Gömülü Sistemler ve Uygulamaları Sempozyumu Bildiri Kitabı

GÖMSİS 2012

İstanbul Teknik Üniversitesi
29-30 Kasım 2012

GÖMSİS 2012 - Gömülü Sistemler ve Uygulamaları Sempozyumu,
İstanbul Teknik Üniversitesi, 29-30 Kasım 2012

Kurullar

Düzenleme Kurulu

Müştak Erhan Yalçın, *İstanbul Teknik Üniversitesi* (Başkan)
Ramazan Yeniçeri, *İstanbul Teknik Üniversitesi*
Mehmet Akif Özkan, *İstanbul Teknik Üniversitesi*
Emre Göncü, *İstanbul Teknik Üniversitesi*

Program Kurulu

Ali Emre Harmancı. *İstanbul Teknik Üniversitesi*
Arda Yurdakul, *Boğaziçi Üniversitesi*
Günhan Dündar, *Boğaziçi Üniversitesi*
İlker Hamzaoğlu, *Sabancı Üniversitesi*
Müştak Erhan Yalçın, *İstanbul Teknik Üniversitesi*

Bilim Kurulu

Ahmet Onat, *Sabancı Üniversitesi*
Ahmet Özkurt, *Dokuz Eylül Üniversitesi*
Ali Emre Harmancı, *İstanbul Teknik Üniversitesi*
Alper Şen, *Boğaziçi Üniversitesi*
Anıl Çelebi, *Kocaeli Üniversitesi*
Arda Yurdakul, *Boğaziçi Üniversitesi*
A. Vedat Tavşanoğlu, *Yıldız Teknik Üniversitesi*
Bahri Atay Özgövde, *Galatasaray Üniversitesi*
Berna Örs Yalçın, *İstanbul Teknik Üniversitesi*
Betül Demiröz Boz, *Marmara Üniversitesi*
Cüneyt F. Bazlamaçcı, *Ortadoğu Teknik Üniversitesi*
Çağrı Tanrıöver, *Whizcomm*
Ece Guran Schmidt, *Ortadoğu Teknik Üniversitesi*
Eren Soyak, *Airties*
Eşref Adalı, *İstanbul Teknik Üniversitesi*
Faik Başkaya, *Boğaziçi Üniversitesi*
Feza Buzluca, *İstanbul Teknik Üniversitesi*
Gökay Saldamlı, *Boğaziçi Üniversitesi*
Günhan Dündar, *Boğaziçi Üniversitesi*
Haluk Rahmi Topçuoğlu, *Marmara Üniversitesi*
H. Fatih Uğurdağ, *Özyeğin Üniversitesi*
Hülya Yalçın, *İstanbul Teknik Üniversitesi*
İlker Hamzaoğlu, *Sabancı Üniversitesi*
İrfan Oyman, *Oyman*
İsmail Kadayıf, *Onsekiz Mart Üniversitesi*
Oğuz Ergin, *TOBB Üniversitesi*
Oğuz Tosun, *Boğaziçi Üniversitesi*
Osman Kaan Erol, *İstanbul Teknik Üniversitesi*
Özcan Öztürk, *Bilkent Üniversitesi*
Özlem Durmaz İncel, *Boğaziçi Üniversitesi*
Roy Küçükateş, *Ericsson*
Sarp Ertürk, *Kocaeli Üniversitesi*
Tuncay Uzun, *Yıldız Teknik Üniversitesi*
Yusuf Leblebici, *École Polytechnique Fédérale de Lausanne*

Önsöz

Bu bildiri kitabı, Gömülü Sistemler ve Uygulamaları Sempozyumu 2012 (GÖMSİS 2012)'de sunulacak akademik bildirilerin tam metinleri ile çağrılı konuşmacıların sunuşlarının özetlerini içermektedir.

İstanbul Teknik Üniversitesi Elektrik-Elektronik Fakültesi tarafından düzenlenen GÖMSİS 2012 aynı zamanda Boğaziçi Üniversitesi Bilgisayar ve Elektrik-Elektronik Mühendisliği Bölümlerinin katkılarıyla gerçekleşmiştir. Sempozyumumuz Çizgi TAGEM ve İstanbul Teknik Üniversitesi İleri Elektronik Teknolojileri Araştırma ve Geliştirme (İTÜ-ETA) Vakfı'nın maddi destekleri ile İstanbul Teknik Üniversitesi'nin sağladığı imkanlar sayesinde katılımcılarından katılım ücreti alınmadan düzenlenmiştir.

GÖMSİS 2012, Prof. Dr. Emre Harmancı'nın önderliğinde 2008 yılında başlatılan ve Gömülü Sistemler ve Uygulamaları alanında iki yılda bir yapılması planlanan sempozyumlar dizisinin üçünsüdür. Bu sempozyumların birincil amacı Üniversite - Sanayi işbirliği için bir köprü olmaktır. Bu amaçla, gömülü sistemler konusunda üniversitelerde ve sanayide yapılan Araştırma ve Geliştirme çalışmalarını bir çatı altında buluşturmayı hedeflemektedir.

Bu yıl da geçen yıllarda olduğu gibi sanayideki çalışmaları çağrılı konuşmalar, üniversitedeki çalışmaları bildiri ve poster sunumları olarak bir araya getirmektedir.

Sempozyuma toplam otuz bildiri başvurusu yapılmıştır. Bütün bildiriler Program Kurulu tarafından Bilim Kurulunun içinden seçilen üç hakeme gönderilmiştir. Program Kurulu hakem görüşlerini göz önüne alarak bu bildirilerden oniki tanesini sözlü sunuma kabul etmiş, dokuz tanesinin ise poster olarak sunulmasını uygun bulmuştur.

Bu yıl sempozyumda, sanayideki yapılan çalışmaları içeren sekiz adet davetli konuşma bulunmaktadır. Bu sunuşlarda işlenecek konular şunlardır:

- Gömülü Sistemlerde FPGA Kullanımı,
- Mikroişlemci Entegrasyonu ve ARM Ailesi,
- FPGA Bazlı Kamera Tasarımı,
- XMC Uyumlu Esnek Modüler ve Genişletilebilir FPGA Kartı Tasarımı,
- Register Transfer Level (RTL) Tasarım,
- Gömülü Video Sistemleri: Mimariler ve Tasarım Yaklaşımları,
- IP ve Kırmıküstü Sistem Doğrulaması,
- MATLAB & Simulink Algoritmalarının FPGA Üzerinde Gerçeklenmesi.

GÖMSİS 2012'nin düzenlenmesindeki katkılarından dolayı Program Kurulu üyelerine, özellikle bildirilerin değerlendirilme sürecindeki katkılarından dolayı Bilim Kurulu üyelerine en içten teşekkürlerimi sunarım. Sempozyumumuza bildiri gönderen bütün yazarlara, sanayideki çalışmalarını bizimle paylaşan davetli konuşmacılara ve sempozyumun düzenlenmesinde özveri ile çalışan Düzenleme Kurulu üyelerine de teşekkürlerimi sunarım.

Doç. Dr. Müştak E. Yalçın,
GÖMSİS 2012 Düzenleme Kurulu Başkanı
Kasım 2012

Program ve İçindekiler

1. Gün, 29 Kasım 2012

2. Gün, 30 Kasım 2012

Gömülü Sistemlerde FPGA Kullanımı

Ali Erkin Arslan, Yüksel Serdar

Aselsan A.Ş.
Mikroelektronik Güdüm ve Elektro-Optik Grubu
Aselsan Akyurt Tesisleri, Ankara
aearslan@aselsan.com.tr serdar@mgeo.aselsan.com.tr

Özetçe

ASELSAN, askeri elektronik alanında geliştirdiği ürünler ile ülke savunmasında teknolojik ihtiyaçları karşılamakta en önemli rolü üstlenmektedir. ASELSAN'ın geliştirdiği aviyonik, navigasyon, termal görüntüleme, hedef tespit ve nişangah sistemlerinin performans, hız, güvenilirlik ve askeri çevre şartlarında görev yapma özelliklerine sahip olmasından dolayı; bu sistemler üzerinde esneklik, güvenilirlik, paralel ve gerçek zamanlı işlem yapma yeteneklerine sahip FPGA entegreleri içeren tümüyle ASELSAN tasarımı özgün gömülü sistem kartları kullanılmaktadır. Bu sunumda ASELSAN tecrübeleri temel alınarak gömülü sistemlerde FPGA kullanım alanları, özellikle görüntü ve video algoritmalarının FPGA ile gerçeklenmesi, tasarım örnekleri, sayısal tasarım geliştirme ve doğrulama yaklaşımı ve süreçleri, RTL (Register Transfer Level) seviyesi tasarım girişi ve gelişmekte olan ESL (Electronic System Level) tasarım konularında bilgi verilecektir.

Ali Erkin Arslan Hakkında
Ali Erkin Arslan, Orta Doğu Teknik Üniversitesi Elektrik-Elektronik Mühendisliği bölümünden 2000 yılında mezun olmuştur. 2003 yılında yüksek lisans ve 2012 yılında doktora derecelerini yine aynı üniversiteden almıştır. 2000 yılından beri ASELSAN Mikro-Elektronik, Güdüm ve Elektro-Optik Grubu Elektronik Tasarım Müdürlüğü'nde çalışmakta olup halen Sayısal Tasarım Lideri olarak görev yapmaktadır. Çalıştığı konular sayısal gömülü kart ve FPGA tasarımları, video ve görüntü işleme algoritmaları, hedef izleme ve istatistiksel olarak doğrusal ve doğrusal olmayan sistemlerde kestirme yöntemleridir.

Yüksel Serdar Hakkında
Yüksel Serdar, Orta Doğu Teknik Üniversitesi Elektrik-Elektronik Mühendisliği bölümünden 1994 yılında mezun olmuş ve 1998 yılında aynı üniversiteden yüksek lisans derecesi almıştır. 1994 yılında ASELSAN Mikro-Elektronik, Güdüm ve Elektro-Optik Grubu Termal Sistemler Tasarım Müdürlüğü'nde sayısal tasarım mühendisi olarak çalışmaya başlamıştır. 2004 yılında Elektronik Tasarım Müdürlüğü'nde Kıdemli Uzman Mühendis olarak çalışmaya başlamış olup, Nisan-2011 tarihinden itibaren Elektronik Tasarım Müdürü olarak görev yapmaktadır. Çalıştığı konular sayısal gömülü kart ve FPGA tasarımları, termal görüntüleme ve aviyonik sistemler tasarımları, sistem mühendisliği ve proje yönetimidir.

Mikroişlemci Entegrasyonu ve ARM Ailesi

Sinan Topçu

Ericsson Microelectronics Design Center (EMDC)
İTÜ Teknokent ARI2 B Blok
Maslak, İstanbul
sinan.topcu@ericsson.com

Özetçe

- Akıllı sistemler ve Ericsson vizyonu
- ARM Mikroişlemci Ailesi
- Gömülü sistemler ve EMDC
- Mikroişlemci Alt Sistemleri ve AMBA (Advanced Microcontroller Bus Architecture)
- Cortex-M3 Entegrasyonu Projesi

Sinan Topçu Hakkında
Sinan Topçu, 1981 yılında Ankara'da doğdu. 2003 yılında Bilkent Üniversitesi Elektrik ve Elektronik Mühendisliği bölümünden mezun oldu. 2005 yılında Applied DSP and VLSI Research Group, Department of Electronics, Communication and Software Engineering, University of Westminster'da yüksek lisans eğitimin tamamladı. 2007 yılında STMicrolelectronics İstanbul Tasarım Merkezinde sayısal tasarım mühendisi olarak çalışma hayatına başladı. Halen Ericsson Microelectronics Design Center (EMDC) bünyesinde RTL tasarım, doğrulama ve IP seviyesi entegrasyon alanlarında çalışmaya devam etmektedir. Kendisi boş zamanlarında Taekwondo, Muay Tai ve Brazilian Jiu-Jitsu (BJJ) sporlarıyla ilgilenmektedir.

FPGA Bazlı Kamera Tasarımı

Çağlar Kalaycıoğlu

CMOSVision Ltd. Şti.
GOSB Teknopark
Gebze, Kocaeli
caglar.kalaycioglu@cmosvision.com

Özetçe

Bu konuşmada CMOSVision firmasında yapılan FPGA bazlı kamera tasarımları konusunda kısaca bilgi verilip sonrasında örnek olarak 25 MegaPixel saniyede 53 çerçeve veren Condor kamera tasarımı tanıtılacaktır.

Çağlar Kalaycıoğlu Hakkında
Lisans ve yüksek lisans derecelerini Sabancı Üniversitesi'nden alan Kalaycıoğlu 2010 yılında çalışma hayatına CMOSVision firmasında ASIC/FPGA Tasarım Mühendisi olarak başlamıştır. Bu firmada çalıştığı süre zarfında X-Ray kamera, line kamera ve 25MegaPixel Condor kamera calışmalarını başarıyla devam ettirmenin yanısıra insansız hava aracı için FPGA tasarımı çalışmalarında da bulunmuştur. Çalışmaları sırasında şirketin kod yazım kuralları gibi birçok konuda efor sarfetmiş olan Çaglar Bey şirket içerisinde takım çalışmasının gelişmesini sağlayan roller de üstlenmiştir.

XMC Uyumlu Esnek Modüler ve Genişletilebilir FPGA Kartı Tasarımı

Hakan Sakman

Ericam Görüntüleme ve Savunma Teknolojileri A.Ş.
GOSB Teknopark Hi-Tech Binası
Gebze, Kocaeli
hakan.sakman@cmosvision.com

Özetçe

Bu konuşmada Ericam firmasında geliştirilmiş olan XMC uyumlu bir FPGA kartının esnek ve genişletilebilir bir şekilde nasıl tasarlanmış olduğu ve bu kartın nasıl kullanılabileceği anlatılacaktır.

Hakan Sakman Hakkında
Lisans derecesini İTÜ'den alan Sakman, yüksek lisans derecesini Southampton Üniversitesi'nden almıştır. 1993 yılında Netaş firmasında yine ASIC tasarımı mühendisi olarak çalışmalarına devam etmiştir. 2004 yılında Türkiye'ye dönen Sakman, Tübitak UEKAE'de bir yıl FPGA mühendisi olarak çalıştıktan sonra CMOSVision firmasının Türkiye ofisini kurarak genel müdürlüğünü üstlenip pekçok kamera tasarım projesinin yürütücülüğünğ yapmıştır. 2012 yılı başında Ericam firmasının kuruluşunu gerceklestiren Sakman bu firmanın Ar-Ge'den sorumlu Yönetim Kurulu Üyesi görevini de üstlenmiştir.

GSM Ses Kanalından Sayısal Veri İleten Bir Modemin Tasarımı ve Gerçeklenmesi

Sercan Tunçay, Mehmet Akif Özkan ve Berna Örs Yalçın
İstanbul Teknik Üniversitesi
Elektronik Mühendisliği Bölümü
Maslak, 34469, İstanbul
e-posta: { tuncayse, akif.ozkan, siddika.ors }@itu.edu.tr

Özetçe—**Bu çalışmada GSM ses kanalını sayısal veri iletiminde kullanılabilir hale getirmek için bir modem tasarımı yapılmıştır. Modem tasarımında faydalanılan yöntem dizisi açıklanarak sistem, Spartan-6 üzerinde MicroBlaze ve yardımcı özel tasarım donanımlar yardımıyla yazılım ve donanım ortak bir çalışma sonucu gerçeklenmiştir. Yapılan testler sonucunda karşılaşılan problemler değerlendirilerek performansın iyileştirmesi için sisteme yazılımsal ve donanımsal geliştirmeler uygulanmış, gelecek çalışmalar için çeşitli çözüm önerilerinde bulunulmuştur.**

I. GİRİŞ

Mobil Haberleşme için Küresel Sistem (Global System for Mobile Communications - GSM) günlük hayatta sivil ve askeri uygulamalarda yaygın olarak kullanılır hale gelmiştir. GSM ses kanalı yalnızca insan konuşmasının aktarımı için özelleşmiştir ve bu kanal üzerinden şifreli bir konuşma veya resim dosyası gibi insan sesinin doğal yapısına benzerlik göstermeyen verileri iletmek mümkün değildir [1-3].

Bu çalışmada, Doğrusal Öngörülü Kodlama (Linear Predictive Coding - LPC) temelinde insan sesine benzeyen sentetik konuşmalar üreten bir teknik temel alınmıştır [2]. Böylece GSM ses kanalı, tasarlanan modemin ürettiği bu yapay sesler yardımıyla sayısal veri iletimine uygun hale getirilmiştir .

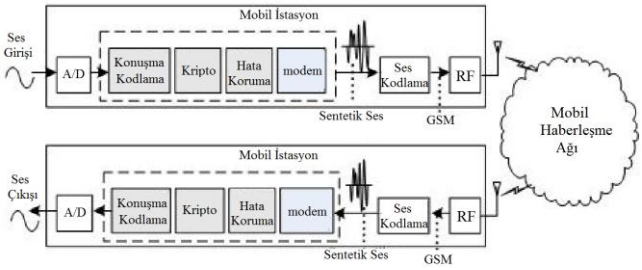
Sistemin tüm adımları öncelikle C dilinde algoritmik olarak gerçeklenerek test edilmiş ve işlem süreleri göz önünde bulundurularak performansın arttırılması için yazılımsal ve donanımsal çözümler geliştirilmiştir.

Bildirinin ikinci bölümünde insan sesinin modellenmesinden bahsedilerek tasarımda faydalanılan modemin GSM ağı üzerindeki yeri, dayandığı teknik ve iç mimarisi açıklanmıştır.

Üçüncü bölümde modemin gerçekleme aşamasında kullanılan donanımlar ve yazılımlardan bahsedilip elde edilen öncül test sonuçlarının değerlendirilmesi yapılarak, sistemin gerçek zamanlı çalışabilmesi için uygulanan performans iyileştirmeleri açıklanmıştır.

II. MODEM

GSM ses kanalı üzerinden şifreli bir konuşma veya bir resim dosyası gönderilmek istendiğinde, bu veriler insan sesine benzemediğinden dolayı iletim bandının verimli kullanılabilmesi için GSM ağı tarafından bastırılacaktır[3,4]. İletilmesi arzu edilen sayısal verilerde meydana gelecek bu bastırılma yani kayıplar, Şekil-1'de gösterilen şemadaki mobil cihaz ve kablosuz mobil haberleşme ağı içerisinde bulunan ses kodlayıcıların sesi sıkıştırmasının bir sonucudur.



Şekil-1: GSM ağı üzerinden güvenli iletişim şeması [3].

Bu sebeple sayısal veriler mobil cihaz içerisinde ses kodlayıcısı tarafından sıkıştırılıp radyo frekansında (Radio Frequency - RF) baz istasyonuna ses kanalı üzerinden gönderilmeden önce özel bir yöntemle yeniden kodlanmalı ve insan sesine benzetilmelidir.

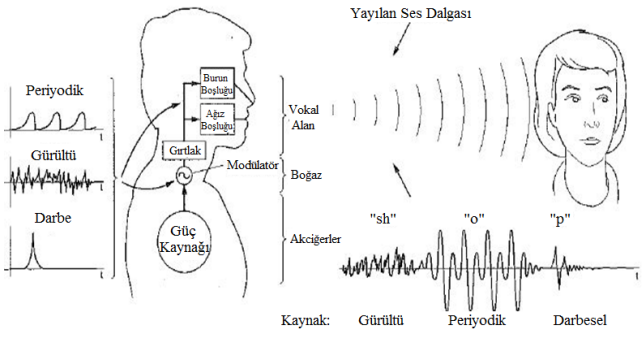
Bu bölümde kısaca insan sesinin modellenmesi, özel kodlama için seçilen modemin dayandığı teknik ve iç mimari açıklanacaktır.

