaret ve bunların çarpılmasından oluşan işaret şekil 2.1'de gösterilmiştir[5].



**Şekil 2.1:** FMSD radar frekans karakteristiği

Elektromanyetik dalgalar ışık hızı ile hareket ettiğinden hedef ile radar arasındaki mesafe [1]:

$$\tau = \frac{2R}{c} \quad (2.1)$$

R: Uzaklık

C: Işık hızı

Çıkan periyotta işaret frekansı:

$$f_t = f_0 + \frac{\Delta f}{T_m/4} t \quad (2.2)$$

$f_0$  : Merkez frekans

$\Delta f$  : Frekans değişim aralığı

$T_m$ : Üçgen dalga periyodu

Yansıyan sinyalin frekansı

$$f_r = f_0 + \frac{\Delta f}{T_m/4} \left( t - \frac{2R}{c} \right) \quad (2.3)$$

Frekans farkı:

$$f_b = f_t - f_r = \frac{8R\Delta f}{T_m c} \quad (2.4)$$

Uzaklık:

$$R = \frac{c f_b T_m}{8 \Delta f} \quad (2.5)$$

$f_b$  : FMSD radar çıkışından elde edilen sinyalin baskın işaretinin frekansı

## 2.2.2 İŞARET İŞLEME YÖNTEMİ

İşaret işleme yönteminde öncelikle FMSD radarın çıkışı analogdan sayısalı çevrilmektedir. Kaç noktalı FFT alınacak ise o kadar örnek toplanıp FFT uygulanmaktadır. Gerçek zaman ölçütlerini sağlaması için FFT işlemi bir sonraki bitecek örnek toplama işlemine kadar bitmek zorundadır. FFT uygulandıktan sonra işaret frekans düzlemine geçmiştir. FFT çıkış verisi merkezine göre simetrik olduğundan optimizasyon için işaret sadece yarısı alınır. Daha

sonra gürültüden ve çıkış işaretinin harmoniklerinden dolayı belirli bir eşik değeri ile baskın işaret tespit algoritması uygulanmaktadır[4].

FFT işareti kullanılan nokta sayına bölmektedir, böylece işaretin frekansı:

$$f_b = n f_s / N \quad (2.6)$$

$n$  : Elde edilen işaretin örnekler içindeki sırası

$f_s$  : Örnekleme frekansı

$N$  : FFT nokta sayısı

Buradan mesafe ölçümü için radar parametreleri ile birlikte 2.5 denklemi uygulanır.

## 2.2 DSP

Sayısal işaret işlemciler sinyal işleme uygulamaları için geliştirilmiş özel mimarili işlemcilerdir. FFT, filtreleme, konvolüsyon, çarpım, kaydırma ve veri transferi gibi işlemleri hızlı bir şekilde yapabilmektedirler. Bu işlemler MAC (Multiply-accumulate), ALU (Arithmetic logic unit), circular buffer, DMA gibi üniteler yardımı ile yapılmaktadır.

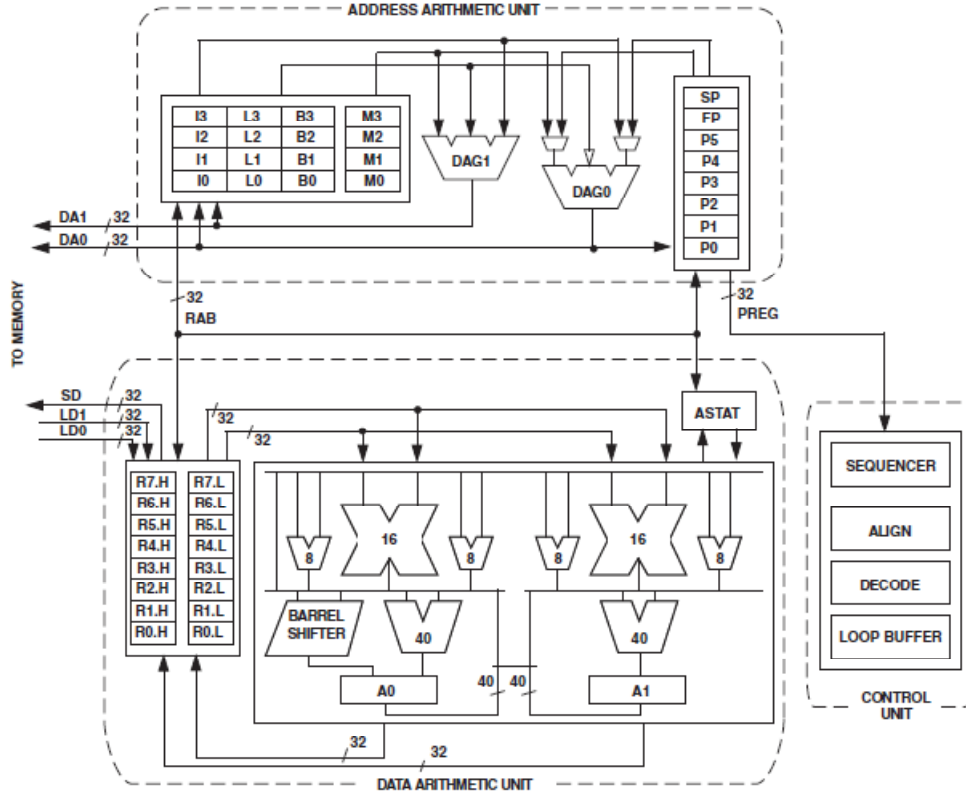
MAC tek saat darbesinde çarpma ve toplama işlemi yapmaya, circular buffer sürekli tekrar edecek şekilde katsayılar gibi sabitleri depolanmasına ve kaydırılmasına yarar, DMA işlemciyi yormadan veri aktarımını yapar.