A. Doğrusal Öngörülü Kodlama

LPC, haberleşme alanında iletim bandından tasarruf sağlamak için belirli bir ses zarfının kodlanarak sıkıştırılması, böylece gönderilecek veri boyutunun küçültülmesinde kullanılır [5,6]. Bu sıkıştırma için (1) ile verilen denklemde gösterildiği gibi o anki değer, daha önceki değerler yardımıyla ifade edilmeye, bir başka deyişle öngörülmeye çalışılır.

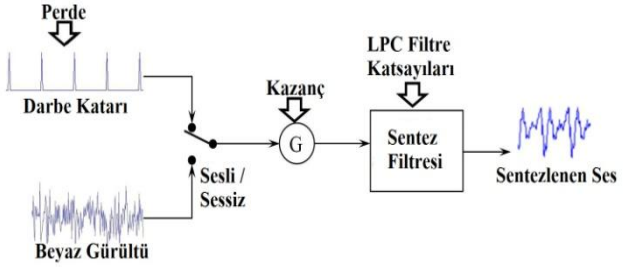
$$s(n) \approx a_1 * s(n-1) + a_2 * s(n-2) + \cdots \\ \ldots + a_p * s(n-p) \qquad (1)$$

Şekil-2'de gösterildiği gibi insanda sesin yapısı sesli, sessiz ve darbesel sesler olmak üzere üç ana alt başlıkla incelenebilir [7]. Bu 3 yapı incelendiğinde sessiz harflerin



Şekil-2: İnsanda sesin modellenmesi [7].

beyaz gürültüye, sesli harflerin sessiz harflere göre daha yüksek enerjiye ve düzenli bir periyoda(perdeye) sahip olduğu, darbesel harflerin ise zamanda sesli harfler gibi periyodik bulunamadıkları ancak kısa bir zaman aralığında yüksek güç ihtiva ettikleri görülür.



Şekil-3: LPC modeli.

İnsandaki vokal yapılar göze alınarak Şekil-3'de verilen model oluşturulabilir. Bu modelde sesli-sessiz harf seçimi ses tellerine, kazanç katı akciğere, çıkış katındaki filtre de ağız-burun yapısalına karşılık gelir. Modemin görevi, bu parametreleri doğru biçimde inceleyip GSM ses bandında iletime uygun veriler (sesler) sentezlemek ve alıcı tarafında hatasız olarak geri analiz etmektir. Modem, sentetik seslerin kanal tarafından bastırılma riskini azaltmak için yalnızca sesli sesler kullanarak sentez yapmaktadır [2,8].

Sentetik ses sentezi ve analizinde kullanılacak öğeler; perde (20ms gibi ufak bir aralıkta sesli seslerin periyodu), enerji ve vokal yapıya karşılık düşen (1) denklemindeki LPC katsayıları olarak özetlenebilir.

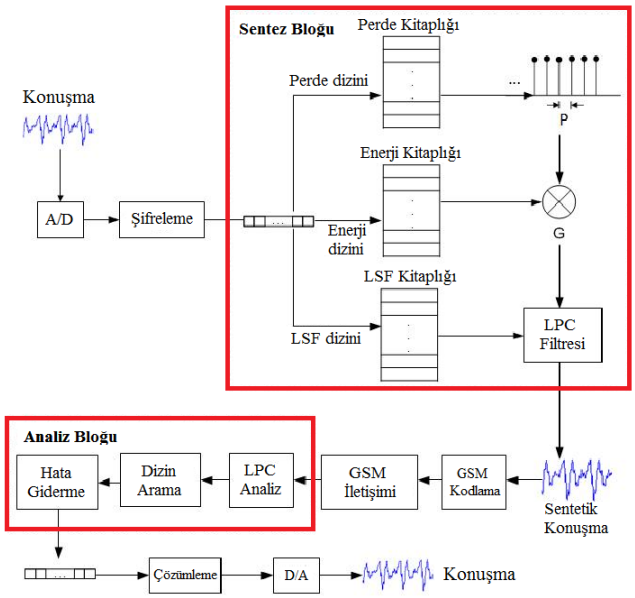
$$s(n) = Gu(n) + \sum_{k=1}^{p} a_k * s(n-k) \qquad (2)$$

Öngörü katsayılarını içeren (1) denklemi Şekil-3'de verilen sistemin tam matematik modelini karşılayacak biçimde (2) denklemine genişletilebilir. Burada "G" kazanca, "u" darbe dizisine ya da beyaz gürültü fonksiyonuna, "a_k" doğrusal öngörü katsayılarına, "s" de sentezlenen konuşma sinyaline karşılık gelmektedir.

B. Modemin İç Mimarisi

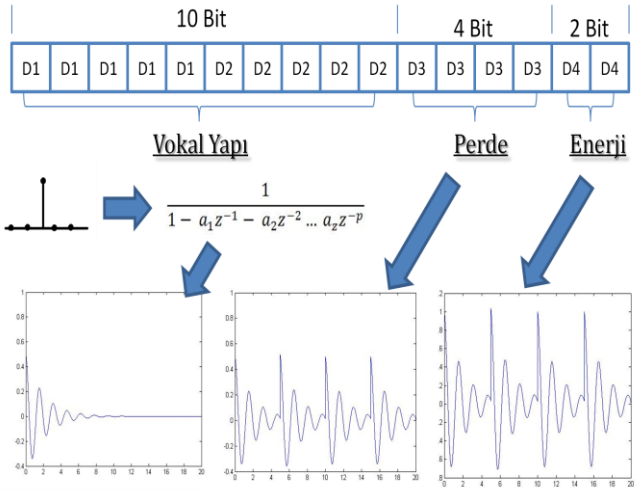
Tasarlanan modem 20ms'lik çerçeveler üzerinden analiz ve sentezleme yapmaktadır. Her bir çerçeve, yani sentetik ses paketi 16 bitlik bilgi içerir. Bu da 0.8Kbps'lik bir aktarım hızı anlamına gelir. Şifreli bir konuşma göz önüne alındığında Şekil-1'de gösterildiği gibi öncelikle konuşmalar şıkıştırılmalı ardından şifrelenip modemin girişine verilmelidir. Konuşmanın zamanda örneklenip sıkıştırılmadan doğrudan modeme verilmesi, gerçek zamanlı iletişime olanak tanımayacaktır.

Modemin iç mimarisi Şekil-4'de verilmiştir. Daha önce değinildiği gibi modem; vokal yapı, perde ve enerji bilgilerini kullanarak sentetik ses paketleri üretip çözümlemektedir. Vokal yapı için 10. dereceden doğrusal öngörü kullanılmıştır. 20ms'lik ses paketleri için 8KHz'de örnekleme frekansı kullanılmakta, bu sebeple her paket 160 adet sentetik ses örneği içermektedir.



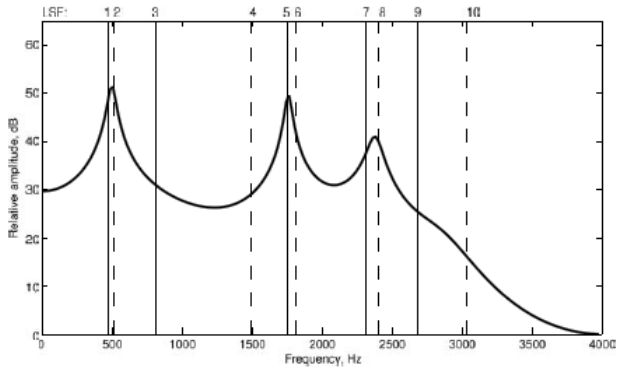
Şekil-4: Modemin iç mimarisi [2].

Modem verici olarak çalıştığı durumda Şekil-5'de gösterildiği gibi girişinden 16 bitlik bir dizi alır. Bu dizinin 10 biti üretilecek yapay sesin vokal yapısında, 4 biti perdede, 2 biti de enerjisinin sentezlenmesinde kullanılır. Özellikle ilk 14 bitlik vokal yapı ve perdeye ait kod kitaplığının oluşturulması modemin tasarımı ve başarımındaki kilit noktadır. GSM ses bandından iletime uygun kod kitaplarının oluşturulması ilgili kaynaklardan detaylarıyla incelenebilir [2,8].



Şekil-5: Giriş verisininden sentetik ses üretimi.

LPC parametreleri kod kitaplığında lineer spektrum frekanslarına (Linear Spektral Frequencies - LSF) dönüştürülerek saklanır. Bu dönüşüm ileriki bölümlerde gösterileceği gibi sisteme her ne kadar yüksek işlem yükü getirse de LPC parametrelerinin güvenli bir şekilde kuantalanabilmesini sağlar. LPC parametreleri tüm kutup bir filtrenin katsayılarıdır ve düşük anlamlı bitlerin kırpılması sonucu yapılan kuantalamalar köklerin yerini kaydırıp filtrenin darbe cevabını (ses sentezini) kararsızlaştırabilir.



Şekil-6: LPC spektrumunda LSF değerlerinin yerine bir örnek. Tek sayılı LSF değerleri düz, çift sayılılar çizgili dikey doğrularla ifade edilmiştir [5].

LSF değerlerleri kuantalama neticesinde oluşacak bozulmalara karşı LPC değerlerine göre daha dayanıklıdır [5,9]. LSF' ler LPC sentez filtresinin frekans cevabında rezonans frekansları üzerinde Şekil-6'da gösterildiği gibi yer alır. 10 bit karşılığında 2^{10} adet farklı vokal yapı oluşturmamak adına, 10 adet LSF katsayısı ilk 4 ve son 6 olmak üzere 5'er bitlik iki alt kod kitaplığına bölünmüş, böylece $2^5 + 2^5$ adet farklı vokal yapı oluşturulması yeterli olmuştur.

C. Modemin Verici Olarak Kullanılması

Modem 16 bitlik bir diziyi aşağıdaki adımları izleyerek transfer eder.

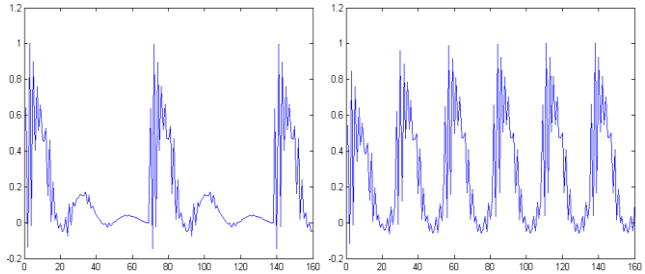
1) ***Vokal Modelin Oluşturulması:*** 10 bitlik girdi ile LSF kod kitaplığından bir vokal yapı seçilir.

2) ***LSF-LPC dönüşümü:*** Seçilen vokal yapıya ait LSF parametreleri LPC parametrelerine yani sentez filtresinin katsayılarına dönüştürülür.

3) ***Ses sentezi:*** Giriş verisine göre belirlenen LPC (vokal yapı) parametrelerinden, seçilen perde bilgisine göre ses sentezlenir.

4) ***Enerji Belirleme:*** Üretilen çıkış dizisi, seçilen enerji bilgisine karşı düşen katsayı ile çarpılarak yapay ses sentezi tamamlanır.

Şekil-7'de aynı vokal yapı, farklı darbe sıklığı (perde) için sentez filtresinin ters Z dönüşümü, yani üretilen ses paketi gösterilmiştir. Seçilen perde bilgisine göre, sentez filtresine giren darbelerin sıklığı değiştirilmiş olur. Modem mimarisinde her bir sayısal veri girişi için daha önce açıklandığı gibi çıkışta oluşturulan sentetik ses paketi 160 adet örnek içerir.



Şekil-7: Aynı vokal yapıda fakat farklı perdede iki paket.

D. Modemin Alıcı Olarak Kullanılması

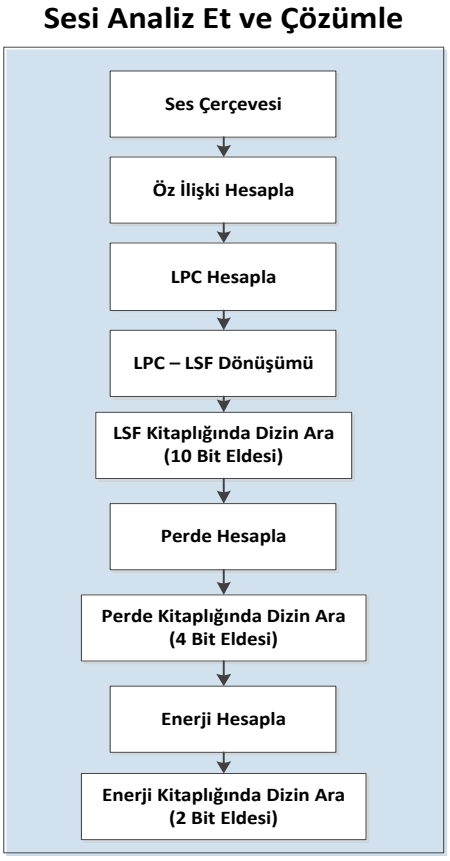
Modemin analiz aşaması sentezden daha karmaşıktır. Analiz adımları blok diyagram olarak şekil-8'de verilmiştir. Girişten 8KHz'de örneklenerek oluşturulan 20ms'lik bir ses çerçevesi çözümlenirken aşağıdaki adımlar takip edilir.

1) ***LPC Katsayılarının Eldesi:*** Ses çerçevesinin vokal yapısı analiz edilerek doğrusal öngörü katsayıları elde edilir.

2) ***LPC – LSF Dönüşümü:*** Elde edilen 10 adet LPC katsayısı LSF parametrelerine dönüştürülür. Dönüşüm sonucu elde edilen katsayıların kod kitaplığındaki yeri, yani dizini bulunur. Dizin numarası, çözümlenecek sayısal verinin ilk 10 bitine karşılık gelir.

3) ***Perdenin Hesaplanması:*** Perde hesabında doğrusal öngörü katsayılarından faydalanılarak spektral temelli analizler yapılabileceği gibi, modem uygulamaya özel belirli ses paketleri ürettiğinden genlik kontrolü temelinde analizler de yapılabilir. Perde hesabında yöntem çeşitliliği çok olmakla beraber, hız ve hata oranı arasında bir seçim yapılması gerekmektedir. Doğrusal öngörü katsayıları ile yapılan perde analizleri hız açısından maliyetli olması sebebiyle genlik temelinde bir yöntem kullanılmıştır. Perde analizi sonucu 4 bit daha çözümlenmiş olur.

4) ***Enerji Hesabı***: Enerji bilgisi için, orijinal çerçeve kod kitaplığındaki 4 farklı enerji katsayısı ile çarpılarak normalize edilmiş dizi ile karşılaştırılır ve hatanın en az olduğu değer tespit edilir. Böylece son 2 bit geri elde edilir.

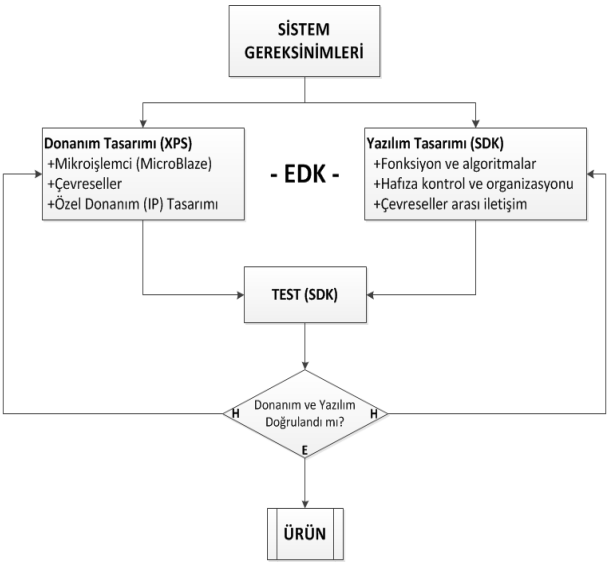


Şekil-8: Modemin alıcı olarak çalışması.

III. MODEMİN GERÇEKLENMESİ

Sistemin gerçeklemesinde yazılım ve donanım geliştirme ortamı olarak Xilinx EDK programından faydalanılmıştır. Bu programın alt programları ve çalışma esnasında faydalanıldıkları alanlar Şekil-9'daki şemada gösterilmiştir.

Bu bölümde modemin gerçeklenmesinde kullanılan fiziki ve sanal donanımlar (Intellectual Property – IP) ile yazılım blokları açıklanarak, karşılaşılan problemlere getirilen yazılımsal ve donanımsal çözümlere değinilmiştir.



Şekil-9: Uygulama geliştirme şeması.

A. *Donanımlar ve IP'ler*

Sistem Atlys geliştirme kartı üzerinde gerçeklenmiştir [10]. Kartın üzerinde fiziki donanım olarak Spartan6 sahada programlanabilir kapı dizisi (Field Programmable Gate Array – FPGA) ve çifte aktarım hızlı (Double Data Rate - DDR2) rastgele erişimli bellek (Random Access Memory - RAM) kullanılmıştır [11].

MicroBlaze çekirdeğiyle birlikte, bir adet zamanlayıcı, bir adet DDR kontrol, bir adet kesme kontrol ve iki adet de I/O IP çekirdeği kullanılmıştır. DDR kontrol IP'si karmaşık DDR haberleşme ve kontrol adımlarını halledерek kullanıcının yazılım ile hafızaya erişimini basite indirgemektedir. Zamanlayıcı, senkron veri alışverişini düzenlemede ve fonksiyonların işlem sürelerini hesaplamada kullanılmıştır. MicroBlaze çekirdeğinin bir adet kesme girişi bulunduğundan, birden çok kesmenin kontrol ve işletilmesi için kesme kontrol IP'si kullanılmıştır. Bu IP'ler Xilinx tarafından sağlanmaktadır.

Sistem performansını arttırmak amacıyla özel olarak tasarlanan korelasyon hesaplama IP'si donanıma eklenmiştir. Kesme özelliği içeren özel tasarım IP'lerin uBlaze çekirdeğine eklenip, yazılım-donanım köprüsünün kurulmasına ilişkin detaylar ilgili kaynaktan incelenebilir [11].

MicroBlaze ve diğer tüm çevreseller 100MHz'de, özel tasarlanan IP 50MHz'de çalıştırılmıştır. Fonksiyon sürelerini hesaplamak ve senkron giriş çıkışları yönetmek için zamanlayıcı kesme periyodu 500us olarak belirlenmiştir.

B. *Yazılımlar*

Modemin, ikinci bölümde açıklananmış olan sentez ve analiz parçaları C dilinde algoritmik olarak gerçeklenerek, yapılan ilk test sonucunda Tablo-1'de verilen işlem süreleri ile Tablo-2'de verilen bit başına düşen zaman maliyetleri elde edilmiştir. İki tabloda da, gerekliliği ve önemi bir önceki bölümde açıklanan LPC-LSF dönüşümünün diğer süreçlere göre oldukça orantısız ve gerçek zamanlı iletişime olanak tanımayacak derecede uzun sürdüğü görülmüştür. Sistem 20ms'lik çerçevelerle çalıştığından, gerçek zamanlı iletişim için her bir çerçevenin en geç 20ms içerisinde sentez veya analiz edilmesi gerektiği açıktır. LPC-LSF dönüşümü incelendiğinde kök bulma adımının en uzun süreye sahip olduğu görülmektedir. Bu sebeple öncelikle kök bulma sürecinin, ardından da ses çerçevesinden LPC katsayılarına geçişin hızlandırılması için çalışılmıştır.

Tablo-1: Fonksiyon süreleri. 0.5ms altındaki hassasiyetler ortalama değerler gözetilerek belirtilmiştir.