DSP'ler genelde hız gerektiren işaret işleme uygulamaları için kullanılır; ses ve görüntü işleme, veri sıkıştırma ve aktarımı , ses tanıma, MRI, filtreleme, güvenli iletişim, radar ve sonar vb.

### 2.2.1 Donanım

Projemizde Analog devices firmasının değiştirilmiş Harvard mimarili BF533 modelli DSP'sini kullandık. Bu işlemcini özellikleri:

- 600 MHz saat hızı
- Sabit noktalı
- 16-bit MACs, iki 40-bit ALUs, dört 8-bit video ALUs, 40-bit shifter
- Programlanabilir voltaj regülatörü
- 148K bytes yonga üzerinde hafıza
- SDRAM, SRAM, flash, ve ROM destekleyen harici hafıza kontrolörü



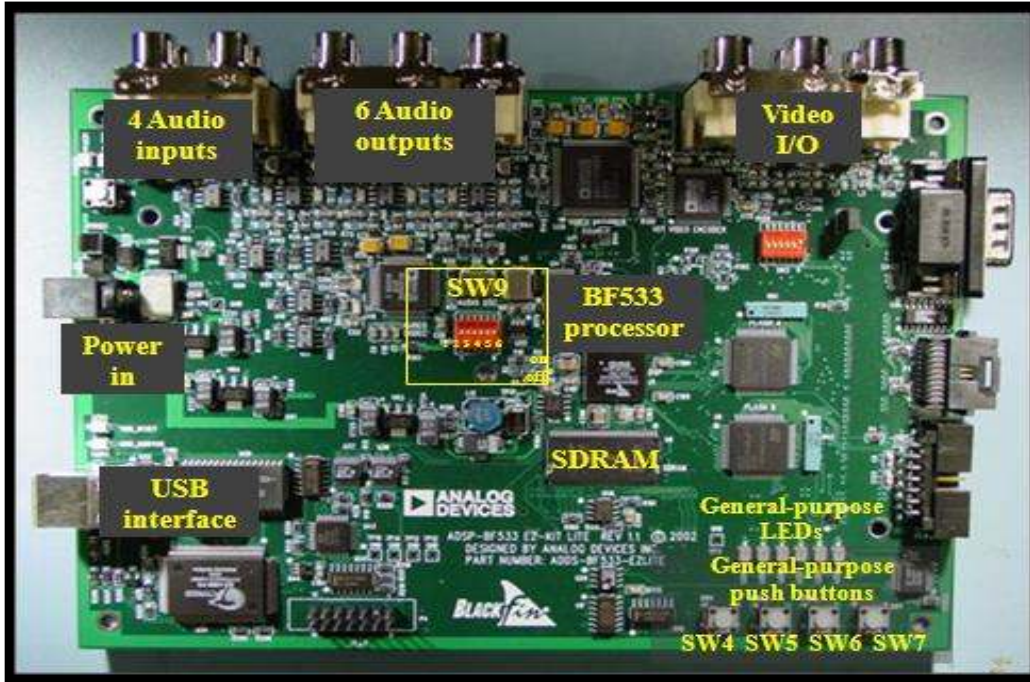
Şekil 2.2: Blackfin işlemci çekirdek şematığı

Şekil 2.2 Blackfin işlemcinin iç mimarisini göstermektedir. 16-bit iki MACs 16x16 bit çarpma işlemini tek saat darbesinde yapmaktadır. Aritmetik işlemler 16-bit ALU'da anında gerçekleştirilmektedir. 40 bitlik kaydırma işlemi gerçekleştirilebilmektedir. İki adet veri adres üreticisi(DAG) hafıza adreslerini üretmektedir [12].

### 2.2.1.1 Harici üniteler:

- Paralel iletişim arayüzü(PPI)
- 2 çift kanal, senkron seri port
- 8 stereo I2S kanalları
- 2 hafızadan hafızaya DMA
- 8 harici DMA
- SPI-portu
- PWM destekli üç 32-bit sayıcı/zamanlayıcı
- 16 genel amaçlı I/O pini (GPIO)
- UART
- Debug/JTAG ara yüzü