Fonksiyon	Süre (ms)
Modemin Alıcı Olarak Kullanımı (16 bit)	**950**
Çerçeve - LPC Katsayıları	**22,5**
Ses Paketi - Özilişki Parametreleri	21
Özilişki Parametreleri - LPC Katsayıları	1,5
LPC - LSF Dönüşümü (10 bit eldesi)	**924,5**
P ve Q Polinomlarının Oluşturulması	0,5
P Polinomunun Köklerinin Eldesi	465
Q Polinomunun Köklerinin Eldesi	455
Köklerden LSF katsayılarına Geçiş	2,5
Kod Kitaplığında İlk dört LPC taraması (5 bit eldesi)	0,75
Kod Kitaplığında Son altı LPC taraması (5 bit eldesi)	0,75
Perdenin Eldesi (4 bit eldesi)	**2**
Perdenin hesabı	1,75
Perde kod kitaplığında dizin taraması (4 bit eldesi)	0,25
Enerjinin Eldesi (2 bit eldesi)	**1**
Modemin Verici Olarak Kullanımı (16 bit)	**30**
LSF - LPC Dönüşümü (10 bit girdisi)	4
Ses Sentezi (4 bit girdisi)	25
Enerji Sentezi (2 bit girdisi)	1

Tablo-2: Süreçlerde bit başına zaman maliyeti.

Fonksiyon	Veri Eldesi (zaman(ms)/bit)
Vokal Yapı Analizi	954,5 / 10 = **95,45**
Perde Analizi	2 / 4 = **0,5**
Enerji Analizi	1 / 2 = **0.5**
Vokal Yapı Sentezi	4 / 10 = **0,4**
Ses Sentezi	25 / 4 = **6,25**
Enerji Sentezi	1 / 2 = **0,5**

C. *LPC-LSF Dönüşümü*

Doğrusal öngörü derecesinin 10 olarak kullanılması sebebiyle sistem girişten aldığı ses çerçevesinden 10 adet LPC katsayısı üretir. LPC-LSF dönüşümünde, katsayıları

LPC parametrelerinden oluşturulan aşağıdaki (3-4) iki polinomun köklerinin hesaplanması gerekir [9]. Öngörü derecesi "p" 10 olduğundan bu, iki adet 10. dereceden fonksiyon anlamına gelir.

$$P(z) = A_0 * z^p + A_1 * z^{(p-1)} + \ldots + A_p \quad (3)$$
$$Q(z) = B_0 * z^p + B_1 * z^{(p-1)} + \ldots + B_p \quad (4)$$

Bu denklemlerde A_0 ve B_0 1'e eşit olmakla beraber diğer katsayılar aşağıda verilen (5) ve (6) denklemleriyle ifade edilir [9].

$$A_k = (a_k - a_{p+1-k}) + A_{k-1} \quad k = 1 \ldots p \quad (5)$$
$$B_k = (a_k + a_{p+1-k}) - B_{k-1} \quad k = 1 \ldots p \quad (6)$$

Bu iki polinomun, iki önemli özelliği mevcuttur;

1) P ve Q polinomlarının tüm kökleri "Z" bölgesinde birim çember üzerindedir

2) P ve Q polinomlarının kökleri birim çember üzerinde sırasıyla birbiri ardına gelir ve sıfır ile pi arasındadır, $0 \leq w_{q,0} < w_{p,0} < w_{q,1} < w_{p,1} < \cdots \leq \pi$

Polinomların simetrik oluşundan faydalanılarak dereceleri *p/2* yani 5'e indirilebilir. Daha da ileri gidilerek çeşitli değişken atamalarıyla bu iki denklem aşağıdaki (7-8) formlara indirgenebilir [9].

$$P_{10}(x) = 16A_0x_5 + 8A_1x_4 + (4A_2 - 20A_0)x_3 + (2A_3 - 8A_1)x_2 + (5A_0 - 3A_2 + A_4)x + (A_1 - A_3 + 1/2\ A_5) \quad (7)$$

$$Q_{10}(x) = 16B_0x_5 + 8B_1x_4 + (4B_2 - 20B_0)x_3 + (2B_3 - 8B_1)x_2 + (5B_0 - 3B_2 + B_4)x + (B_1 - B_3 + 1/2\ B_5) \quad (8)$$

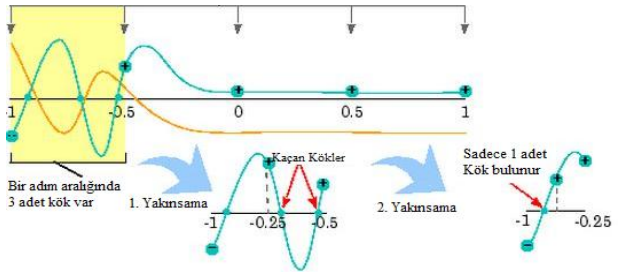
Her biri 5. Dereceden olan P ve Q polinomlarının kökleri reel ve -1 ile +1 arasındadır. Bulunan 10 adet kök (9) denklemine konularak LSF parametreleri elde edilir [9].

$$LSF(i) = \frac{cos^{-1}(x_i)}{2\pi T} \quad (9)$$

D. Kök Bulma

P ve Q polinomlarının köklerinin hesaplanmasında ikiye bölme metodundan faydalanılmıştır. Bu yöntemde köklerin -1 ile +1 arasında ve reel olduğu göz önünde bulundurularak, polinomun -1 değerinden başlanarak belirli adım aralıklarında değeri hesaplanır ve işaret değişiminin olduğu bölümde "x" ekseninden geçildiği yani o bölgede bir kökün bulundu belirlenmiş olur. Bu aşamadan sonra o bölge sınırları içerisinde köke yakınsanır. Bu algoritmanın performansını etkileyen iki parametre vardır:

1) ***Adım Aralığı:*** Adım aralığı ne kadar küçük olursa işlem yükü o denli artacaktır. Bu sebeple adım aralığı mümkün olduğunca büyük seçilmelidir. Buradaki sınırlama Şekil-10' da gösterildiği gibi bir adım aralığının içerisine iki kökün birden girmemesidir. Bu aralık uygulamaya göre optimize edilmelidir. Modemin gerçeklenmesinde adım sayısı ilk testlerde 100 seçilip ardından 40'a kadar indirilerek optimize edilmiş ve nihai adım aralığı $(1 - (-1))/40 \cong 0.05$ olarak belirlenmiştir.



Şekil-10: Bir adım aralığında birden çok kök bulunma durumu [13].

2) ***Hata Tanımı:*** Sıfır geçişinin tespit edildiği adım aralığı içerisinde köke yakınsama miktarı, yani hata miktarının belirlenmesi ikinci kıstastır. LSF kod kitaplığındaki değerler göz önünde bulundurularak hata ±0.0001 olarak belirlenmiş ancak daha sonra hatanın ±0.0015 olarak tanımlanmasının sistem içinde tolere edilebildiği belirlenmiştir..

Tablo-2: Kök bulma süresinin parametrelere göre değişimi.

Hata Tanımı	Adım Sayısı	Süre (ms)
0.0001	100	1325
0.001	100	1117
0.0015	100	1083
0.0015	60	761
0.0015	40	695

İki parametrenin değiştirilmesi kök bulma sürecine Tablo-2'deki gibi etki etmiştir. Adım sayısı 40'ın altına indirildiğinde bazı durumlarda kök atlandığı, yani bir aralığa birden çok kök girdiği görülmüştür.

Kök bulmada kullanılan algoritma Şekil-11'de verilmiştir. Adımın +1 değerine gelmesi veya aranan köklerin bulunması durumunda döngüden çıkılmaktadır. Normal bir akışta adım sayısı 40 seçildiğinde P veya Q polinomlarından birinin kökleri, 40 kez işaret değişim kontrolü, 5 kez de kök bulma fonksiyonu çağırılarak bulunacaktır. Hem işaret değişimini algılamak hem de köke yakınsamak için polinom değerlerinin sürekli hesaplanması gerekmektedir. Polinom hesaplamasına yapılan inceleme bir sonraki başlık altında paylaşılmıştır.

Şekil-11: Kök bulma algoritması.

Parametre optimizasyonun ardından ikinci bir adım olarak kök bulma sürecine Newton-Raphson yöntemi entegre edilmeye çalışılmıştır. Bu aşamada karşılaşılan problem Newton-Raphson (NR) algoritmasının 5 adet kökü bulması için başlangıç koşullarının nasıl belirleneceği

olmuştur. Bunun için NR ve ikiye bölme algoritmaları harmanlanarak, NR'nin başlangıç koşulları ikiye bölme algoritmasındaki "adımların" başlangıç noktası olarak seçilmiş böylece ikinci bir iyileştirme elde edilmiştir.

Kök bulma aşamasında bir üçüncü iyileştirme işaret değişiminin hızlı algılanabilmesiyle sağlanmıştır. İşaret değişim fonksiyonu modüler biçimde kendisine verilen iki nokta arasında fonksiyonun işaret değiştirip değiştirmediğini analiz etmektedir. Eğer bir işaret değişimi varsa bu aralıkta kök aranacaktır. Süreç başladığında öncelikle -1 ile -0.95 daha sonra -0.95 ve -0.90 aralığında işaret değişimi aranacaktır. Arama "x" ekseninde monoton artan değerlerle yapılacağı için aslında her yeni aralık için hesaplanacak iki değerden biri, bir önceki aralıkta hesaplanmış olacaktır. Bu sebeple fonksiyonunun modülerliği bozularak her adım aralığında bir sonraki aralık için gerekli değerin hafızada tutulması sağlanmış böylece 80 yerine 40 adet nokta işaret değişimi algılama için yeterli kılınmıştır.

Kök bulma sürecinde yapılan iyileştirmelerin yerel ve toplam analiz süresine olan etkileri dördüncü bölümde topluca verilmiştir.

E. Hızlı Polinom Hesabı

Kök bulma aşamasındaki iyileştirmelerin ardından sistem daha detaylı incelendiğinde esas zaman alan sürecin polinom değerlerinin hesaplaması olduğu görülmüştür. Polinom hesabı sistemin ilk testlerinde modülerliği sağlamak adına (10)'da gösterildiği gibi "mat.h" kütüphanesinden faydalanılarak gerçekleştirilmiştir. Burada C[n] polinomun katsayılarıdır. Hesabı hızlandırmak için alternatif yöntemler incelenmiş ve polinom "pow()" fonksiyonları yerine Horner yöntemiyle (11)'da gösterildiği biçimde hesaplanmıştır [14]. Horner yöntemi donanımda paralelleştirmeye müsait olmasa da, yazılımla çarp-topla komutu içeren işlemcilerde kolaylıkla hesaplanabilir. Hesaplanması arzu edilen fonksiyon iteratif olarak giriş değişkeninin (örn. x) parantezine alınarak ilerlenir ve bu şekilde ifade edilir [14].

$$\mathrm{y} = \mathrm{C[n]} * \mathrm{pow(x,n)} + \mathrm{C[n-1]} * \mathrm{pow(x,n-1)} + \cdots + \mathrm{C[0]} \tag{10}$$

$$\mathrm{y} = \big(\ldots\big((\mathrm{C[n]} * \mathrm{x} + \mathrm{C[n-1]}) * \mathrm{x} + \mathrm{C[n-2]}\big) * \mathrm{x} + \cdots + \mathrm{C[1]}\big) * \mathrm{x} + \mathrm{C[0]} \tag{11}$$

F. Korelasyon (Öz-İlişki) Donanımı

Modeme ulaşan bir ses paketi içerisinden (x) denklemindeki LPC katsayılarının eldesi için iki aşamalı bir yol izlenmiştir. İlk olarak (12) ile verilen denklemle öz-ilişki parametreleri hesaplanır. Bu işlemin ardından öz-ilişki katsayılarından faydalanılarak (13)'de verilen denklem takımı oluşturulabilir [8]. Burada "N" çerçeve boyutunu, "n" incelenen çerçeveyi, $r_n(i)$ öz ilişki değerlerini, $s_n(m)$ ses paketindeki örnekleri, a_k ise LPC katsayılarını ifade etmektedir.

$$r_n(i) = \sum_{m=1}^{N} s_n(m) * s_n(m+i) \tag{12}$$

$$\sum_{k=1}^{p} a_k * r_n(i-k) = r_n(i) \tag{13}$$

Yukarıda (13) ile verilen eşitlik (14)'de matris formunda gösterilmiştir. Bu matrisin çözülmesiyle LPC katsayıları elde edilir. Matris $A_{i,j} = A_{i+1,\ j+1}$ özelliği ile Toeplitz yapısındadır. Teorik olarak matrisin tersi alınıp kendisiyle çarpılarak çözüm elde edilebilse de, sayısal sistemlerde sınırlı bit ve ondalık sebebiyle hatalar oluşabilmektedir [8].

Öte yandan Teoplitz yapısından faydalanılarak doğru ve hızlı biçimde sonuca ulaşılabilir. Bu tip çözümlerde, yaygın olarak tercih edilen Levinson – Durbin algoritması tercih edilmektedir[8,9]. Çözüm yaklaşık "p^2" adet çarp-topla işlemi içermektedir ve Levinson-Durbin algoritması sayesinde onuncu dereceden (p=10) bir sistemin çözümünde 25 kat daha az aritmetik işleme ihtiyaç duyulmaktadır [15].

$$\begin{bmatrix} R_n(0) & R_n(1) & . & R_n(p-1) \\ R_n(1) & . & . & R_n(p-2) \\ \vdots & \vdots & \vdots & \vdots \\ R_n(p-1) & . & . & R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ \vdots \\ R_n(p) \end{bmatrix} \tag{14}$$

Levinson - Durbin algoritması hem iteratif işlemler içerdiğinden hem de ses paketi – LPC sürecinde oransal olarak öz-ilişki katsayılarının hesaplamasına göre daha az zaman harcadığından yazılımsal olarak gerçeklemesi yapılmıştır.

Öte yandan öz-ilişki katsayıları aşağıda Şekil-12 ile verilen algoritma yardımıyla hesaplanmaktadır. Burada (12) denklemi yazılımla gerçeklenmiş ve her bir öz-ilişki katsayısı için 160 çevrimlik bir "for" döngüsü kullanılmıştır. Kök bulmanın ardından en uzun süreye sahip olan bu adımı hızlandırmak için korelasyon katsayılarının donanım yardımıyla paralel olarak hesaplanmasına karar verilmiştir.

```
for ( p=0; p<ongoru+1; p++) {
  for ( i=0; i<(paket_uzunlugu - p ); i++) {
   Korelasyon[p]=paket[i]*paket[i+p]+Korelasyon[p];
  }
}
```

Şekil-12: Korelasyon katsayılarının hesaplanması.

11 adet katsayı hesabının aynı anda paralelleştirilmesi MicroBlaze işlemcisi ile diğer çevresellerle birlikte Spartan6 FPGA sı içine sığmayacak kadar büyük bir tasarım olmuştur. Bu sebeple 6 adet hesaplama paralelleştirilerek, yazılım yardımıyla donanıma önce ilk 6 sonra son 5 katsayı paralel olarak hesaplattırılmıştır. Yapılan bu geliştirme sayesinde korelasyon hesaplama süreci 21ms'den 1,5ms'ye indirilmiştir.

G. Sentez Üzerine

Modemin verici olarak çalıştığı durumdaki performansı gerçek zamanlı iletişim için yeterli değildir. Sürenin uzun olması 160 adet çıkış örneğinin topluca hesaplanmasından kaynaklanmaktadır. Öncelikle modüler tasarlanan sentez

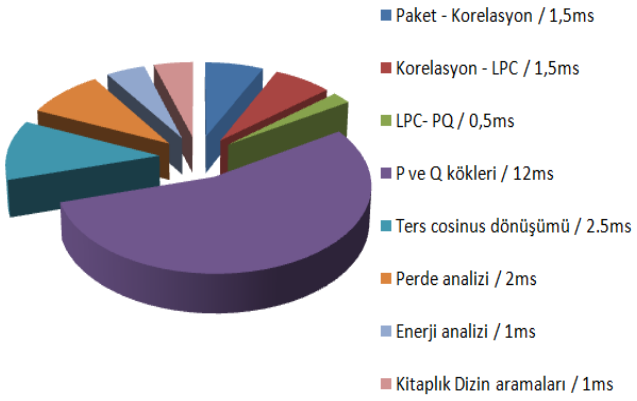
(ters Z) filtresi, sistemin tüm kutup olması yani payda yalnızca 1 bulunmasından faydalanılarak yazılım gerçeklemesinde pay katsayıları ile yapılan çarpma ve toplama işlemleri kaldırılmış ve sentez süresi 30ms'den 22 ms'ye indirilmiştir. Bununla beraber tüm ses paketi aynı anda hesaplanmak yerine, 8KHz'de bir o anki çıkış değeri için bir adet örnek hesaplanıp, böylece yazılımsal bir çözüm geliştirilebilir.Öte yandan filtreler donanımda gerçeklemeye elverişlidir. Dilenirse sentez filtresi için donanımsal bir çözüm tercih edilerek bir adet sentez IP'si yardımıyla modemin verici çalışma durumu için hız problemi ortadan kaldırılabilir.

IV. SONUÇLAR

Bu çalışmada GSM ses kanalının sayısal veri iletimine uygun hale getirilmesi, böylece kanal üzerinden şifreli bir görüşme veya resim transferi gibi uygulamaları mümkün kılmak için bir yöntem belirlenmiş ve gerçeklemeye çalışılmıştır. Sistemin tüm parçaları öncelikle algoritmik olarak hayata geçirilmiş ardından gerçek zamanlı iletişimin sağlanabilmesi için yazılımsal ve donanımsal çözümler geliştirilmiştir. Yapılan tüm iyileştirmelerin Tablo-1'de verilen ilk test sonuçlarına olan etkisi Tablo-3 de, son durumda alt blokların tüm sistem içerisindeki yüzde olarak işlem payları Şekil-13'de topluca verilmiştir.

Tablo-3: Geliştirmelerin yerel ve genel süreçlere etkisi. Süreler mili saniye üzerinden verilmiştir.

Geliştirme	**Ana/Alt Süreç**	**Eski Süreç/ Fonk.**	**Yeni Süreç/ Fonk.**	**Tüm Analiz Süresi**
Yok	-	-	-	**950**
Adım ve Aralık Parametre Optimizasyonu	LPC-LSF / Kök Bulma	925 920	675 670	695
Hızlı Polinom Hesabı (Horner Metodu)	LPC-LSF / Kök Bulma	675 670	35 30	60
Newton-Raphson Algoritması	LPC-LSF / Kök Bulma	35 30	25 20	50
Hızlı İşaret Değişim Algılama	LPC-LSF / Kök Bulma	25 20	17 12	42
Donanımsal Korelasyon	Paket-LPC / Öz-İlişki	22,5 21	3 1,5	**22**



Şekil-13: Geliştirmeler sonucunda elde edilen analiz süreleri ve yüzde olarak payları.

Tablo-3'de verilen iyileştirmelerle birlikte LPC-LSF dönüşümü 17 mili saniyeye kadar indirilmiştir. Dönüşüm bu haliyle bile Şekil-13'de gösterildiği gibi tüm sürecin yüzde olarak yarısından fazlasını kaplamaktadır. LPC-LSF ve dizin arama blokları tamamen yazılımsal olarak çalıştığından, çekirdeğin 100MHz yerine 200MHz'de çalıştırılması yazılımsal süreçleri iki kata kadar hızlandıracaktır. Günümüzde 1GHz ve üzerinde çalışan işlemcilerin mevcut olduğu göz önünde bulundurularak ayrık Mikroişlemci ve FPGA ikilisi ile gerçek zamanlı çalışma hedefine ulaşılabileceği açıktır.

Ancak çalışmaların yapıldığı Spartan-6 üzerine kurulan MicroBlaze çekirdeği erişilebilen maksimum frekansta (100MHz) çalıştırılmıştır. Çalışmanın kullanım alanını da düşünülerek yeni geliştirmelerdeki esas hedefin, sistemin mümkün olduğunca düşük frekanslarda FPGA üzerinde yazılım-donanım ortak gerçeklenebilmesi ve düşük enerji sarfiyatlarının elde edilebilmesi olacağı söylenebilir.

İşlemci ve IP'lerin frekanslarını değiştirmeden gerçek zamanlı çalışma için yapılabilecek yeni geliştirmeler ve gerekli görülen eklentiler aşağıda paylaşılmıştır.

A. Gerekli Geliştirmeler

1) Korelasyon katsayılarının tamamı paralel olarak hesaplanıp, böylece yazılım adımları aradan çıkartılıp 1ms mertebesinde bir hızlanma sağlanabilir. Bununla birlikte korelasyon katsayıları yeterince hızlı hesaplanırsa perde analizinde de bu bilgiden faydalanılabilir.

2) LPC - LSF dönüşümünde elde edilen 10 adet kökün ters kosinüsünün alınması 2,5ms sürmektedir. Fonksiyonellik düşünüldüğünde ödenen zaman bedeli çok fazladır. Bu durumu aşmak için kod kitaplığı LSF değerleri ile değil doğrudan kök değerleri ile oluşturulursa analiz 2,5ms daha hızlandırılabilir. Bu durumda modemin verici çalışma durumuna kök-LSF dönüşümü gerekli olacaktır ancak verici tarafında gelen bu ek süre gerçek zamanlı çalışmaya engel teşkil etmeyecektir.

3) Kök hesabının daha da hızlandırılması için polinom hesaplama yöntemi de donanıma çevirilebilir. Ancak IP ler tam sayılarla çalıştığından, yazılım ile donanım arasındaki veri geçişlerinde oldukça çok "float" ve "integer" dönüşümü yapılması gerekecek bu da zaman kaybına neden olacaktır. Kök bulma sürecindeki iyileştirmeler daha çok yöntem üzerine olmalıdır. Newton-Raphson için daha iyi başlangıç koşulları aranabilir. Aynı şekilde köklerin hangi adım aralıklarında daha çok çıktığı istatiksel olarak belirlenip öncelikle bu bölgelerde kök aranabilir.

4) Yazılım tasarımında "for" döngüleri birçok yerde kullanılmıştır. Bu döngüler açık biçimde doğrudan yazılarak (code inlining) istenilen ölçüde kod boyutundan feragat edilip hız artışı sağlanabilir.

5) Sayısal işaret işleme bloklarında kullanılan "float" değişkenler yerine sabit noktalı (fixed point) bir yaklaşımla

tüm hesaplamalar tam sayılar üzerinden yapılabilir. Böylece Mikroişlemci-IP arasındaki sayısal veri alış-verişinde "float" ve "integer" dönüşümlerine gerek kalmayacak, polinom değeri hesaplama gibi çokça kullanılan fonksiyonların donanıma geçirilmesinden verim alınacaktır.

B. Gerekli Eklentiler

1) İki mobil cihaz için analog ses seviyeleri farklı olabilir. Enerji bilgisinin doğru analiz edilebilmesi için iki cihaz kendi aralarında 4 farklı enerji girdisi ile haberleşip birbirlerini kalibre etmelidir.

2) Bu çalışmada bireysel ses paketlerinin sentez ve analizi üzerine yoğunlaşılmıştır. Gerçek iletişimde ses paketleri birbiri ardına iletilecek ve analog dünyada bir bütün olacaklardır. Bu sebeple sentezleyici ve analiz taraflarında pencereleme (windowing) ve örtüştürme (overlap) teknikleri uygulanmalıdır.

3) Sisteme hata giderme bloğu eklenmelidir. Her sistemde olduğu gibi, özellikle analog dünyanın da işin içine girdiği bu özel iletişim kanalında hatalar meydana gelecektir. Hataların düzeltilebilmesi veya en azından tespit edilebilmesi için sisteme bu işlemi yürütecek ek bir modül tasarlanmalıdır.

V. KAYNAKÇA

[1] LaDue, C.K.; Sapozhnykov, V.V.; Fienberg, K.S., "A Data Modem for GSM Voice Channel", *Vehicular Technology, IEEE Transactions on* , vol.57, no.4, pp.2205-2218, July 2008.

[2] M. A. Ozkan, B. Ors and G. Saldamli. "Secure voice communication via GSM network", In *Proceedings of the 7th International Conference on electrical and electronics Engineering (ELECO)*, Bursa, Turkey, December 2011.

[3] Yucun Yang; Suili Feng; Wu Ye; Xinsheng Ji, "A Transmission Scheme for Encrypted Speech over GSM Network," *Computer Science and Computational Technology, 2008. ISCSCT '08. International Symposium on* , vol.2, no., pp.805-808, 20-22 Dec. 2008.

[4] N.N. Katugampala, S. Villette, A. Kondoz, "Secure voice over GSM and other low bit rate systems," Secure GSM and Beyond: End to End Security for Mobile Communications, IEE Seminar on (Digest No. 2003/10059), Lodon,11 Feb. 2003.

[5] Ian McLoughlin, "Applied Speech and Audio Processing", Cambridge University Press,New York, 2009.

[6] N. S. Jayant and P. Noll, "*Digital Coding of Waveforms: Principles and Applications to Speech and Video"*, Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

[7] Alan V. Oppenheim and Ronald W. Schafer, "Digital Signal Processing", Prentice Hall, 1975.

[8] Özkan, M. A., 2011. GSM Ağı Üzerinden Güvenli Ses İletimi, *Lisans Tezi*, İ.T.Ü. Elektrik Elektronik Fakültesi, İstanbul.

[9] A.M.Kondoz, "Digital Speech:Coding for Low Bit Rate Communication Systems", John Wiley&Sons, Second edition 2004.

[10] Atlys, Atlys Board Reference Manual, 2011, Digilent.

[11] Tunçay S., 2012. Bir GSM Modemin FPGA Üzerinde Gerçeklenmesi, *Lisans Tezi*, İ.T.Ü. Elektrik Elektronik Fakültesi, İstanbul.

[12] Spartan-6, Spartan-6 Family Overview, 2011, Xilix.

[13] Signal Processing Blockset – LPC to LSF/LPS Conversion, Matlab Tutorial, 2007.

[14] Gavin S. Reynolds, "Investigation of different methods of fast polynomial evaluation", MSc in High Performance Computing, The University of Edinburgh, 2010.

[15] Gomez, P., 2004. Speech Coding & Linear Predictive Coding (LPC), *PhD Thesis*, Florida International University, Miami.

DSP ve FPGA'ler Arası Haberleşmede PCI express Kullanımı

Mumin Gözütok
Argenç Ltd.
ODTU Teknokent, Ankara
e-posta: mumin@argenc.com.tr

Göksel Günlü
Turgut Özal Üniversitesi
Elektrik-Elektronik Mühendisliği Bölümü
Keçiören, Ankara
eposta:ggunlu@turgutozal.edu.tr

Ramazan Günlü
Argenç Ltd.
ODTU Teknokent, Ankara
eposta: ramazan@argenc.com.tr

Özetçe—**Bu çalışma, gömülü sistemlerde DSP ve FPGA 'lerin birbirleriyle yüksek hızlarda haberleşmesinde kullanılan PCI express(PCIe) veri iletim altyapısını, halihazırdaki mevcut çözümleri, performanslarını ve muhtemel kullanım alanlarını ortaya koymak için hazırlanmıştır.**

I. GIRIŞ

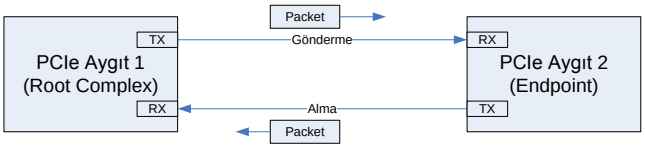
Yüksek hız ve performans gerektiren gömülü sistem uygulamalarında DSP ve FPGA'lerin farklı özelliklerinden faydalanabilmek için birlikte kullanımı çok önemlidir. DSP'ler, matematiksel işlem yetenekleri, sahip oldukları donanımsal hızlandırıcılar ve optimize yazılım kütüphaneleri gibi özellikleriyle hızlı ve güvenli sistem geliştirmede tasarımcıya büyük olanaklar sunmaktadırlar. Ancak çok sayıda girişi aynı anda yönetmekte ya da çok sayıda farklı formatlarda çıkış üretmekte, yani dış dünya ile bağlantı noktalarında, gerçek zamanlı yazılımsal yönetim problemleri, genel amaçlı giriş/çıkışların hız/performans yetersizliği, Direct Memory Access (DMA) ile ilgili kısıtlamalar gibi nedenlerle tasarımcıyı sınırlandırmaktadırlar. Bu durumda yüksek hız ve performans için önerilen yol, dış dünya ile bağlantı noktalarında FPGA'leri kullanıp, matematiksel işlem ve algoritmaların gerçeklenmesi için DSP'lerin yeteneklerinden faydalanmaktır. Bu ortak mimaride ortaya çıkan yaygın problem DSP ve FPGA arasındaki haberleşmenin nasıl olacağıdır. Yeni nesil gömülü sistemlerde haberleşme mimarileri güvenilirlik, hız ve fiziksel yapı gibi nedenlerle parallel veri yolu mimarisinden, seri veri yoluna doğru kaymaktadır. PCIe (Peripheral Component InterConnect Express), SRIO (Serial Rapid IO), Ethernet gibi yaygın kullanılan ve çeşitli ihtiyaçlara göre tasarlanmış seri veri yolu mimarileri mevcuttur. Bu çalışmada PCIe veri yolunun, yüksek hızlı gömülü sistemlerde DSP ve FPGA arasındaki iletişim kanalı olarak kullanılması anlatılacaktır.

II. PCI EXPRESS

PCIe genel olarak bilgisayar mimarisinde kullanımı ile bilinen ancak gömülü sistemlerde de çok sıklıkla kullanılan güçlü bir seri iletişim protokolüdür. Intel, Texas Instruments, Xilinx, PLX gibi yaklaşık 800 firmanın katılımıyla oluşturulan PCI-SIG grubu tarafından geliştirilmektedir. Türkiye'den de Vestel ve Aselsan bu grubun üyesidir[10]. PCI-SIG 1992 yılında kurulmuş ve uzun zaman o yılların veri iletişim formatları olan PCI ve PCI-X ile ilgilenmiştir. İlk PCIe standardını 2003 yılında PCIe 1.0a (Gen1) adıyla yayınlamıştır. İlk standartda veri iletim hızı tek veri yolu (lane) için 2.5 GT/s (giga transfer per second) olarak belirlenmiştir[1]. İlerleyen yıllarda standart sürekli iyileştirilmiştir. Günümüzde kullanılan en yeni PCIe standardı 3.0 'dır ve 8 GT/s hız desteklenmektedir[9].

A. PCIe Mimarisi

PCIe en temelde iki aygıtı noktadan noktaya birbirine bağlar. Bu temel bağlantı Şekil-1 'de verilmektedir. Link iki tek yönlü (dual-simplex) çalışır. Elektriksel olarak 2 LVDS (low voltage differential signaling) çiftinden oluşur. Bunlar gönderme çifti ve alma çiftidir[2]. Her link en az bir veri yolu desteklemelidir. Her veri yolu bir diferansiyel veri iletim çiftini ifade eder. Bant genişliğini artırmak için aygıtlar birden fazla veri yolu destekleyebilirler.Veri yolları N veri yolu sayısını ifade etmek üzere xN ile ifade edilir. PCIe 1.0a'da x1, x2, x4, x8, x16 ve x32 link genişlikleri tanımlanmıştır[1]. Veri yolu sayısı arttıkça veri iletim hızı da artmaktadır. Örneğin x8 genişliğinde bir link 20 Gigabit/sec veri transfer hızını destekler.

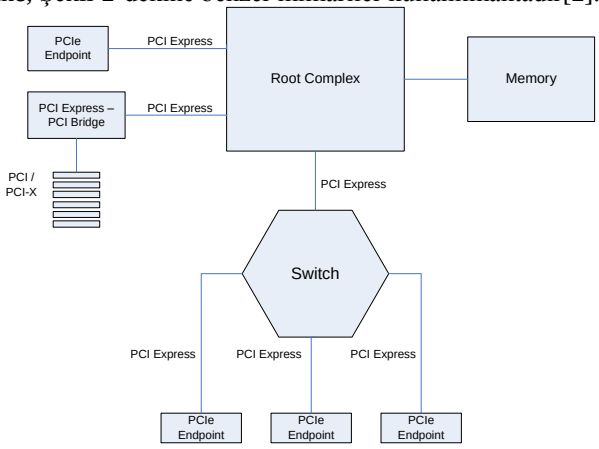


Şekil 1 : PCIe Temel Mimarisi

1) *Kök Kompleks (Root Complex) ve Sonlanım (Endpoint) Kavramları:* PCIe protokolü çok kapsamlı ve yaygın kullanılan bir protokoldür. Günümüze kadar 1.0a, 1.1, 2.0 (Gen2), 2.1 (Gen2) ve 3.0(Gen3) olmak üzere çeşitli versiyonları geliştirilmiştir. Şekil 1'deki yapıda verilen aygıtlar farklı versiyonları destekliyor olabilir. Bu noktada iyi haber geliştirilen tüm bu versiyonlar, önceki versiyonlar ile uyumludur[8]. Kötü haber ise haberleşmeye başlanılmadan önce iki aygıtın birbirinin kapasitesini bilmesi gerekliliğidir. Aksi takdirde daha yeni versiyona sahip cihaz karşıdaki cihazın kapasitesinden daha hızlı veri aktarımı yapabilir ya da anlamadığı bazı paketler gönderebilir. Yani haberleşmeye başlamadan önce kapasite öğrenme ya da standardın deyimi ile linkin eğitilmesi (link training) işleminin yapılması gerekir[2]. Buradaki problem bu iki aygıttan hangisinin bu işlemi başlatacağıdır. Bu problemi aşmak için fabrika çıkışlı olarak bazı aygıtlar kök kompleks, bazı aygıtlarsa sonlanım olarak adlandırılır. Fabrika çıkışlı her ikisini birden destekleyen aygıtlar da

mevcuttur. Linkin eğitilmesi ve devamı ile görevli aygıt kök kompleksdir. Kök kompleks en eski standarda gore eğitme işlemini başlatarak hattaki aygıtın kapasitesini öğrenir[2]. Ardından iki aygıt için de uygun olan bir standartta haberleşmeyi başlatır.

2) *Daha Karmaşık Mimariler :* Bu yetenekte bir protokolün yanlızca iki aygıt arasındaki bağlantıyı tanımlaması ve desteklemesi tabiki beklenen bir durum değildir. Daha büyük sistemlerde Şekil-1'deki mimari yerine, Şekil-2'dekine benzer mimariler kullanılmaktadır[2].

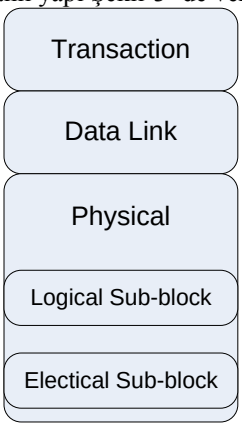


Şekil 2 : Kapsamlı PCIe Mimarisi

Şekil-2'de verilen mimaride kök kompleksin birden fazla PCI express portu mevcuttur. Burada dikkat edilirse genel kural bozulmamış ve kök kompleksdeki herbir port tek bir aygıta bağlanmıştır. Bu aygıt bir sonlanım, köprü ya da anahtar olabilir. Bu çalışma kapsamında DSP ve FPGA arasındaki haberleşmeye odaklanıldığı için Şekil-1'deki mimari temel çıkış noktası olacak, Şekil-2'deki mimari ve aygıtların detaylarına girilmeyecektir.

B. PCIe Katmanları

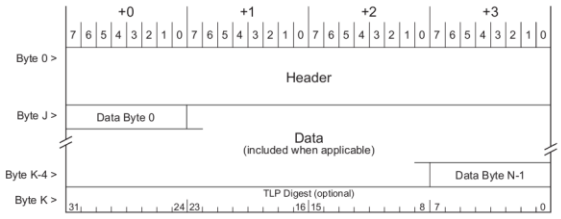
PCIe, günümüz protokollerinde sıklıkla rastlanıldığı gibi katmanlı şekilde tasarlanmıştır. Bu katmanlar yukarıdan aşağıya doğru İşlem Katmanı (Transaction Layer), Veri Link Katmanı (Data Link Layer) ve Fiziksel Katman (Physical Layer)'dır. Bu katmanlı yapı Şekil-3 'de verilmektedir[2].



Şekil 3 : PCIe Katmanları

Bu yapı standart geliştirilerine katmanların tek tek ele alarak iyileştirilmesine veya değiştirilmesine olanak sağlarken tasarımcılara da iş paylaşımı imkanı sağlamaktadır. Örneğin donanım tasarımcıları sadece fiziksel katman ile ilgilenirken, yazılım tasarımcıları fiziksel detayları hiç düşünmeden işlem katmanı ile ilgilenebilmektedir. Bunun yanında, gömülü sistem tasarımcıları, çok özel durumlar dışında veri link katmanı ile hiç ilgilenmemektedirler. Buradaki işlemler otomatik olarak aygıtlar tarafından yapılmaktadır.

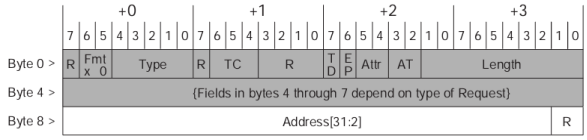
1) *İşlem Katmanı:* PCIe yazılımcılarının gördüğü ve veri iletimi için kulladığı katman burasıdır. PCIe protokolünde veriler paket temelli olarak iletilmektedir[2]. Yazılımcı bu paketin içeriğini oluşturur ve kullandığı aygıta paketi göndermesi komutunu iletir. Paketin güvenli bir şekilde karşı tarafa gönderilmesi ve kullanıcıya bu durumun raporlanması aygıtın sorumluluğundadır. Bu paketin (TLP, Transaction Layer Packet) en genel yapısı aşağıdaki şekilde verilmektedir.



Şekil 4 : TLP Genel Görünümü[2]

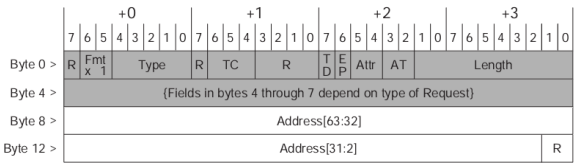
Paket üç ana bölümden oluşmaktadır. Bu bölümler Başlık(Header), Veri ve TLP Özet(Digest) 'tir.

Başlık kısmı paketle ilgili özel bilgileri içermektedir. Başlık aygıtlar tarafından desteklenen adresleme şekline göre 32-bit ya da 64-bit olabilmektedir. Başlığın detayları Şekil-5 'de verilmektedir. Burada başlık boyutu 3 DW (double word) 'dir.



Şekil 5 : 32-bit Başlık[2]

64-bit başlık yapısı ise Şekil-6 'da verilmiştir. Burada başlık boyutu 4 DW 'dir.