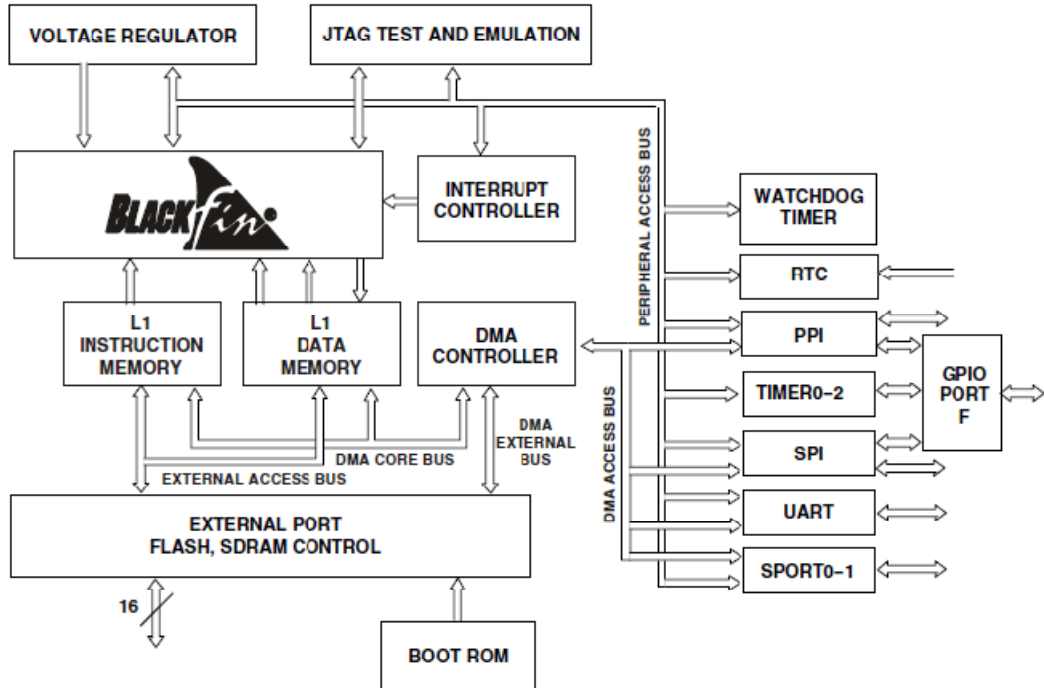
Geliştirme kartı olarak BF533 Ez-kit lite kullanıldı. Şekil 2.3 görülmektedir.



Şekil 2.3: ADSP BF533 Ez-Kit

### 2.1.2.2 Hafıza Mimarisi

Blackfin 32 bit adresleme kullanmaktadır. İşlemcinin dahili ve harici hafızası şekil 2.4'te gösterilmektedir.



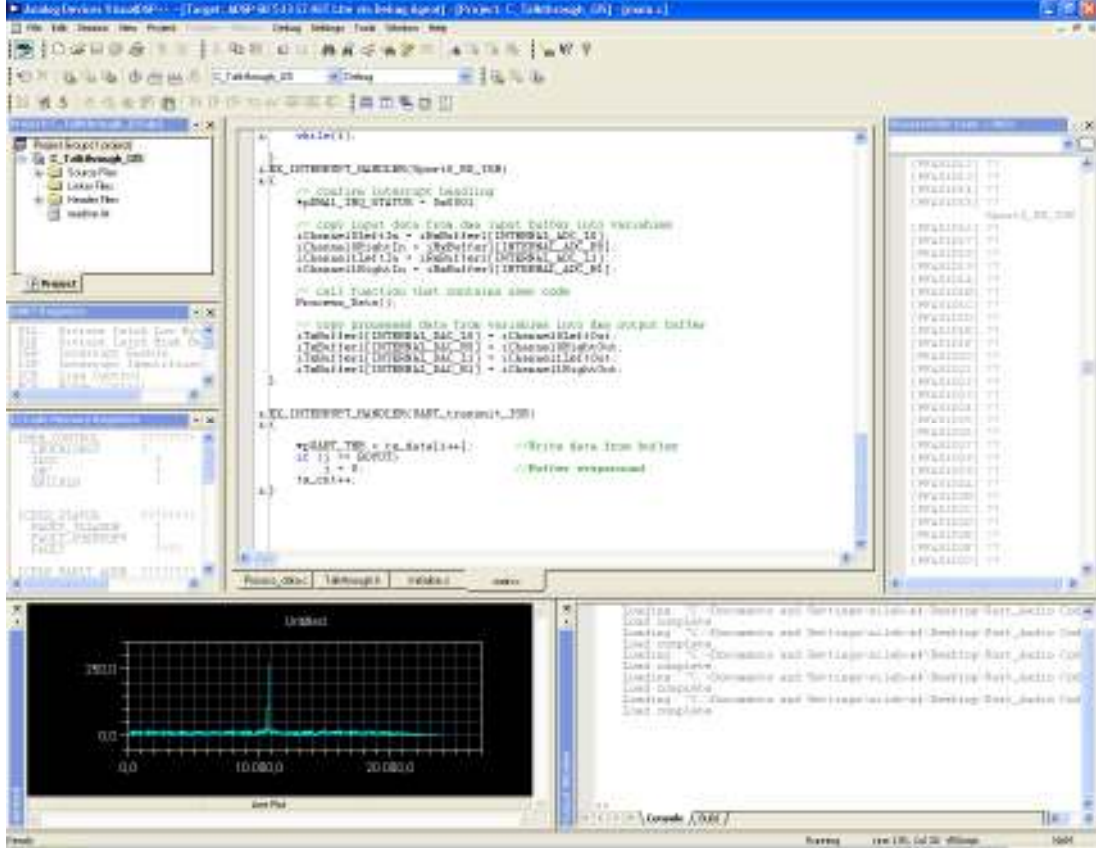
Şekil 2.4: Fonksiyonları ve hafıza diagramı

80K byte (birinci L1) program hafızası, 32 K byte SRAM (ikincisi L1) veri hafızası olarak kullanılmaktadır[10]. Çevre üniteleri ile veri transferi DMA aracılığı ile yapılmaktadır.



### 2.2.3 Yazılım ortamı

VisualDSP++ C/C++ ve assembly dilleri için Analog Devices firmasının yazılım geliştirme ortamıdır. Şekil 2.5 programın ana penceresi görülmektedir. VisualDSP++ “assembly”, C/C++ dillerini derleme, bağlama ve USB veya JTAG aracılığı ile yükleme işlemlerini gerçekleştirmektedir. VisualDSP++ ayrıca BTC (Background Telemetry Channel) aracılığı ile işlemcinin hafızalarından veri çekip çizdirme, kodu hafızaya veya işlem zamanına göre optimize etme, kaydedicileri, hafızayı, “pipelining”i gözlemlene gibi kod geliştirme özelliklerine sahiptir.



Şekil 2.5: VisualDSP++ ana ekranı

## 3. TASARIMIN GERÇEKLENMESİ

FMSD radar işareti kaynağı olarak MATLAB kullanıldı. Üretilen işaret ses kartı çıkında DSP kartına verildi. DSP sinyal işleme işlemlerini yürütüp mesafeyi tespit ettikten sonar bilgisayara seri port ile bilgisayara gönderip, Putty isimli terminal program aracılığı ile mesafe bilgisi gözlemlendi.

### 3.1 FMSD RADAR GERÇEKLENMESİ

Radar modeli M-file olarak Matlab'ta gerçekleştirildi. Radar parametreleri 0-50 metre için 0-20 KHz işaret üretecek şekilde seçildi. Sinyal üretici 26 GHz merkez frekanslı, 5 Hz'lik üçgen dalga ile değişen 2 GHz frekans değişimli sinyal üretilip uzakğa göre gecikme verildi ve çarpıcıya sokuldu. Çıkış işareti ses kartında BF533 alınmak üzere dışarı basıldı. Sistemin blok şeması şekil 3.1'de görülmektedir.

Matlab için radar parametreleri:

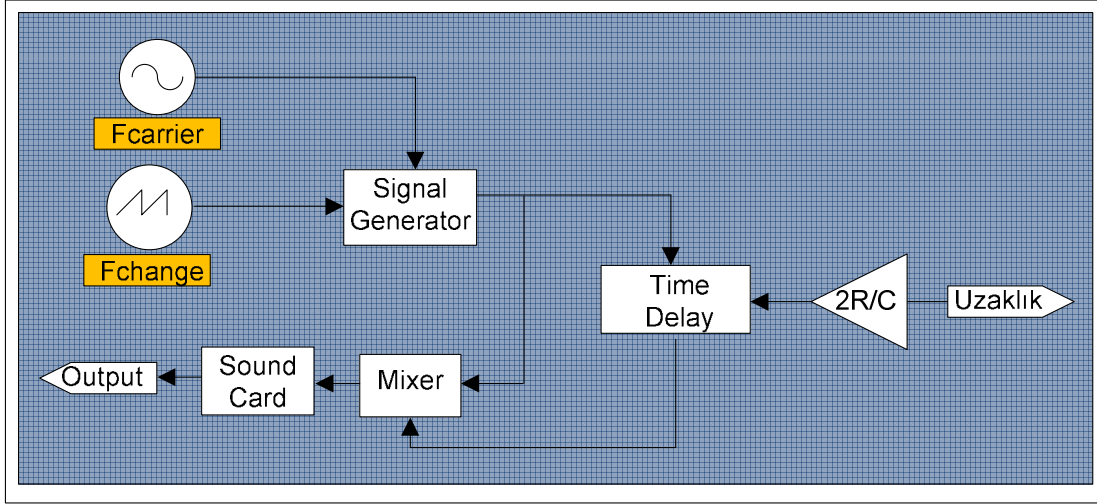
Merkez Frekans: 26 GHz

Lineer frekans değişimi: 2 GHz

Üçgen dalga frekansı: 5 Hz

Işık hızı:  $3 * 10^8$  m/s

Ses kartı örnekleme frekansı: 44100 Hz

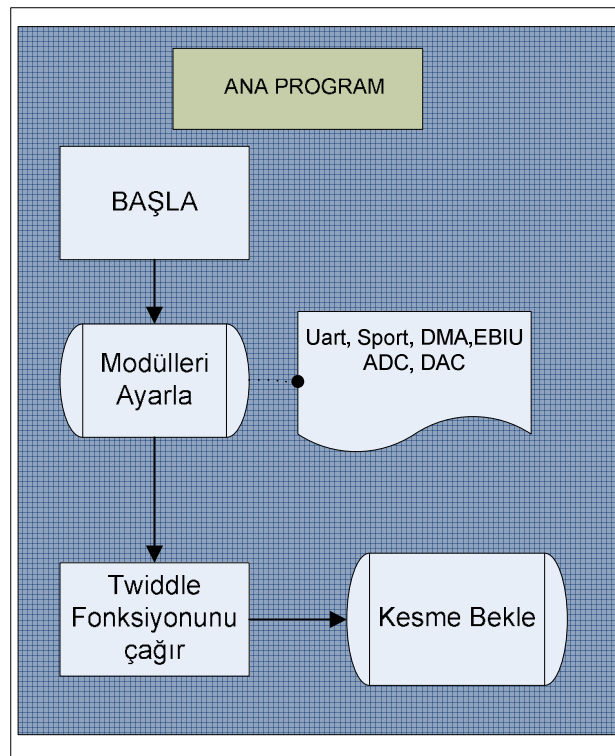


Şekil 3.1: FMSD radar Matlab modeli blok şeması

### 3.2 DSP GERÇEKLEMESİ

BF533 DSP bilgisayarın ses kartın gelen sinyali alıp AD836 ADC'si ile sayısalaya çevirmektedir. Daha sonra FFT'si alınıp, baskın frekans tespit algoritmasından geçirilip mesafe bilgisi bilgisayara uart ile gönderilmektedir. Program C dilinde yazılmıştır.