Şekil 6 : 64-bit Başlık[2]

Başlıkda 1. DW paket uzunluğu, tipi gibi pakete ilişkin önemli bilgileri taşımaktadır. 2. DW byte maskelemeyle ilgili ayarları barındırır. 3. ve 4. DW 'ler ise paketin hangi adrese gönderildiğini taşır. Şekil-1 'deki PCIe temel bağlantı şeklini gözönünde bulundurursak aygıtlar uç uca bağlı olduğu için paketin hangi aygıta gönderildiği bilinmektedir. Bilinmesi gereken paketin bu aygıtın hangi adresine yazılacağıdır. Bu noktada PCIe 'de sıklıkla rastlanan BAR (base address register) kavramına değinmek gerekmektedir. BAR çoğunlukla PCIe aygıtlarının içerisindeki yazmaçları (register) tanımlar. Örneğin bu yazmaçlar BAR0, BAR1,

BAR2 olabilir. Bu yazmaçlara aygıtın hangi adresleri kabul edeceği kullanıcı tarafından programlanır. Aygıt bu adresleri içeren bir TLP aldığında kabul eder ve aygıtın türüne gore bazen otomatik olarak ilgili adreslere gelen veriyi yerleştirir, bazen de yazılımcıyı yeni bir adres aldığına ilişkin uyarır. Eğer TLP adresi BAR adresleri ile uyuşmuyorsa TLP gözardı edilir ve herhangi bir işlem yapılmaz. Bu teknik sayesinde PCIe aygıtları sanki harici belleklere ulaşıyormuş gibi birbirlerinin belleklerindeki verilere ulaşabilmektedir. Örneğin bir FPGA dış dünyadan örneklediği bir sinyali bu altyapıyı kullarak, DSP 'nin dahili ya da harici belleklerine yerleştirebilir. Bu işlem çoğunlukla DSP'den bağımsız yapıldığından DSP'nin o an yaptığı işlemi kesmesine gerek yoktur. DSP istediği an FPGA tarafından hazırlanmış ve gönderilmiş bu verileri, kendi belleklerinden çekerek işleyebilir.

TLP içerisindeki veri kısmına veri yükü (data payload) da denilmektedir. Şekil-5 ve Şekil-6 'da verilen başlık yapısı dikkatli incelenirse paketteki veri sayısını ifade eden "Length" (uzunluk) bölümü 10-bittir. Dolayısıyla PCIe paketlerinin maksimum veri yükü 1024 Byte'tır. Ancak bu veri yükü boyutunu PCIe aygıtlarının tamamı desteklemez. Örneğin bazı aygıtlar için en fazla veri yükü boyutu 128 byte, 256 byte ya da 512 byte olabilir.

TLP içerisindeki en son kısım olan TLP özet kısmında iletilen veri yüküne ilişkin CRC değeri tutulmaktadır. Bu kısım opsiyoneldir. Çünkü verinin güvenli iletilmesi için daha aşağıdaki katmanlarda bu CRC değeri hesaplanmakta ve diğer aygıta iletilmektedir. Diğer aygıtta yine bu değer otomatik olarak kontrol edilmektedir.Bu kısım veri link katmanı içerisinde otomatik olarak yapıldığı için TLP içerisindeki bu kontrol kısmı opsiyonel bırakılmıştır.

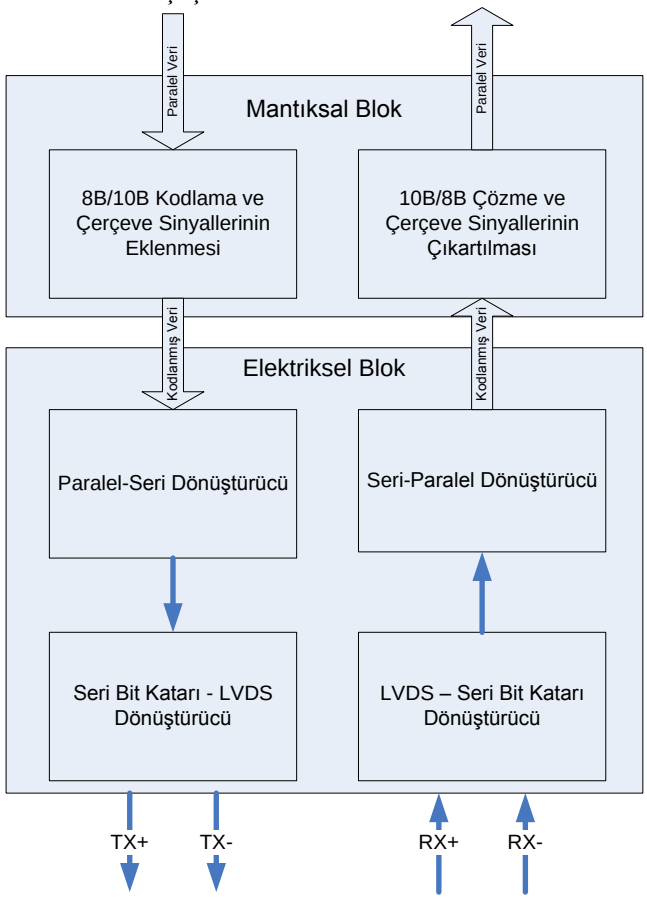
2) *Veri Link Katmanı :* Veri link katmanı, fiziksel katman ile işlem katmanı arasındaki geçiş katmanıdır. Ana görevi TLP iletimi esnasında güvenli bir mekanizma sağlamak yani TLP'nin karşı aygıta düzgün iletildiğini garanti altına almaktır[8]. Bu işlem için çok çeşitli hata kontrol mekanizmaları kullanır. Bu katmandaki işlemler genellikle otomatik olarak aygıt tarafından yapıldığı için tasarımcıya bir yük getirmemektedir. Onun için bu çalışma kapsamında bu katmana ilişkin detaya girilmeyecektir.

3) *Fiziksel Katman :* Fiziksel Katmanın görevi veri link katmanından gelen TLP'leri kodlayıp, çerçeve sinyallerini(framing symbols) ekleyerek hedefe iletmektir. Bu katman iki bloğa bölünmektedir. Bunlardan ilki mantıksal blok (logical sub-block), diğeri ise elektriksel blok (electrical sub-block) 'tur.

Mantıksal Blok, gönderilecek verilerin hazırlandığı gönderim kısmı ve alınan verilerin çözüldüğü alma kısmı olmak üzere iki kısma ayrılır.

Mantıksal blokta 8B/10B kodlama(encoding) ve çerçeve sinyallerinin eklenmesi işlemi yürütülür. 8B/10B kodlama 1394b, SATA, USB 3.0 gibi yaygın kullanılan bir çok protokolde kullanılmaktadır. Temel olarak 8 bitlik veri, özel şekilde kodlanarak 10 bitlik sembollere dönüştürülür. Bu dönüşüm ile herbir 8 bit için 10 bit gönderildiğinden hız düşmektedir ancak elektriksel hatta DC dengesi(balance), hattan iletilen 1 ve 0 adetlerinin farklarını kontrol altında tutma ve veride yeterli sayıda 1 ve 0 geçişi elde ederek clock kurtarma (clock recovery) gibi çok faydalı özellikler katmaktadır[12].

Elektriksel blokta öncelikle mantıksal bloktan paralel şekilde gelen veriler TX hatlarından iletilmeye uygun şekilde seri bit katarına (bit stream) dönüştürülür. Ardından bu veri katarı TX+ ve TX- diferansiyel hatlarından gönderilmek üzere LVDS 'e dönüştürülür. Fiziksel katman blokları ve veri akışı Şekil-7 'de verilmektedir.



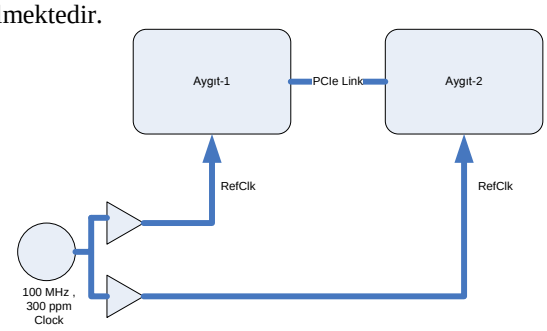
Şekil 7 : Fiziksel Katman Blokları ve Görevleri

C. *PCIe Clock Yapısı*

PCIe standardı alma ve gönderme aygıtlarının işlevlerine yerine getirebilmesi için 100 MHz ve en kötü ±300ppm frekans stabilizasyonuna sahip bir referans clock (RefClk) tanımlamaktadır[3]. Bunun yanında RefClk'a ilişkin 3 mimari tanımlanmaktadır. Bunlar Ortak RefClk Mimarisi, Ayrık RefClk Mimarisi ve Veriye Gömülmüş RefClk Mimarisidir. Bu mimarilerden en yaygın kullanımı olan Ortak RefClk 'tur. Çünkü bu mimarinin kullanımı "Spread Spectrum Clocking" tekniğinin kullanımına izin vermektedir[3]. Bu teknikle clock sinyalinin tek bir frekans bileşeni ile değil, temel 100 MHz etrafında, spektruma dağılmış şekilde iletilmesi sağlanmaktadır. Bu sayede EMI (Electromagnetic Interferance) azaltılmaktadır[13]. Ancak standardın belirlediği iki aygıta sağlanan clocklar arasındaki kaymanın 12 ns olduğu göz önünde bulundurulduğunda fiziksel olarak büyük boyutlardaki kartlarda bu değerin tutturulması problem olmaktadır[4].PCIe'de kullanılan clock mimarileri aşağıda detaylandırılmıştır.

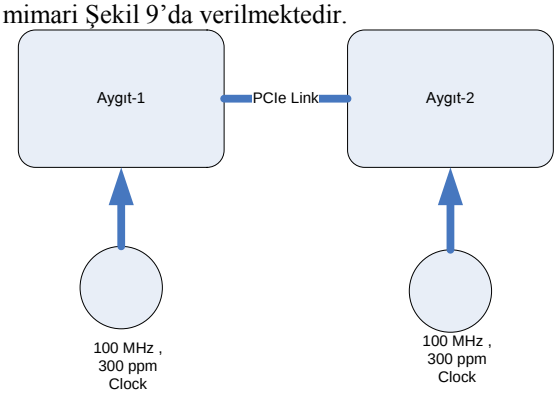
1) *Ortak RefClk :*Bu mimaride iki aygıta beslenecek RefClk tek bir kaynaktan üretilmektedir. Genellikle aygıtlar

bu mimariyle çalışmaktadır. Mimari Şekil 8'de verilmektedir.



Şekil 8 : Ortak RefClk Mimarisi

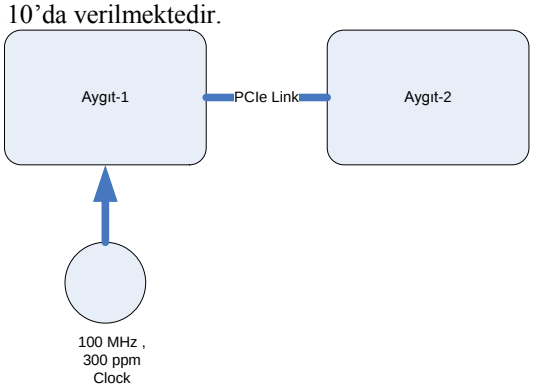
2) *Ayrık RefClk :* Bu mimaride aygıtlar farklı RefClk'larla beslenmektedir. Uygulanacak clock sinyallerinin 100 MHz ±300 ppm olması çok önemlidir[5]. Bu mimari Şekil 9'da verilmektedir.



Şekil 9 : Ayrık RefClk Mimarisi

Bu mimari Spread Spectrum Clocking desteklememektedir. Mimarinin avantajı ortak RefClk mimarisinde olduğu gibi, tek bir clock sinyalini tüm yapıda elektronik kartlar, backplane ve konnektörler boyunca taşınmamaktadır. Bunun yerine PCIe aygıtının ilgili girişlerinin yakınına yerleştirilecek bir clock kaynağı ile mimari gerçeklenmiş olmaktadır.

3) *Veriye Gömülmüş RefClk :*Bu mimari PCIe 2.0 ile desteklenmeye başlamıştır. Burada sadece gönderici aygıtı besleyen tek clock kaynağı mevcuttur. Diğer aygıt clock sinyalini gelen veriyi kullanarak türetmektedir.Bu mimari Şekil 10'da verilmektedir.



Şekil 10 : Veriye Gömülmüş RefClk Mimarisi

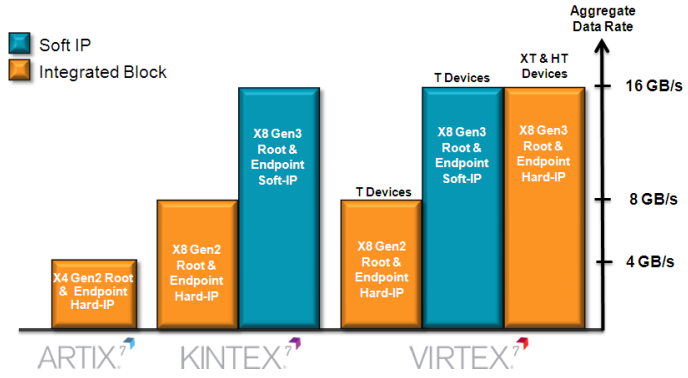
Bu mimarinin avantajı çok basit oluşudur. Çünkü tek clock kaynağı kullanılmaktadır. Ancak bu mimari 2007 yılında PCIe 2.0 ile desteklenmeye başlandığı için piyasadaki PCIe aygıtlarının bazıları tarafından desteklenmemektedir.

D. FPGA 'de PCIe desteği

Bilindiği gibi piyasada bir çok FPGA ailesi mevcuttur. Bu çalışma kapsamında deneyler daha çok Xilinx FPGA 'ler üzerinde yapıldığı için bu aileden bahsedilecektir.

Xilinx FPGA 'lerde PCIe desteği Spartan-3 'lerle birlikte başlamıştır. Spartan-3 serisi FPGA 'lerde harici bir fiziksel katmanla ve Soft IP desteği ile gerçeklenen PCIe[14], Spartan-6 serisi FPGA 'lerde MGT (Multi GigaBit Transiver) desteği ile harici fiziksel katmana gerek duymadan SoftIP desteği ile gerçeklenebilmektedir[6]. Xilinx Geliştirme Platformu ile ücretsiz gelen SoftIP tek veri yolu ve Gen 1 sonlanım tipi aygıt desteklemektedir. Kök complex olarak görev yapamamaktadır.

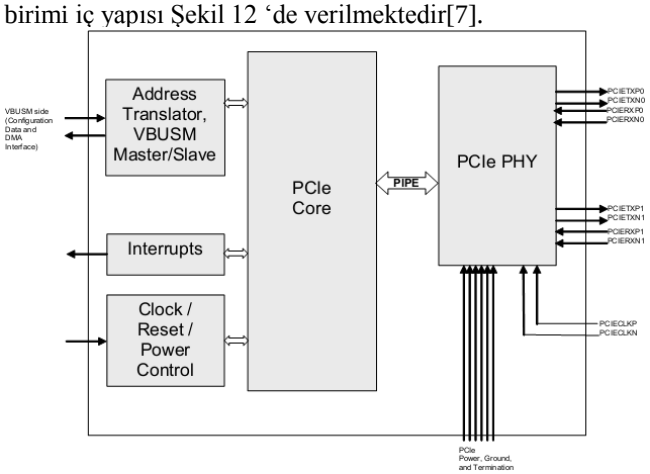
Xilinx PCIe desteğini 7 serisi FPGA'lerde geliştirerek sürdürmüştür. Bu FPGA'lerin destekleri ve hızları Şekil 11'de verilmektedir[11].



Şekil 11 : Xilinx 7 Serisi PCIe desteği

E. DSP 'de PCIe desteği

Bu çalışma kapsamında Texas Instruments firmasının 8 çekirdekli DSP serisi olan TMS320C6678 üzerinde çalışmalar gerçekleştirilmiştir. DSP 2 veri yolu Gen2 kök kompleks veya sonlanım desteklemektedir. DSP PCIe çevre birimi iç yapısı Şekil 12 'de verilmektedir[7].

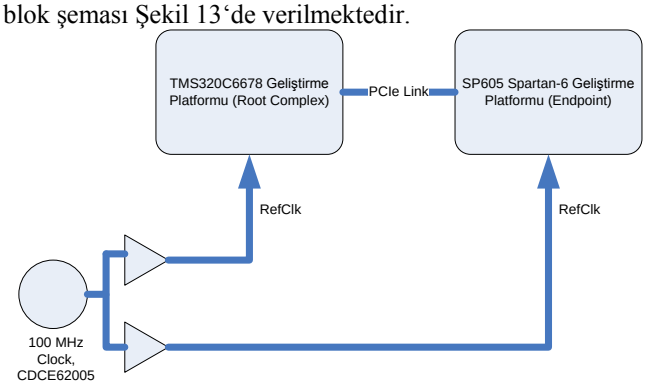


Şekil 12 : TMS320C6678 PCIe Çevre Birimi Yapısı

Çevre biriminde "Address Translator, VBUSM Master/Slave" kısmı PCIe çevre birimi tarafından çözülen TLP'lerin veri yükünü ilgili adres bölgesine yerleştirir. Böylelikle DSP çekirdeği PCIe TLP'leri ile ilgilenmez. Veri gönderme için PCIe çevre birimine programlanan adres bölgelerine herhangi bir veri yazıldığında bu veri otomatik olarak TLP haline getirilerek PCIe üzerinden gönderilir. Maksimum kanal kapasitesini kullanmak için belleğe yazma işlemi EDMA (Enhanced Direct Memory Access) ile yapılabilir.

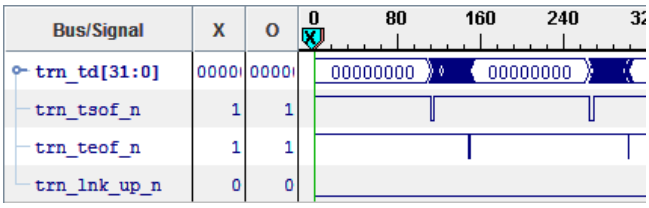
F. Örnek Uygulama

Çalışma kapsamında SP605 Spartan-6 geliştirme kiti ile TMS320C6678 Geliştirme Kiti Ortak RefClk mimarisi kullanılarak PCIe üzerinden birbirine bağlanmıştır. SP605 üzerindeki FPGA XC6SLX45T SoftIP desteği ile PCIe Gen1 tek veri yolu desteklemektedir. Bu nedenle TMS320C6678 2 veri yolu Gen2 desteklediği halde, haberleşme 1 veri yolu Gen1 hızında yapılmıştır. Bağlantı blok şeması Şekil 13'de verilmektedir.



Şekil 13 : Örnek Uygulama Bağlantı Şeması

Yapılan uygulamada PCIe üzerinden görüntü transferi gerçekleştirilmiştir. Analog kamera üzerinden alınan video sinyali çözülerek Spartan-6 tarafından her 32 pixel'de bir DSP'ye aktarılmıştır. Giden PCIe veri katarlarının bir kesiti Şekil 14'de verilmiştir. Burada görüntü Xilinx ChipScope yazılımı vasıtasıyla FPGA içerisindeki sinyalleri izleyek elde edilmiştir.Şekil 14 'de verilen trn_td değişkeni giden PCIe TLP 'yi gösterirken, trn_tsof_n, TLP başlangıcını, trn_teof_n ise TLP bitişini göstermektedir. Trn_lnk_up_n değişkeni, DSP ile FPGA arasındaki linkin durumunu göstermektedir.