Ana program gerekli modüler için başlangıç ayarlamaları yapıp FFT fonksiyonunda kullanılacak katsayıları hesaplayan fonksiyonu çağırılmaktadır. Bu yapı şekil 3.4 te gösterilmiştir.



### Şekil 3.4: DSP ana fonksiyonunu blok diyagramı

AD1836 ayarlama işlemi SPI bağlantısı ile yapılmıştır. Bunun gerçekleşmesi için AD1836'nın Flash hafıza erişimi ayarlanmış ve reset pini bağlanmıştır.

SPI ile yapılan AD1836 başlangıç ayarları.

1. Reset
2. Port A 0x01 yazarak AD1836 izin ver
3. PF4(programmable flash)'ı yüksek yap. Bu pin geliştirme kartında AD1836 izin ver pinine bağlıdır.
4. SCLK 2 MHz
5. SPI portunu DMA yazma modu, 16-bit veri, en önemli bit ilk SPI master olarak ayarla.
6. DMA5'i iletim moduna al
7. DMA başlangıç adresi, döngü sayısı ve adres arttırım miktarını ayarla
8. DMA ve SPI'ya izin ver
9. Yapılana kadar bekle
10. SPI'yı kapat

DMA5'in başlangıç adresine AD1836 ayarlarını barındıran hafızanın başlangıç adresini yüklemekteyiz.

ADC'nin temel ayarları:

Örnekleme Frekansı: 48 KHz

Veri genişliği: 24 bit

İnen kenar reset

Data iletişim modu: I2S

Kazanç yok

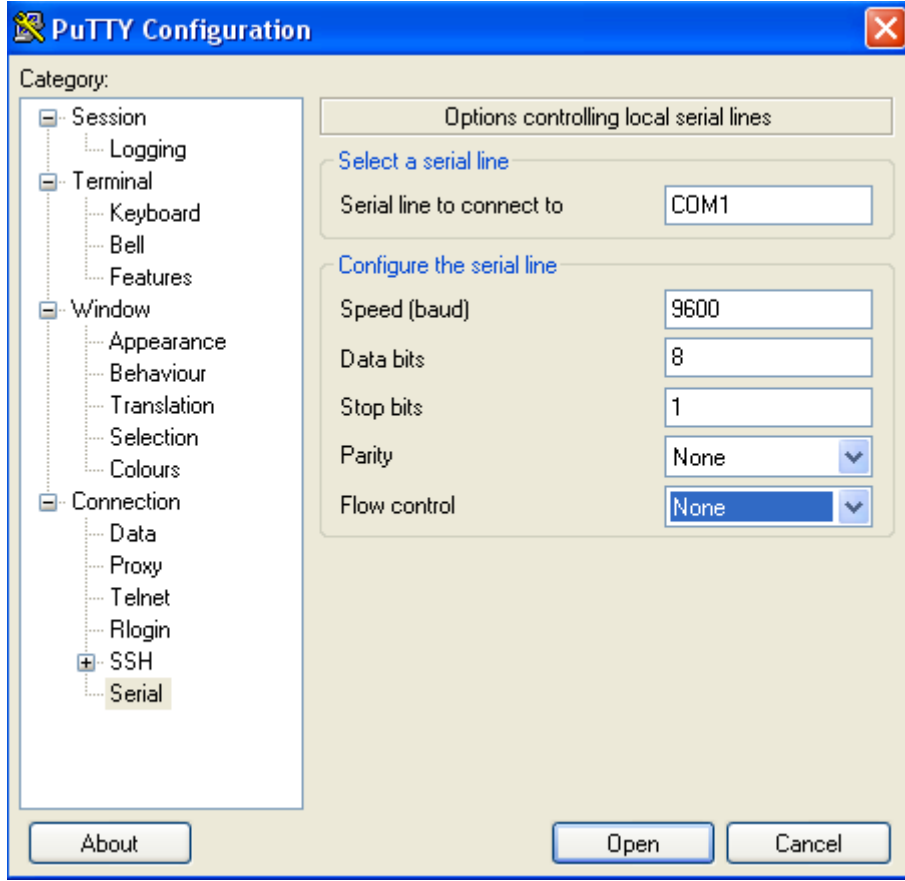
Filtre: DC, sadece işaretin AC bileşenlerini al

Sport kanal ayarlaması: AD1836 ile işlemci arasında veri alışverişi Sport kanalı ile yapılmaktadır. Temel ayarları; veri alıp verme modu, harici saat işareti, en yüksek anlamlı bit önce, 24 bit veri stereo frame senkronizasyonu.

DMA ayarlamaları: DMA1'i otomatik tampon modu ve veri alımı, DMA2 otomatik tampon ve veri gönderimi, 32 bit veri aktarımı. Veri aktarım bitiminde kesme oluşturacak şekilde ayarla ve izin ver.

Uart ayarlamaları: Uart ayarları doğru veri aktarımı için Putty program ile aynı seçilmiştir.