Şekil 14 : PCIe Veri Katarları

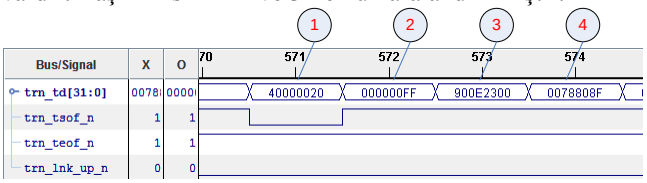
Uygulamada bir TLP 128 Byte veri yükü taşıyacak şekilde tasarlanmıştır. Esasen DSP 256 Byte, FPGA ise 512 Byte veri yükü boyutunu desteklemektedir. Teorik olarak veri yükü ne kadar büyük seçilirse, veri gönderim verimliliği o seviye artmaktadır. Örneğin TLP içerisinde 4 Byte veri yükü olacak şekilde bir tasarım yapıldığında en az 12-Byte'lık başlık bu verinin başına ekleneceğinden 4-Byte için 16-Byte veri gönderilmiş olacaktır. Ancak veri yükü 256-Byte seçildiğinde, 256-Byte için 268-Byte gönderileceğinden kanal kapasitesi çok etkilenmeyecektir. Teorik olarak durum bu olsa da aygıtların donanımsal özelliklerinden dolayı pratikte her zaman bu durum geçerli olmamaktadır. Örneğin DSP'de çeşitli veri yükü boyutlarına göre elde edilen iletim kapasitesi Tablo 1 'de verilmektedir.

Tablo 1: Veri Yükü – İletim Kapasitesi Tablosu

Veri Yükü Boyutu	İletim Kapasitesi
64	717 MB/s
128	780 MB/s
256	530 MB/s

Teorik değerlerle pratik değerlerin uyuşmamasının nedeni kullanılan aygıtın (TMS320C6678) üzerindeki gecikmelerdir. Bu durum bütün PCIe aygıtlarda geçerli olmayabilir, ancak gömülü sistemlerde PCIe kanal kapasitesi planlaması yapılırken bu durumun göz önünde bulundurulması gerekmektedir.

Şekil 15'de TLP başlangıcı daha detaylı gösterilmektedir. Analiz edilen bölgede 3 DW başlık ve 1 DW veri mevcuttur. Paketin devamında 31 DW veri daha vardır. Başlık kısmı 1-2 ve 3 ile numaralandırılmıştır.



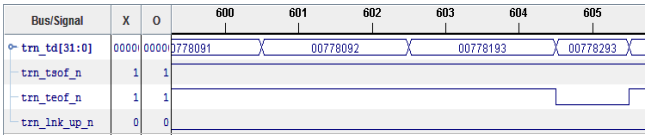
Şekil 15 : TLP Başlık Yapısı

1 numaralı kısımda bulunan veri Şekil 5'de verilen Byte 0-3 arasında kalan bölgedir. Sonda yazan 0x20 değeri TLP boyutuna işaret etmektedir. Buradan paketin 32 DW veri taşıdığı anlaşılmaktadır.

2 numaralı kısım yine Şekil-5'de verilen Byte 4-7 arasında kalan bölgedir. Burada bulunan 0xFF değerinden veriye herhangi bir maskeleme uygulanmadığı anlaşılmaktadır.

3 numaralı kısım Şekil-5'de verilen Byte 8-11 arasında kalan bölgedir ve TLP'nin gönderdiği adresi taşımaktadır. Şekil 15'deki örnekte paket 0x900E2300 adresine gönderilmektedir. Dikkat edilirse adres 32-bit'tir. Dolayısıyla TLP başlık 3 DW içermektedir.

4 numara ile işaretlenen kısım paketteki ilk veridir. Paketin içinde toplam 32 adet daha veri mevcuttur. Paketin bitişi Şekil 16'da verilmektedir. Dikkat edilirse TLP içerisinde opsiyonel bırakılan TLP özet kısmı mevcut değildir.



Şekil 16 : TLP Sonu

DSP gelen bu verileri, otomatik olarak, Şekil 12'de verilen "Address Translator, VBUSM Master/Slave" yardımı ile, paketin içindeki adres kısmında yazan bölgeye yerleştirir. Bu işlem yapılırken DSP çekirdekleri meşgul

edilmez. Yukarıdaki örneğe dönecek olursak gelen 32 DW 'lik veri, 0x900E2300 adresinden başlanılarak yerleştirilir. DSP adres haritasında bu bölgenin fiziksel olarak mevcut ve PCIe çevre birimi tarafından erişilebilir olması gerekmektedir.

Gerçekleştirilen bu uygulama ve elde edilen kanal kapasitesi ile gri seviye 575 satır analog video için 300 fps aktarım hızı elde edilmiştir. DSP belleklerine video bu hızda yerleştirilmektedir. Uygulamada 8 çekirdekli DSP kullanılmasının nedeni bu hızda belleklere yerleştirilen video işaretinin yine aynı hızlarda çeşitli algortimalarla işlenebilmesini sağlamaktır.

III. SONUÇLAR

Bu çalışma kapsamında gömülü sistemlerde DSP ve FPGA'lerin ortak kullanımı sonucu doğan haberleşme probleminin, PCIe protokolü ile çözüm yöntemine değinilmiş, bu yöntemin performansı örnek uygulama ile irdelenmiştir. DSP ve FPGA'lerin hız ve kapasitelerinin her geçen gün arttığı düşünüldüğünde, bu iki aygıtın haberleşme için gerçeklenen kanalın performansı toplam sistemin performansında kritik rol oynamaktadır. Nitekim kullanılan aygıtlar ne kadar hızlı olursa olsun, gerçek sistem kapasitesini verinin ne hızlarda paylaşıldığı belirlemektedir. PCIe piyasada kullanılan yaygın, yüksek performans haberleşme yöntemlerinden biridir. Ayrıca çoğu aygıt tarafından herhangi bir lisans ücretine gerek olmaksızın desteklenmektedir. Bu özellikleri diğer haberleşme yöntemlerine gore üstünlüklerini oluşturmaktadır.

Bu çalışmayı olabildiğince açık ve amaca yönelik yapmak için PCIe protokolünün bazı detaylarına yer verilememiştir. PCIe ile ilgili tüm detaylara PCI-SIG internet sitesi http://www.pcisig.com/home 'dan erişilebilir.

IV. TEŞEKKÜR

Argenç çalışanlarına bu çalışmaya verdikleri tüm katkılarından dolayı ve GÖMSİS organizasyon yetkililerine bu ve benzeri çalışmaları duyurma fırsatı verdikleri için teşekkür ederiz.

V. KAYNAKÇA

[1] PCI-SIG., "PCI Express Base Specification Revision 1.0a" , April 2003.

[2] PCI-SIG., "PCI Express Base Specification 2.0 Revision 0.9" , September 2006

[3] PCI-SIG, "PCI Express Card Electromechanical Specification Revision 1.1", March 2005.

[4] Silicon Labs, "Selecting the Optimum PCI Express Clock Source Rev 1.0"

[5] PCI-SIG,"PCI Express Jitter and BER Revision 1.0",February 2005

[6] Xilinx,"Spartan-6 FPGA Integrated Endpoint Block for PCI Express User Guide v3.0", April 2010.

[7] Texas Instruments, "KeyStone Architecture Peripheral Component Interconnect Express (PCIe) User Guide",December 2010.

[8] Adam H. Wilen, Justin P. Schade, Ron Thornburg, Intel, "Introduction to PCI Express", 2003

[9] PCI-SIG., "PCI Express Base Specification 3.0" http://www.pcisig.com/specifications/pciexpress/base3/.

[10] PCI-SIG, "Membership Roster", http://www.pcisig.com/membership/about_us/membership_roster/.

[11] Xilinx, "PCI express", http://www.xilinx.com/technology/protocols/pciexpress.htm

[12] Al X. Widmer, Peter A. Franaszek (1983). "A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code". IBM Journal of Research and Development 27 (5): 440.

[13] Miguel Rodriguez, PLX Technology and Lee Mohrmann, National Instruments," Spread-spectrum clocking reduces EMI in embedded systems",October 7, 2010, http://www.edn.com/design/analog/4363708/Spread-spectrum-clocking-reduces-EMI-in-embedded-systems-item-2.

[14] LogiCORE™ IP Endpoint PIPE v1.8 for PCI Express® User Guide July 23, 2010, http://www.xilinx.com/support/documentation/ip_documentation/pcie_pipe_ug167.pdf

Gömülü Sistemler ve Uygulamaları Sempozyumu - GÖMSİS 2012
İstanbul Teknik Üniversitesi, 29-30 Kasım 2012, İstanbul.

Güvenli RFID Sistemleri İçin Bir Kimlik Doğrulama Protokolünün Gerçeklenmesi

Semih Alparslan ve Berna Örs Yalçın
İstanbul Teknik Üniversitesi
Elektronik ve Haberleşme Mühendisliği Bölümü
Maslak, 34469, İstanbul
e-posta: siddika.ors@itu.edu.tr

***Özet*—Günümüzde Radyo Frekansı ile Tanımlama (Radio Frequency Identification, RFID) sistemlerinin kullanımı hızla yaygınlaşmaktadır. Bu sistemlerin kullanımının yaygınlaşması güvenlik açıklarını da beraberinde getirmektedir. Birçok uygulama sahasına girmiş bu sistemlerin güvenlik açıklıklarına sahip olması ve bu güvenlik açıklıklarının kötü amaçlı insanlar tarafından kullanılması ciddi bir problem teşkil etmektedir. Bu nedenlerden dolayı RFID sistemlerin güvenli şartlar altında kullanılması zorunluluk haline gelmiştir. Bu güvenli şartları sağlamak için RFID uygulamalarında kriptoloji algoritmalarının kullanılması en akılcı çözümdür.**

Bu çalışmada güvenli bir RFID sistem oluşturmak için bir kimlik doğrulama protokolü gerçeklenmiştir. Bu protokolün nasıl gerçeklendiği daha iyi kavrayabilmek adına ön bilgiler olarak öncelikle RFID sistemlerinden ve onun ana öğelerinden bahsedilmiştir. Sonrasında, gerçekleme aşamasında kullanılan tasarım araçları tanıtılmıştır. Son olarak, güvenli RFID sistemi tasarlamak için kullanılan protokol detaylı bir şekilde anlatılmıştır. Protokol gerçeklenirken gömülü sistemlere olan uygunluğu sebebiyle şifreleme algoritması olarak Küçük Şifreleme Algoritması (Tiny Encryption Algorithm, TEA) kullanılmıştır.

Protokolü gerçekleme aşamasında, öncelikle, sistemde kullanılacak olan donanım parçaları olan şifreleme, şifre çözme ve rastgele sayı üreteci blokları Verilog donanım tanımlama dilinde tasarlanmıştır. Donanımların hepsinin tamamen çalışır durumda olduğu test edildikten sonra tüm sistemi ve bu donanım bloklarını kontrol etmek amacıyla Microblaze mikroişlemcisi üzerinde C dili ile yazılım tasarımı yapılmıştır. Son olarak donanım ve yazılım tasarımı yapılan güvenli RFID sistem Sahada Programlanabilir Kapı Dizinleri (Field Programmable Gate Array, FPGA) üzerinde gerçeklenmiştir.

I. Giriş

Son yıllarda otomatik tanımlama işlemleri satın alma, dağıtım lojistiği, sanayi, üretim şirketleri ve malzeme akış sistemleri gibi birçok sektörde çok popüler hale gelmiştir. Ayrıca insanlar, hayvanlar, mallar ve nakledilen ürünler hakkında bilgi tutulması da otomatik tanımlama işlemleri ile daha işlevsel hale gelmiştir [1]. Bu nesnelerin otomatik tanımlanması işlemleri için Radyo Frekansı ile Tanımlama Sistemleri kullanmak oldukça etkili bir yaklaşımdır. Her gün birçok nesneye onları tanımlamak için RFID sistemler eklenmektedir.

RFID sistemleri arasındaki haberleşmeyi sağlamak için ISO/IEC 18000 standardı bulunmaktadır. Ancak bu Standard radyo frekansı ile doğrulamanın nasıl yapılacağı hakkında bilgi vermemektedir. RFID sistemlerin doğrulama işlemi hakkında bilgi vermemesi güvenlik açısından sorgulanmalarına yol açmaktadır. RFID sitemlerine doğrulama uygulamasının eklenmesinin en iyi yolu şifreli doğrulamanın kullanılmasıdır [2].

Bu çalışmada güvenli bir RFID sistemin şifreleme algoritması da kullanılarak seçilen protokol doğrultusunda nasıl tasarlanacağı gösterilmektedir. Doğrulama mekanizması ile ilgili olarak temel hedef, gönderilecek ya da alınacak olan mesajın karşılıklı olarak doğrulama işlemine tabii tutulmasıdır. Doğrulama işlemi basamakları; okuyucu biriminin etikete rastgele mesaj göndermesi, daha sonra etiketten bu mesajı şifreli olarak alması ve aldığı mesajın şifresini çözmesi şeklindedir. Eğer alınan şifresi çözülen mesaj ile gönderilmiş olan rastgele mesaj aynı ise etiket ve okuyucunun aynı şifreleme anahtarına sahip olduğu anlaşılmaktadır. Yani bu mekanizma aynı anahtara sahip olmayan etiket ve okuyucunun haberleşememesini sağlamaktadır.

Protokolde şifreleme işlemi Küçük Şifreleme Algoritması (Tiny Encryption Algorithm, TEA) ile sağlanacaktır. Doğrulama mekanizmasının okuyucu ve etiket birimleri Sahada Programlanabilir Kapı Dizinleri (Field Programmable Gate Array, FPGA) üzerinde gerçekleneceklerdir. Sistem gerçeklenirken etiket birimi genellikle taşınabilir oldukları için az güç tüketmesi ve az alan kaplaması amaçlanmıştır. Okuyucu birimi için ise birden çok doğrulamayı aynı anda yapabilmesi için hızlı sistem bir sistem tasarımı amaçlanmıştır. Bu doğrultuda şifreleme, şifre çözme ve rastgele sayı üreteci algoritmaları FPGA üzerinde donanım olarak gerçeklenmişlerdir. Bu donanımları ve tüm sistemi kontrol etmek için FPGA'nın içerisinde bulunan Microblaze mikroişlemcisi kullanılmıştır.

II. Radyo Frekansı İle Tanımlama Sistemleri

Radyo frekansı ile tanımlama sistemleri, durağan ya da hareket halindeki nesneleri tekil ve otomatik olarak radyo frekansı ile tanımlamak için kullanılır. Günümüzde tanımlama işlemi için birçok yöntem mevcut olsa da, son zamanlarda ilgiler radyo frekansı ile tanımlama üzerine yoğunlaşmış durumdadır. Geçtiğimiz dönemlerden beri klasik olarak kullanılan barkod sistemleri kolay kullanıma sahip olmalarına karşın, saklayabildikleri veri miktarlarının az olması ve barkot üzerindeki etiketin değerin değiştirilmesinin imkansız olması barkot sistemini dezavantajlı kılmaktadır. Bu esnek olmayan kullanımın çözümü, içerisinde akıllı

kartlar barındıran ve tanımlama için kablosuz cihazlar aracılığıyla radyo frekansını kullanan RFID sistemler ortaya çıkmıştır [2].

RFID sistemler ilk olarak 1940'lı yılların başlarında İngiltere'de dost ve düşman uçakların tanımlanmasında kullanılmıştır. Bunu 1970'li yıllarda nükleer malzeme izleme uygulamaları takip etmiş, ticari uygulamaları ise 1990'lı yıllarda başlamıştır. RFID sistemlerin uygulama alanlarına örnek olarak: ürün dağıtım zinciri uygulamaları, hasta tanımlama, kütüphane, müze, sanat galerisinde ürün tanımlama, taşımacılıkta değerli ürün izlenmesi gibi örnekler verilebilir [3].

RFID sistemleri temel olarak, tanımlanmak istenen nesnenin üzerine yerleştirilen "etiket" ve içerisinde bulunan anten vasıtasıyla bu etiket ile haberleşen "okuyucu" olmak üzere temel olarak iki ana öğeden meydana gelmektedir.

A. Etiket

RFID etiketi, radyo frekansını kullanarak okuyucudan gelen sinyalleri alan, sorgulayan daha sonra da cevaplayan ve tanımlamak istenen nesnelerin bilgilerini taşımak üzere nesnelere yerleştirilen sistem bileşenleridir. Taşınabilir oldukları için sınırlı hafıza kapasitelerine sahiptirler. Etiketler fonksiyonları bakımından aktif, yarı pasif ve pasif olmak üzere üçe ayrılır [3].

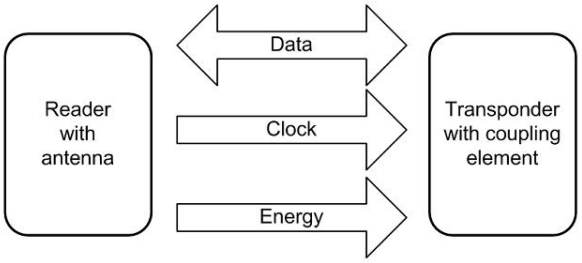
Aktif etiketler devrelerinin çalışmasını ve haberleşme için sinyal üretimlerini kendi güç kaynakları içlerinde barındırdıkları güç kaynağından sağlarlar. Kendi içinde barındırdıkları piller yani güç kaynakları sayesinde daha uzak haberleşme mesafeleri ve daha iyi çalışma performanslarına sahiptirler.

Yarı pasif etiketler de kendi güç kaynaklarını içerirler ancak içerdikleri bu güç kaynağı sadece kendi devresine güç sağlamaktadır. Haberleşme okuyucudan gelen sinyallerdeki güç ile sağlanmaktadır.

Pasif etiketler ise, üzerlerinde güç kaynağı barındırmazlar. Devrelerini ve haberleşmesini okuyucudan elektromanyetik dalga aracılığıyla aldığı güç ile beslemektedirler. Güç kaynakları içermedikleri için daha kısa mesafeli haberleşmelere olanak sağlarlar. Tercih edilme sebepleri, ucuz ve basit yapılı olmalarıdır. Bundan ötürü pasif etiketlerin en fazla yer aldığı uygulama alanları güç kaynağının uygulanamadığı, pil ömrünün daha öncelikli olduğu ve işlem kapasitesinin ikinci planda olduğu alanlardır.

B. Okuyucu

Bir RFID sistemde okuyucunun görevi, antenini kullanarak etiketleri uyarmak, etiketlerin içerdiği veriyi okumak ve bir ağ aracılığıyla bu veriyi bir sunucu bilgisayara göndermektir [4]. Okuyucular kullanım alanlarına göre mobil ya da sabitlenmiş olabilirler. RFID sistemlerde okuyucular aynı zamanda veritabanlarına bağlı olarak çalışırlar. Okuyucu, etiketi sorgulamak amacıyla etikete bir işaret gönderir ve bu işaret neticesinde uyarılan etiket kendi verisini okuyucuya geri gönderir. Okuyucu aldığı bu cevabı veritabanına göndererek veritabanında etiketin kayıtlı olup olmadığı sorgulanabilir. Sistemin iki ana öğesi olan okuyucu ve etiket arasındaki 3 hattan oluşan haberleşme Şekil 1'de görülmektedir. Bu haberleşme belirli frekanslarda gerçekleştirilmektedir. Bu hatlar; iki yönlü olmak üzere veri, sadece okuyucudan etikete doğru olan saat bilgisi ve aktif olmayan etiketler için yine sadece okuyucudan etikete doğru olan ve etiketi beslemek için gönderilen enerji olarak tanımlanır [2].



Şekil 1. : Etiket ve okuyucu arasındaki hatlar

Etiket ve okuyucunun hangi formatlarda haberleşeceği, haberleşirken hangi modülasyonu kullanacakları, girişim engelleme metotları ve protokol parametreleri ISO/IEC 18000-3 standardında tarif edilmiştir. Bu standarda göre, RFID okuyucu ve etiket 13.56 Mhz frekansında haberleşmektedir [5]. Okuyucuları etiketlerden ayıran en önemli özellik, genellikle taşınamaz yapıda oldukları için içlerinde güç kaynağı barındırmalarıdır. Bir diğer fark ise aynı anda birden çok etiketle haberleşme yapabilme kapasiteleridir.

III. Gerçeklenecek Protokol

Bu çalışmada Feldhofer'ın daha güvenli RFID sistemler tasarlanması adına yazmış olduğu kimlik doğrulama protokolü FPGA üzerinde gerçeklenecektir [2]. Feldhofer bu protokolde RFID sistem protokollerine yeni bir yaklaşım sunmaktadır. Günümüzde şifreli tanımlama sistemleri ürünlerin güvenliklerini sağlamak için zorunlu hale gelmiştir.

Bu protokol günümüzdeki diğer protokollerden ayrı olarak şifreli şekilde haberleşmeyi yeni bir yaklaşım olarak sunmaktadır. Protokolde etiket için sınırlı işlem kapasitesi, alan kısıtlaması, düşük güç tüketimi gibi kısıtlamalar sebebiyle iki yönlü meydan ileti-cevap kimlik doğrulaması şeması önerilmiştir. Paket ve çerçeve biçimleri var olan protokollerde olduğu gibi ISO/IEC 18000 standardını içermektedir.

A. ISO/IEC 18000 Standardı

ISO/IEC 18000-3 Standardı okuyucu ile etiket arasındaki haberleşmenin hangi frekansta, hangi özellikler veya kısıtlamalar altında olacağını bildirmiştir.

Etiket ile okuyucu arasındaki haberleşme modülasyon ile çalışmaktadır. Okuyucu haberleşmek için endeksi % 10 ve % 100 olan Genlik Kaydırmalı Anahtarlama (Amplitude Shift Keying - ASK) modülasyonu kullanmaktadır. Veri şifrelemesi "256'da 1" ya da "4'te 1" veri şifreleme biçimi ile mümkün olur. Veri şifreleme biçimine göre çıkış yolu oranı saniyede 26.69 kbit hızına ulaşabilmektedir [5].

Söz konusu iletişim protokolü okuyucu ve etiket arasındaki komutların ve verilerin iki yönlü olarak aktarılma şeklini tanımlar. Bu protokol "okuyucu önce konuşur" ilkesine dayanmaktadır. Bu ilke, hiçbir etiketin okuyucudan direktif

alıp bunu tam anlamıyla çözmeden iletime başlamaması gerektiğini açıklar. Her komut, okuyucudan etikete olmak üzere 'istek' ve etiketten okuyucuya olmak üzere 'yanıt' içermelidir. Bu istek ve yanıtlar bir çerçeve içinde Çerçeve Başlangıcı (Start of Frame - SOF) ve Çerçeve Sonu (End of Frame - EOF) ile sınırlandırılmıştır. Bu başlangıç ve bitiş sınırlarının arasında her bir istek ve yanıt çerçevesi; Bayrak, Komut Kodu, Parametreler ve Çevrimsel Hata Denetimi (Cyclic Redundancy Check - CRC) içermektedir [5].

- Bayraklar: Bir ya da iki alt taşıyıcı frekansını ve yanıt için hangi veri oranının kullanılması gerektiğini göstermektedir. Uygun görülen etiketleri adreslemek için ekstra bilgiler sunulmaktadır. Etiketin yanıtı bayrakları kullanarak haberleşme sırasında oluşan hataları göstermektedir.
- Komut Kodu: Bir baytlık sabit, hangi isteğin gönderildiğini göstermektedir. Üç adet temel komut mevcuttur. Bunlardan birincisi olan zorunlu komutlar etiket tarafından gerçekleştirilmelidir. Seçmeli komutlar uygulama için gerekli ise etiket tarafından gerçeklenebilir. Özel komutlar ise kendi komutlarını protokole eklemek isteyen üreticiler tarafından kullanılabilir.
- Parametreler ve Veri Alanları: İstek ve Yanıtı işlemek için gerekli bilgileri barındıran özel komutlardır.
- CRC: Çevrimsel Hata Denetimi kendi hariç SOF'den sonra gelen bütün baytların belli bir algoritma içerisinde hesaplanmasıyla oluşturulmakta ve haberleşme esnasında herhangi bir hata olup olmadığını ortaya çıkarmak için kullanılmaktadır [5].

B. Doğrulama Mekanizması

Protokolde şifrelemeli doğrulama mekanizması kullanılmaktadır. Şifreli doğrulama yöntemleri gizli anahtar ve umumi anahtar şifreleme olarak ikiye ayrılmaktadır. Bu iki şifreleme yöntemi de bu protokol kapsamında kullanılabilmektedir. Doğrulama mekanizmasında iddiacı yani talep eden bir birim ve sağlayıcı bulunmaktadır. Bu iddiacı ve sağlayıcının ikisi de etiket ya da okuyucu olabilir. Protokol kapsamında sağlayıcı olan kısım davalı kısma rastgele bir sayı olabilecek kimlik sorma talebi gönderir. Davalı kısım bu rastgele sayıyı sahip olduğu gizli anahtarla işleyerek kimliğini kanıtlamak üzere tekrar sağlayıcı kısma gönderir. Sağlayıcı kısım davalıdan aldığı veriyi kontrol ederek davalının gizli anahtarı bilip bilmediğine bakar. Güvenlik için bu haberleşme esnasında gizli anahtarın üçüncü bir kişi tarafından ele geçirilmemesi gerekmektedir.

Bu protokolde Şekil 2'de görüldüğü gibi iki aşamalı kimlik sorma sistemi kullanılmıştır.

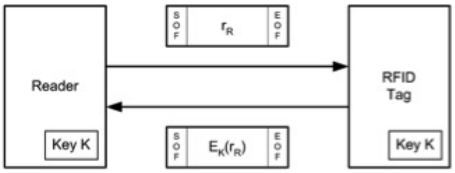
$$A \rightarrow B: \quad E_K(t_A)$$

$$A \leftarrow B: \quad r_B$$
$$A \rightarrow B: \quad E_K(r_B)$$

Şekil 2. Kimlik sorma sistemi

Bu sistemde öncelikle A birimi kendi içinde sakladığı zaman bilgisini şifreleyerek B birimine gönderir. Bu şifreli zaman bilgisini alan B birimi şifreyi çözerek aldığı zaman bilgisini kendi zaman bilgisiyle karşılaştırır. Eğer iki zaman birimi birbirine eşit ise protokolün ikinci aşamasına geçilir. İkinci aşamada, B birimi kendi içinde ürettiği rastgele sayıyı A'ya gönderir. A birimi aldığı bu sayıyı şifreleyerek B birimine gönderir. B aldığı bu şifreli sayıyı çözerek kendi yolladığı rastgele sayı ile karşılaştırır ve bu iki sayı eşitse kimlik sorgulama işlemi başarıyla tamamlanmış olur.

A ve B birimleri aslında Okuyucu ve Etiket bileşenleridir. Protokolün detaylı şeması Şekil 3'de gösterilmektedir.



Şekil 3. Doğrulama protokolü

Protokolde okuyucu etikete, rastgele sayı üreteci kullanarak ürettiği 128 bitlik r_R rastgele sayısını gönderir. Etiket ise rastgele sayıyı $E_K(r_R)$ şeklinde şifreleyerek okuyucuya gönderir. Okuyucu bu sayıyı alarak şifresini çözer ve yolladığı sayı ile karşılaştırır. Eğer sayılar aynı ise okuyucu etiketin kimliğini tanıdığından emin olur.

Okuyucu ve etiket arasındaki bu iletişimin sağlıklı olması için belirli bir çerçeve içerisinde sağlanmalıdır. İstek çerçevesi Şekil 4'te gösterildiği gibi olmaktadır.

SOF	Flags	0xA0	IC Mfg code	UID	Random number r_R	CRC	EOF
	8 bit	8 bit	8 bit	64 bit	128 bit	16 bit	

Şekil 4. İstek çerçevesi

İstek çerçevesi 8 bitlik bayraklar, 8 bitlik 0xA0 komutu, 8 bitlik kullanıcıya özel komut, 64 bitlik her etikete özgü UID (Unique Identifier), 128 bitlik rastgele sayı ve son olarak 16 bitlik CRC içermektedir.

Cevap çerçevesi de neredeyse istek çerçevesine benzer olarak Şekil 5'te görülmektedir.

SOF	Flags	UID	Signed data $E_K(r_R)$	CRC	EOF
	8 bit	64 bit	128 bit	16 bit	

Şekil 5. Cevap çerçevesi [2].

Cevap çerçevesi ise 8 bitlik Bayraklar, 64 bitlik her etikete özgü olan UID (Unique Identifier), 128 bitlik şifrelenmiş olan rastgele sayı ve son olarak 16 bitlik CRC içermektedir [2].

C. TEA Şifreleme Algoritması

Feldhofer'ın önerdiği protokolde şifreleme işlemini gerçekleştirmek için Gelişmiş Şifreleme Standardı (Advanced Encryption Standard - AES) kullanılmıştır [2]. Ancak günümüzde teknolojinin gelişmesiyle birlikte küçülen etiket

boyutlarıyla paralel olarak yeni şifreleme standartları ortaya çıkmaktadır. Gerek etiketin boyut kısıtları gerekse daha az enerji tüketme zorunluluğundan dolayı daha az enerji tüketen ve gerçeklendiğinde daha az yer kaplayacak şifreleme standartları kullanma zorunluluğu ortaya çıkmıştır. Bu zorunluluktan ötürü protokolün gerçeklenme aşamasında Wheeler ve Needham'in 1994 yılında gömülü sistem uygulamaları için geliştirmiş olduğu TEA kullanılmıştır. Gömülü sistemlerdeki yüksek performansı, gerçekleme kolaylığı, hızlı olması, düşük enerji tüketimine imkan vermesi, düşük masraflı olması ve güvenli olması hafif (lightweight) olması özelliği ile TEA gömülü sistem tasarımlarına oldukça uygundur [6].

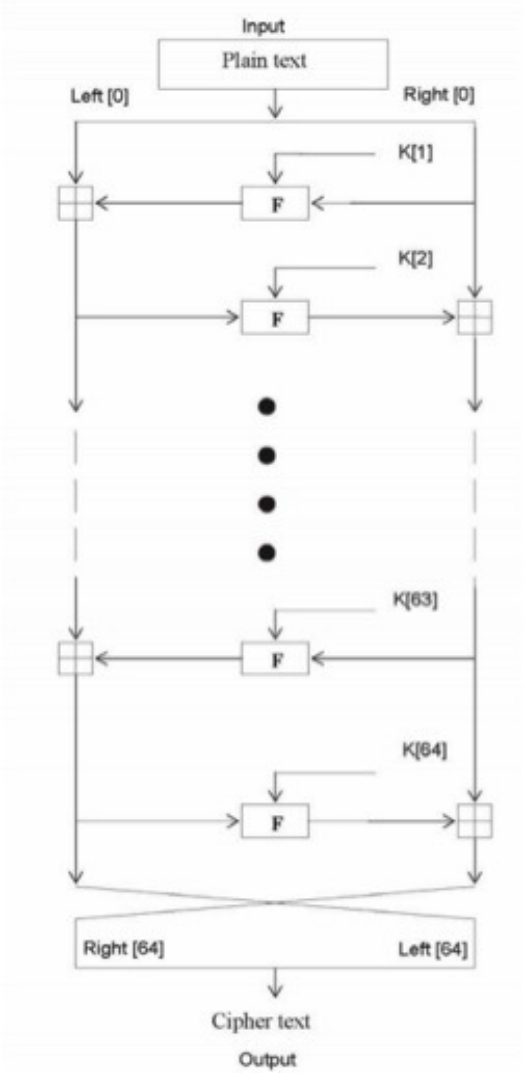
TEA minimum hafıza alanı ve maksimum hız hedeflenerek oluşturulmuş bir şifreleme algoritmasıdır. Karışık cebirsel işlemleri kullanan ve Feistel türü bir şifreleme yapan bir algoritmadır. Feistel türü şifreleme türü olduğu için blok şeklinde şifreleme yöntemine göre oluşturulmuştur. TEA farksal şifre analizine oldukça dirençlidir. Aynı zamanda sadece altı tur sonra tam yayınım sağlamaktadır. Bunun anlamı, şifrelenecek metinde 1 bit değiştirildiğinde çıkıştan alınan şifreli metine bu değişiklik 32 bit olarak yansımaktadır. Zaman performansı ise bilgisayarlarda ve iş istasyonlarında oldukça etkileyicidir [7].

TEA algoritması blok şifreleme yapısında olması sebebiyle girişine uygulanan 64 bitlik metni bit bit olarak şifrelemek yerine 64 biti tek blok olarak alarak sanki tek bir bitmiş gibi şifrelemektedir [7]. TEA simetrik yapılı şifreleme algoritması olması nedeniyle şifreleme ve şifre çözme algoritmaları yapıları birbirlerine oldukça yakındır.

TEA şifreleme ve şifre çözme yapısında 128 bit uzunluklu şifreleme anahtarı kullanılmaktadır. Bu 128 bit uzunluklu anahtar $K[0]$, $K[1]$, $K[2]$ ve $K[3]$ şeklinde dört adet 32 bit uzunluklu anahtarlara bölünerek işlemlere sokulur [7]. Şekil 6'te algoritmanın şifreleme yapan kısmının blok diyagram yapısı görülmektedir. Şifreleme yapısı 64 adet Feistel döngüsünden meydana gelmektedir. Şifrelenmek istenen metin 32'şer bitlik iki kısma bölünerek sağ ve sol taraftan şifrelenmek üzere döngüye sokulur. Her bir döngüde farklı anahtar kullanılmaktadır. Girişe uygulanan $Sol[0]$ ve $Sa[0]$ girişleri $K[0]$, $K[1]$, ..., $K[64]$ anahtarları tarafından şifrelenerek bu 64 döngünün sonunda $Sol[64]$ ve $Sa[64]$ çıkışlarından şifrelenmiş metin halinde çıkmaktadırlar [7]. Yapı, her bir döngünün girişi bir önceki döngünün çıkışına bağlanacak şekilde oluşturulmuştur. Her bir döngüye giren 64 farklı $K[i]$ anahtarı, 128 bit şifreleme anahtarının "delta" isimli altın orandan üretilen bir sabitin işleme sokularak oluşturulmaktadır. Delta sabitinin ilk değeri Denklem 1'den hesaplanmaktadır [7].

$$delta = \left(\sqrt{5} - 1\right) * 2^{31} = 9E3779B9_h \qquad (1)$$

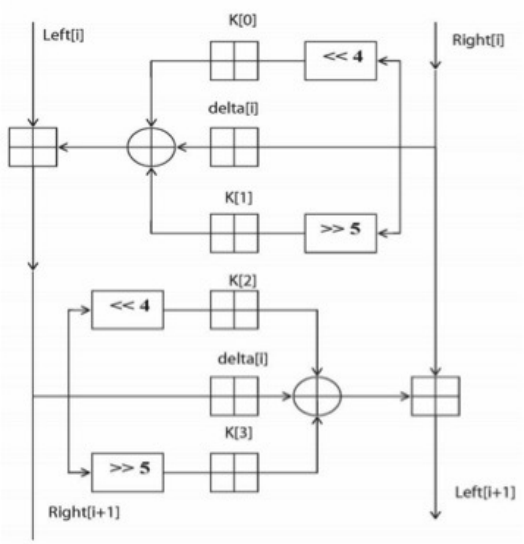
Şifreleme kısmının iç yapısına değinilecek olunursa, 64 Feistel döngüsünde oluştuğu bilinen şifreleme yapısının Şekil 7'da görüldüğü gibi 32 döngüden oluşmaktadır. Yani Şekil 7 iki tane Feistel döngüsü içermektedir. Her bir döngüde döngüye giren metinler toplama, özel veya (exclusive or - XOR), mantıksal kaydırma işlemlerine tabii tutulmaktadır.



Şekil 6. TEA şifreleme rutini [13].

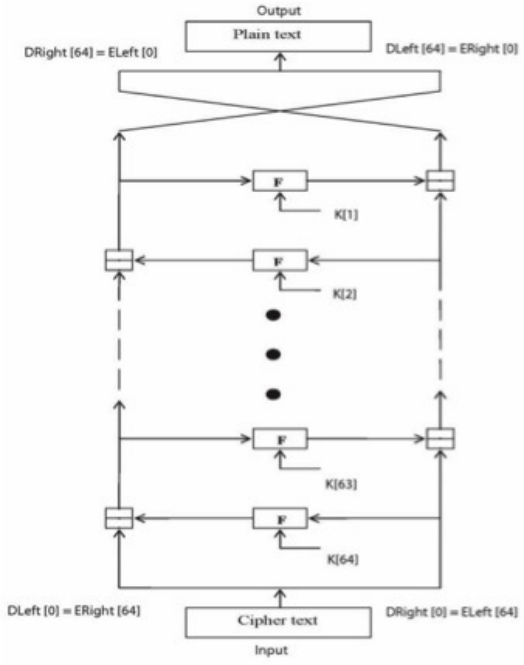
Şekil 7'da da görüldüğü gibi bir döngüye sağ taraftan giren şifresiz metine öncelikle 4 bit sola kaydırma işlemi uygulanır ve daha sonra $K[0]$ anahtarıyla toplama işlemine girer. Yine aynı bilgi 5 bit sağa kaydırılarak $K[1]$ anahtarıyla toplama işlemine girmektedir. Bir de bilginin kendisi direk olarak $delta[i]$ sabitiyle toplama işlemine girmektedir. Daha sonra işleme giren bu üç koldan gelen veriler XOR işlemine girerek sol koldan metinle toplanmaktadır. Bu toplamın sonucu o döngünün sol taraftan verdiği çıkışı olarak bulunmaktadır. Sol taraftan çıkan bu bilgi bir Feistel turu önce sağdan girmiş olan metinle aynı işlemlere tabii tutularak bu döngünün sağ taraftan verdiği çıkış sonucu elde edilmektedir. Bu işleme bu şeklide 32 tur devam edilerek son turun çıkışında girişten verilen şifresiz metinlerin şifreli halleri elde edilmektedir.

TEA simetrik yapılı şifreleme algoritması olması sebebiyle şifre çözme yapısı da şifreleme yapısıyla benzer olmaktadır. TEA Şifre çözme yapısı Şekil 8'de görüldüğü üzere şifreli metinin çıkıştan girişe doğru işlenmesi şeklindedir. Şifre çözme yapısını şifreleme yapısından ayıran



Şekil 7. TEA i. döngüsü [13].

bir diğer önemli nokta ise şifresi çözülecek metinin başlangıçta $K[64]$ anahtarı kullanılarak işleme sokulmasıdır. Yani anahtarların sırası da tamamen yer değiştirmiştir. Aradaki son fark ise sağ ve solda görülmekte olan ana kollardaki toplama işlemlerin yerini çıkarma işlemleri alması şeklinde olmaktadır.



Şekil 8. TEA şifre çözme rutini [13].