1. 9600 bound rate
2. 8 bit uzunluk
3. 1 bitiş biti
4. Parity biti yok
5. Uart kesmesine izin ver

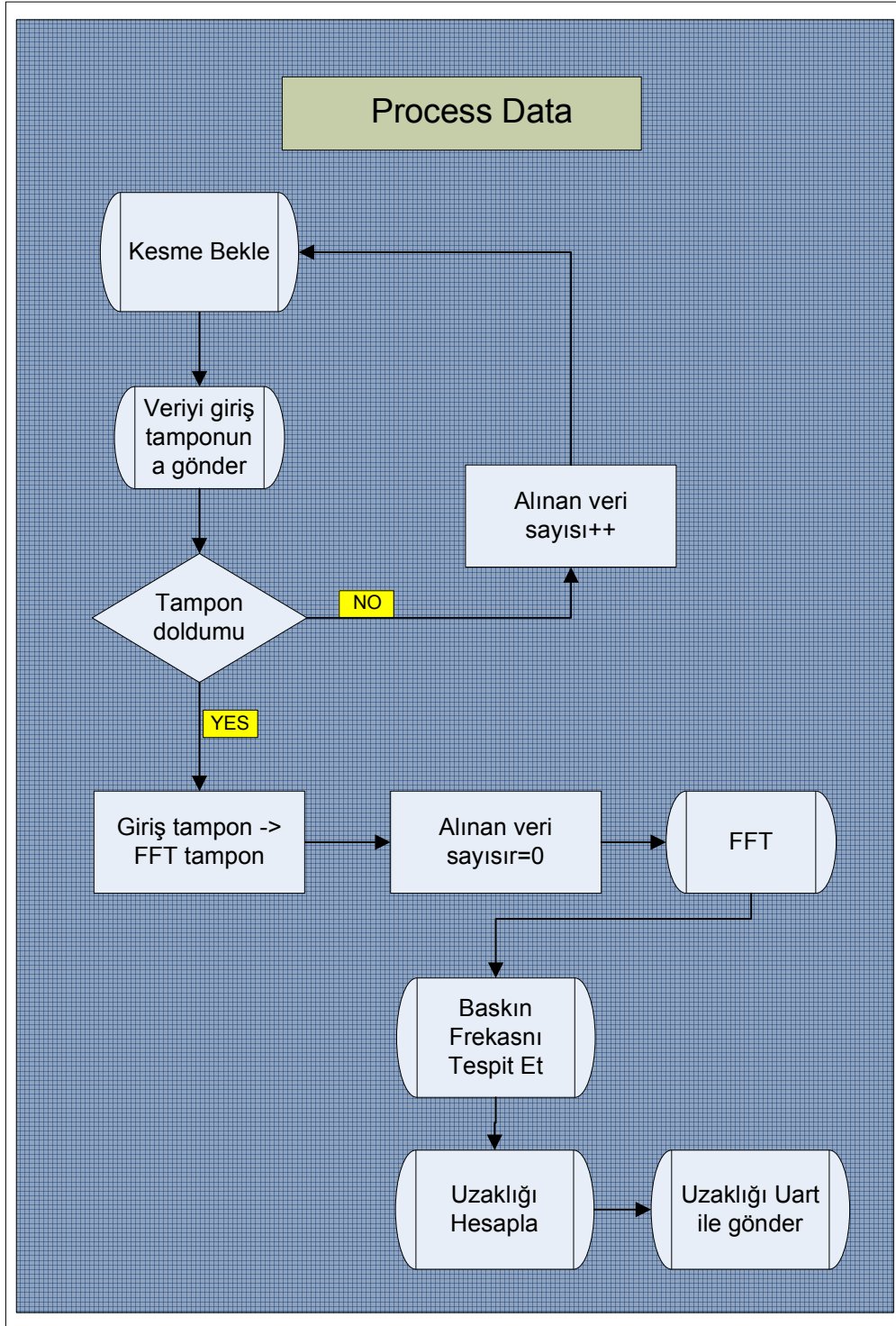


**Şekil 3.5:** Seri iletişim için Putty ayarlamaları

Kesmeleri ayarla: Sport veri alımı ve uart veri gönderimi kesmelerini yükle.

1. Sport0 RX (DMA1) kesme önceliği 2
2. Uart veri gönderimi (DMA7) kesme önceliği 3
3. Kesme vektörlerini ata  
Sport0 RX ISR -> IVG 9  
Uart veri gönderimi -> IVG 11
4. Sport0 RX ve Uart TX kesmelerine izin ver.

Ana fonksiyon işlemleri bittikten sonar DSP kesmeler için beklemektedir. Sport0 veri alım kesmesi Process\_data isimli fonksiyonu çağırılmaktadır. DMA1 ADC veriyi alıp kesme üretmektedir. Bu fonksiyon ADC verilerini toplamakta FFT işlemini, baskın frekans algoritması uygulamakta ve mesafeyi hesaplayıp uart ile göndermektedir. Fonksiyon içinde kesmeler açık olup kesme geldiğin veri alımını yapıp geri dönmektedir. Blok diyagramı şekil 3.6'da gözükmektedir.



Şekil 3.6: Proses\_data fonksiyonunun blok diyagramı

Blackfin sabit noktalı işlemcisi fract değişkeni ile optimize olarak çalışmaktadır. Process\_data ilk olarak 24 bitlik integer değişkenini 16 bit fract'a çevirmektedir. Daha sonra büyüklüğü FFT nokta sayısı 2048 ile aynı olan tampon sırayla yerleştirmektedir. Bu tampon dolduğunda bunu FFT tamponuna atıp 2048 noktalı FFT işlemini gerçekleştirmektedir. Fonksiyon içinde kesmeleri maskelemediğimizden dolayı eğer yeni veri gelirse bu kaybolmayıp giriş tamponuna konacaktır. 2048 noktalı FFT işlemini ve diğer işlemleri 2. Giriş tamponu dolmadığından bitirdiği için sistem gerçek zamanlı olarak çalışmaktadır.

FFT olarak radix-2 DIT algoritması kullanılmıştır. Bu algoritmanın çıkışı kompleks olduğundan ilk olarak absölütü alınmıştır. Böylece işaretin frekans dağılımı bir dizi içerisinde elimizde var. Bu dizi simetrik olduğunda sadece yarısını alıyoruz. Örnekleme frekasımız 48 KHz, FFT nokta sayımız 2048 olduğundan her bir veri aralığı 23.4348 Hz'e tekabül etmektedir.

$$f_s/N = 48000/2048 = 23.4375 \quad (3.1)$$

Daha sonra en büyük sayı algoritması FFT dizine eşik değer ile uygulanır ve dizideki numara frekans değerine basitçe aşağıdaki gibi çevrilir.

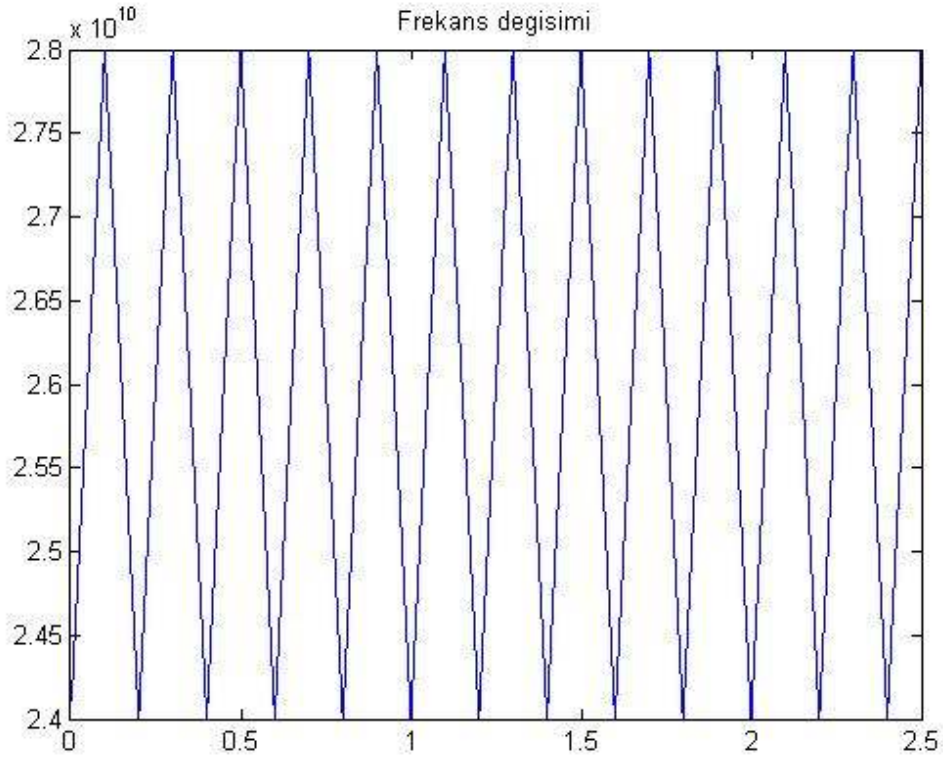
$$fb = 23.4375 * n \quad (3.2)$$

2.5 numaralı denklem ile de mesafe hesaplanır.

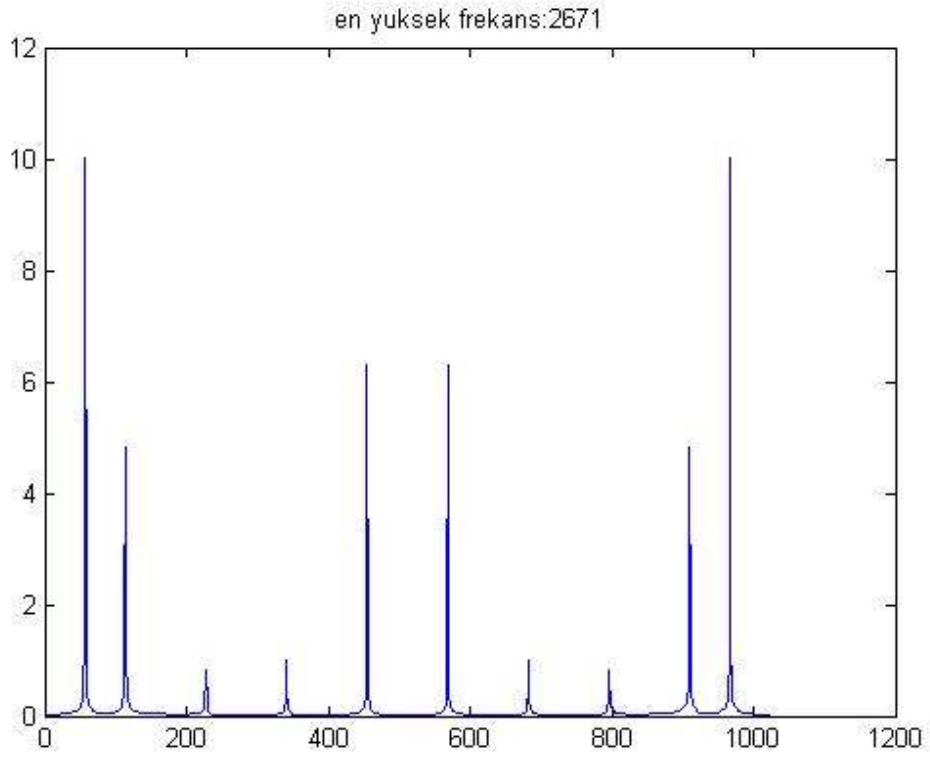
Uzaklık metreden santimetreye çevrilir ve bu dizini sonuna 'cm' ifadesi konularak uarttan gönderilir.

#### 4. SONUÇLAR

10 metre için Matlab çıkışları:



Şekil 4.1: Frekans-zaman değişiminin Matlab çizimi



Şekil 4.2: Frekans dağılımı

10 metre için DSP çıkışının sonuçları:



Şekil 4.3: Putty program görüntüsü

Tablo 4.1 Matlab'tan girilen uzaklık ile ölçülen uzaklıkları vermektedir.

**Tablo 4.1: Ölçümler**

<b>Girilen uzaklık (cm)</b>	<b>Ölçülen uzaklık(cm)</b>
10	10
50	49
100	98
500	497
1000	998
2500	2493
5000	4989

Hata formülü:

$$\sigma = \frac{|\text{Ölçülen} - \text{Gerçek}|}{\text{Gerçek}} \times 100 \quad (3.3)$$

En büyük hata:

$$\sigma_{MAX} = \frac{|98 - 100|}{100} \times 100 = \%2$$

Sistemin çözünürlüğü yaklaşık olarak 10 cm.

Not: Uarttan mesafe verisi gönderilirken önce integer sonra ASCİİ'ye çevrilmektedir. Hatanın önemli bir kısmını bu kapsamaktadır.

Bu projede, taşınabilir, ucuz orta mesafe ölçüm sistemi önerilmiştir. Uzaklık ölçümü için kullanılması önerilen FMSD radarım başarılı bir şekilde modeli oluşturulmuştur. Sistemin işaret işlemi kısmı gerçek zamanlı, gömülü bir ortamda DSP ile gerçekleştirilmiştir. Bu işlem için 2048 gibi çok noktalı bir FFT işlemi gerçekleştirilmiş, veri transferlerinin çoğu DMA'lar aracılığı ile yapılarak işlemci yükü azaltılmış ve kesmeler ile yürüyen optimize bir kod gerçekleştirilmiştir. Sonuç olarak DSP'li FMSD radarın orta mesafe ölçüm sistemleri için uygun olduğu gözlemlenmiştir.

Yurt dışından yüksek fiyatlara getirilen benzer sistemlerin Türkiye'de üretilmediğini fark ettim. Radarın gerçekleştirilmesi ve sistemin tek bir baskı devreye taşınması ile yaklaşık 200 TL gibi bir prototip maliyetine üretilebilir ve ülkemizin sırtından bir yük daha kaldırılabilir.



## KAYNAKLAR

- [1] **Hai Chen, Yan Li, Xin-min Wang**, 2007. Digital Signal Processing for A Level Measurement System Based on FMCW Radar, *IEEE International Conference on Control and Automation*, Guangzhou, CHINA, May 30-June 1, p. 2843- 2847
- [2] **A.G. Stove**, 1992, Linear FMCW radar techniques, 1992. *IEE PROCEEDINGS-F, Vol. 139, No. 5*, OCTOBER 1992, p. 343-350
- [3] **Jin Liu, Xianzhong Chen and Zheng Zhang**, 2006. A novel algorithm in the FMCW microwave liquid level measuring system, *MEASUREMENT SCIENCE AND TECHNOLOGY*, **17**, 135–138
- [4] **Wu Chang, Li Huan, Li Yubai**, 2006. A Practical FMCW Radar Signal Processing Method and Its System Implementation, *6th Intertional Conference on ITS Telecommunications Proceedings*, 2006, s. 1195-1199
- [5] **Graham M Brooker**, 2005. Understanding Millimeter Wave FMCW Radars, *1st International Conference on Sensing Technology*, Palmerston North, New Zealand, November 21-23, p. 152-157
- [6] **Krishnan S.**, 2000. Modeling and Simulation Analysis of FMCW Radar for Measuring Snow Thickness, *Master Thesis*, University of Kansas. Electronics and Communication Engineering, Kansas
- [7] **Douglas L. Jones**, 2006. Decimation-in-Frequency (DIF) Radix-2 FFT, Washington, USA.
- [8] **Panos Papamichalis**, 1997. An Implementation of FFT, DCT, and Other Transforms on the TMS320C30, APPLICATION REPORT: SPRA113, Waltham, Massachusetts
- [9] **Steven W. Smith**, 1997. The Scientist and Engineer's Guide to Digital Signal Processing. California Technical Pub, California.
- [10] **ADSP-BF533**, 2008. Datasheet, Analog Devices, USA.
- [11] **ADSP-BF533**, 2008. EZ-KIT Lite Evaluation System Manual, Analog Devices, USA.
- [12] **ADSP-BF533**, 2008. Hardware Refence, Analog Devices, USA.