D. Rastgele Sayı Üreteci

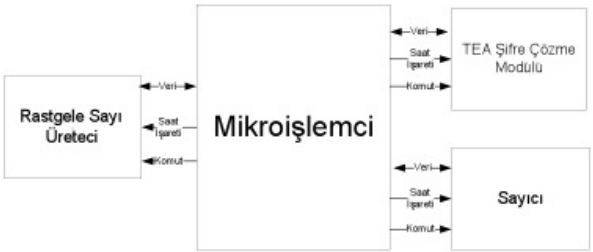
Protokolün doğrulama mekanizmasının gerçekleştirilmesi aşamasında okuyucudan etikete bir rastgele sayı gönderilmesi gerekmektedir. Bu 64 bit uzunluklu rastgele sayıyı üretmek için sistemin okuyucu kısmına rastgele sayı üreteci eklenmesi gerekmektedir. Sistemin daha güvenilir bir hale gelmesi açısından üretilen rastgele sayının tahmin edilemez olması önemlidir. Tamamen rastgele bir sayı üretmek ayrı bir tasarım yükü getireceğinden bu tez aşamasında ürettiği sayıların rastgele olduğu varsayılan doğrusal geri beslemeli ötelemeli kaydedici (Linear Feedback Shift Register - LFSR) kullanılmıştır.

LFSR donanım gerçeklemeleri için uygun olması, büyük periyotlu dizi üretimi özelliği, iyi istatistiksel özellikli dizi üretimi özelliği ve yapısının cebirsel teknikleri kullanarak basit bir şekilde ifade edilebilmesinden ötürü sıklıkta tercih edilmektedir [8].

Sözde-rastgele sayı üreteci olan LFSR girişine tohum yani başlangıç değeri uygulanarak her saat darbesi geldiğinde farklı bir rastgele sayıyı çıkışından vermektedir. L bit uzunluklu bir LFSR'nin çalışma yapısı, ilk olarak aldığı L bit uzunluklu tohum değerini her saat işareti geldiğinde bir düşük anlamlı bitine kaydırır. Her bit bir anlamsız basamağa kaydığı zaman boşta kalan en anlamlı bit olan $L-1$ bitine diğer bitlerin bir kaçının XOR işlemine tabii tutulmasıyla elde edilen değer atanır. LFSR belirli bir periyodu tamamlandığında başlangıç değerine geri dönmektedir. Bu sebepten dolayı daha güvenli sistemler gerçeklemek için bu periyotun mümkün olduğu kadar uzun tutulması gerekmektedir. Protokolün gerçekleme aşmasında 64 bit uzunluklu rastgele sayı gerekliliği olduğundan ötürü bu bit uzunluğunda bir LFSR için maksimum periyodu sağlayacak XNOR barındıran bir geri-besleme fonksiyonu kullanılmıştır.

E. Mikroişlemci

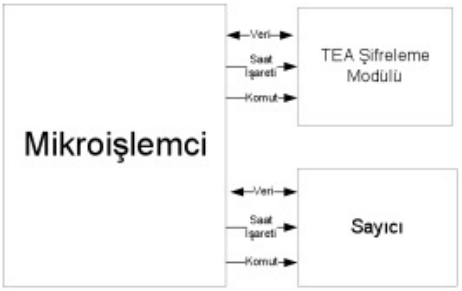
Bitirme tezi mikroişlemci merkezli yapı ile gerçeklenecektir. Tasarlanan donanımlar ve sistemin tamamı üzerinde yazılım koşturulan mikroişlemci tarafından kontrol edilecektir. Şekil 9'de tasarlanacak okuyucu kısmının genel yapısı görülmektedir. Protokolde okuyucu kısmında bulunan mikroişlemci şifre çözme donanımını, sayıcı donanımını ve rastgele sayı üretecini kontrol etmekle sorumludur.



Şekil 9. Mikroişlemcili okuyucu yapısı.

Şekil 10'da ise etiket yapısının genel hali görülmektedir. Etiket kısmındaki mikroişlemcinin görevi ise şifreleme donanımını ve sayıcı donanımını kontrol etmektir.

Her iki yapıda da mikroişlemci donanımlarla iki taraflı veri haberleşmesini ve tek taraflı komut ve saat işareti göndermeyi sağlayacaktır. Mikroişlemcinin bir diğer görevi ise okuyucu kısmı için etiketle, etiket kısmı için de okuyucuyla arada olan haberleşmeyi sağlamaktır. Bu tezde sistemi daha da hızlandırmak adına sayıcı donanımı kullanmak yerine



Şekil 10. Mikroişlemcili etiket yapısı.

yazılım aşamasında değişken olarak tanımlanıp kontrol edilecektir.

IV. Kimlik Doğrulama Protokolünün Gerçeklenmesi

Protokolün gerçeklenmesi aşamasına ilk olarak protokolde yer alan donanımların Verilog donanım tanımlama dili kullanılarak FPGA üzerinde tasarlanmasıyla başlanılmıştır. Protokolde bulunan donanımlar; etiket kısmında yer alan TEA şifreleme donanımı, okuyucu kısmında yer alan TEA şifre çözme donanımı ve yine okuyucu kısmında bulunan rastgele sayı üreteci donanımlarıdır. Donanım tasarımı aşaması tamamlandıktan sonra bu donanımları ve tüm sistemi kontrol etmek amacıyla FPGA içerisindeki Microblaze mikroişlemcisi üzerinde yazılım tasarımı yapılmıştır.

A. Donanım Tasarımı Aşamaları

Şifreleme ve şifre çözme donanımlarının bir turunun yapısı Şekil 7'da gösterilmişti. Bu yapıya istinaden şifreleme ve şifre çözme donanımlarından önce bu iki donanımın alt blokları olan toplama, çıkarma, mantıksal kaydırma ve özel veya donanımları tasarlanmıştır. Daha sonra tasarlanan toplama, çıkarma, mantıksal kaydırma ve özel veya donanımları kullanılarak şifreleme, şifre çözme donanımlarının tasarlanması tamamlanmıştır. Bir diğer donanım olan rastgele sayı üreteci diğer donanımlardan bağımsız olarak tasarlanmıştır.

Donanım tasarımı yapılırken etiket ve okuyucunun gereksinimlerine uygun yapılar göz önünde bulundurulmuştur. Bunlardan birinci derecede önem arz eden gereksinimler, pasif etiketler kendi içlerinde güç kaynağı barındırmadıkları için etiket kısmı için yapılacak tasarımın oldukça az enerji tüketecek yapıda olmasıdır. Yine etiket kısmı için bir diğer önemli tasarım gereksinimi ise etiketlerin taşınabilir olması nedeniyle alanda daha az yer kaplayacak şekilde tasarlanması hedeflenmiştir. Okuyucu kısmı için tasarım hedeflerine değinilecek olunursa, birçok etiketin aynı anda okuyucu ile haberleşme yapma olasılığını göz önünde bulundurarak okuyucunun etiketlere göre çok daha hızlı sistem yapısına sahip olması gerekmektedir. Okuyucular genelde taşınamaz yapılar olmaları nedeniyle kendi içlerinde güç kaynakları barındırmamaktadırlar. Okuyucuların güç ve kapladıkları alan konusunda sıkıntıları olmadığı için okuyucu kısımdaki donanımların tasarımı yapılırken olabildiğince hızlı sistem tasarımı hedeflenmiştir.

1) Toplama Bloğu: Şifreleme ve şifre çözme yapılarında 32 bit uzunluklu iki sayıyı toplamak amacıyla toplama bloğu tasarlanmıştır. Sistemin daha hızlı hale getirmek amacıyla seri toplama yerine paralel toplama bloğu tasarlanmıştır. Toplama bloğu tasarımında ilk olarak iki yarı toplayıcı yapısı kullanarak bir tam toplayıcı elde etme amaçlanmıştır. Ancak elde edilen 32 bit uzunluklu tam toplayıcının ISE'nin kullanıcıya sunmuş olduğu "+" operatörü kullanılarak tasarlanan toplayıcıdan daha fazla alan kapladığı görülmüştür. Bunun üzerine toplama bloğu "+" operatörü kullanılarak tasarlanmıştır. Toplama bloğu şifre çözme ve şifreleme yapılarında çok sayıda kullanılacağı için tasarımda az yer kaplaması oldukça önem arz etmektedir.

Toplama bloğu iki adet 32 bit uzunluklu sayıyı girişlerinden alarak bu iki sayının toplamlarını "f" çıkışından yine 32 bit uzunluklu olarak vermektedir. Toplama işleminin sonucu belirli değerler toplandığında 33 bit uzunluklu olduğunda ise en anlamlı biti atarak diğer 32 biti çıkışa vermektedir.

2) Çıkarma Bloğu: Şifre çözme donanımının tasarlanması aşamasında gerekli olan bir diğer blok ise 32 bit uzunluklu iki sayının farkını veren çıkarma bloğudur. Tasarım aşamasında toplama bloğunda olduğu gibi çıkarma bloğunda da en az alan kaplayan ve en az kapı gecikmesine sahip blok "-" operatörünü kullanarak elde edilmiştir.

Çıkarma bloğunun çalışma prensibi bloğun 'a' ve 'b' girişlerine uygulanan 32 bit uzunluklu sayıların farkını alarak yine çıkışa 32 bit uzunluklu sonucu vermektir.

3) Mantıksal Kaydırma Bloğu: Mantıksal Kaydırma Bloğu hem şifreleme hem de şifre çözme yapısında kullanılmıştır. Şekil 7'da da görüldüğü üzere her bir döngüde iki tane sola 4 bit kaydırma işlemi 2 tane de sağa 5 bit kaydırma işlemi bulunmaktadır. Bu sağa ve sola bit kaydırma işlemleri tek bir blok tasarlanarak gerçekleştirilmiştir. Sadece içerisindeki parametreler değiştirilerek bu blok istenilen uzunlukta istenilen tarafa doğru kaydırma işlemi yapabilmektedir.

Blokta 5 bit uzunluklu 'lenght', 32 bit uzunluklu 'value_in' ve 1 bit uzunluklu 'direction' girişleri bulunmaktadır. Bloğun çıkışı ise 32 bit uzunluklu 'result' değeridir. Blokta yön seçme girişi yüksek iken blok sol tarafa kaydırma işlemi yön seçme girişi düşük iken sağ tarafa doğru kaydırma işlemi yapmaktadır. Bloğun ne kadar uzunlukta kaydırma yapacağı ise bloğun 'lenght' girişi değiştirilerek ayarlanmaktadır. Blok 4 bit sola kaydırma işlemi için kullanıldığında seçme girişi yüksek seçilip kaydırma uzunluğu girişi de 4 seçilerek kaydırılmak istenen 32 bit uzunluklu veri girilir. Blok 4 bit sola kaydırılmak istenen sayının her bir bitini 4 defa bir yüksek anlamlı bite taşır ve en sağda kalan 4 boş bitleri de sıfır değeri ile doldurur. Yine 5 bit sağa kaydırma işleminde ise seçme girişi düşük seçilip kaydırma uzunluğu 5 seçilerek kaydırılmak istenen 32 bit uzunluklu veri girişe uygulanır. Blok 5 bit sağa kaydırılmak istenen sayının her bir bitini 5 defa bir düşük anlamlı bite kaydırır. Bu işlem sonucunda en sağda kalan 5 bite sıfır değeri atanır.

4) Özel Veya Bloğu: Şifreleme ve şifre çözme yapılarının ikisinde de özel veya bloğu bulunmaktadır. İki yapı

içerisinde de bir tur içinde iki adet özel veya bloğuna ihtiyaç duyulmaktadır. Şekil 7'daki yapıda da görüldüğü üzere tasarım aşamasında kullanılacak özel veya bloğu 32 bit uzunluklu üç adet veri girişi barındırmalıdır. Bu blok girişine uygulanan üç verinin özel veya işlemine girmesiyle 32 bit uzunluklu çıkış verisi üretmektedir.

5) Şifreleme Bloğu: Şifreleme ve şifre çözme donanımlarının alt donanım blokları tasarımları tamamlandıktan sonra TEA şifreleme ve TEA şifre çözme üst bloklarının tasarımına geçilmiştir. Belirlenen tasarım hedefleri göz önünde tutularak en uygun tasarımlar yapılmıştır.

Şifreleme donanımının mümkün olduğunca az alan kaplaması ve az güç tüketmesi gerekmektedir. Bu tasarım hedefi doğrultusunda TEA şifreleme donanımının saat işareti ile eş zamanlı olarak çalışmasına karar verilmiştir. Şekil 7'da şifreleme yapısının 32 turundan sadece 1 tanesi görülmektedir. Şifreleme kısmında önce bu bloktan 1 tane tasarlanarak daha sonra gerekli bağlantılar yapılarak şifrelenmek için girişe uygulanan verinin 32 defa aynı bloğa uygulanması yöntemi kullanılmıştır.

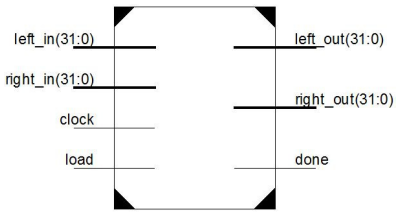
Şifre çözme donanımının olabildiğince hızlı çalışması gerekmektedir. Bu tasarım hedefi doğrultusunda şifre çözme bloğunun tamamen kombinezonsal olarak tasarlanmasına karar verilmiştir. Şekil 7'da görülen şekilden 32 adet arka arkaya bağlanarak oluşturulmuştur. Yani saat işaretinden bağımsız halde girişine şifresi çözülmek üzere verilen verinin şifresini çözerek çıkışa vermektedir.

Saat İşaretli Şifreleme Bloğu

Saat işaretine sahip TEA şifreleme bloğu tasarımda oldukça az alan kaplamakta ve buna bağlı olarak oldukça az güç tüketmektedir. Bu şifreleme donanımı FPGA üzerinde sadece 175 dilim kaplamaktadır. Bu tez aşamasında kullanılan başlangıç kitinin toplam 4656 dilime sahip olduğu düşünülürse 175 dilim kullanarak şifreleme donanımını gerçeklemek tasarım açısından oldukça başarılıdır.

Şifreleme bloğunun genel yapısı Şekil 11'te görüldüğü gibidir. Şifreleme bloğu donanımı 4 adet mantıksal kaydırma, 8 adet toplama ve 2 adet özel veya bloğu içermektedir. Verilog dili ile bu alt bloklar birbirlerine Şekil 7'da gösterildiği şekilde bağlanmıştır. Bu bağlama işlemleri yapılırken her bir bağlantıya birer kablo ya da kaydediciler atanarak bu kablolar ve kaydediciler yapı sağlanacak şekilde alt donanımların giriş ve çıkışlarına atanmışlardır. Şifreleme donanımının anahtar değeri içeride 128 bit uzunluklu bir kablo ile saklı tutulmuştur. Anahtar değerinin 32 bit uzunluklu 4 parçasının gerekli alt bloklara bağlantıları atama işlemleriyle sağlanmıştır. Şifreleme kısmında her bir döngüde farklı olmak üzere kullanılacak delta değişken değerleri ise hesabı fazladan alan yükü getirmemesi amacıyla Denklem'1'den Matlab programı yardımı kullanılarak hesaplanmışlar ve dizi olarak kodun içerisinde tutulmuşlardır.

Şifreleme donanımının yükleme girişi yüksek seviyede iken şifrelenecek olan 32 bit uzunluklu iki veri bloğun 32 bit uzunluklu iki girişine uygulanarak şifreleme işlemine ilk adım atılır. Bu arada saat işareti sürekli olarak donanıma uygulanmaktadır. Şifrelenecek olan sayılar donanım tarafından alındığında ve yükleme girişi yüksekte iken alınan



Şekil 11. Saat işaretli şifreleme bloğu.

sayılar donanım içerisinde iki tane 32 bit uzunluklu kaydediciye atılmaktadır. Bunun sebebi şifrelenecek sayıların saat ile eş zamanlı biçimde işlenmesi için değişken bir değere atama gerekliliğidir. Aynı zamanda daha önceden hesaplanan delta değişken değerleri de başlangıçta bir kaydediciye atılmıştır. Bu kaydedicilere değerler atandığında yükleme girişi düşük seviyeye getirilerek şifreleme işlemi başlatılmaktadır. TEA'nın yapısından da görüldüğü üzere şifreleme mekanizması 32 döngü sonunda tamamlanmaktadır. Döngü sayısını kontrol etmek için bloğun içerisine bir sayıcı eklenmiştir. Sayıcıya başlangıçta 1 değeri atanmakta ve her saat darbesinde sayıcı bir değer arttırılmaktadır. Sayıcı değişkeni 32 değerini aldığında donanımın 'done' çıkışı kendiliğinden yüksek değere çıkarak şifreleme işleminin bittiğini haber vermektedir.

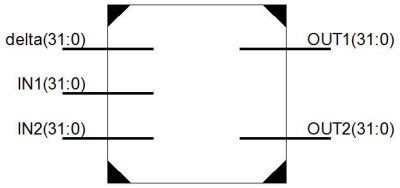
Tasarlanan bloğun ISE aracı yardımıyla yerleştirme ve hat çizimi sonrası benzetimi Şekil 21'da görülmektedir. Benzetim sonuçlarının doğruluğu şifreleme algoritmasının anlatıldığı kaynaklarda mevcuttur. Test aşamasında makalede de bahsedildiği gibi girişlerin her ikisine de $(00000000)_{16}$ verileri uygulanarak çıkışlardan $(9327C497)_{16}$ ve $(31B08BBE)_{16}$ şifreli verileri elde edilmiştir. Benzetim sonuçlarında da görüldüğü üzere donanım 32 saat darbesi sonunda doğru sonucu vermektedir. Doğru sonucu verdiğinde 'done' çıkışının seviyesi yükselmektedir.

Saat İşaretsiz Şifreleme Bloğu

Yazılım tasarımı esnasında Microblaze mikroişlemcisi ile saat işaretli şifreleme bloğu sağlıklı bir şekilde kontrol edilememesinden dolayı bir de saat işaretsiz bir başka deyişle tamamen kombinezonsal olarak şifreleme bloğu tasarlanmıştır. Tamamen kombinezonsal olarak tasarlanan şifreleme bloğu saat çevrimi olmayacağı 32 döngü beklemek zorunda kalmayarak tek seferde şifreleme yapacaktır. Bu da saat işaretli şifreleme bloğuna göre en az 2 kat daha hızlı bir sistemi ifade etmektedir. Kapladığı alan konusunda kaybettiği bu avantajı en az iki kat daha hızlı bir sistem olarak hızdan kazanmaktadır.

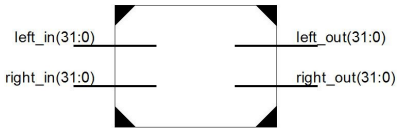
Bu şifreleme bloğu tasarımında ilk şifrelemenin bir döngüsü Şekil 12'daki gibi tasarlanmıştır. Şifreleme için gerekli olan 128 bit uzunluklu anahtar bu blok içerisinde bir kablo değişkeniyle tutulmaktadır. Bloğun üç girişi ve iki çıkışı mevcuttur. Bunlardan ikisi her bir döngüye girmekte olan verilerin girişidir. Daha önce hesaplanmış olan delta değişkeninin değerleri de üst blokta tutularak her bir döngüye kendi delta değeri bu delta girişi vasıtasıyla ulaştırılacaktır.

Üst blok yani saat işaretsiz şifreleme bloğu oluşturulurken Şekil 12'da görülmekte olan bloklardan 32 tanesi bir alt



Şekil 12. Şifreleme bloğunun bir döngüsü.

bloğun çıkışı diğerinin girişi olacak şekilde en üst blokta art arda bağlanmıştır. Yine en üst blokta bu 32 adet alt bloğun delta girişlerine daha önce hesaplanıp tutulmuş delta değerleri verilmiştir. En üst bloğun yani şifreleme bloğunun genel yapısı Şekil 13'de görüldüğü gibidir.

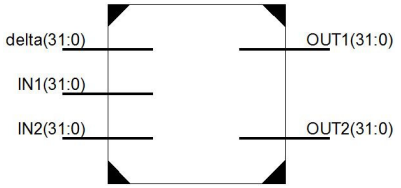


Şekil 13. Saat işaretsiz şifreleme bloğu.

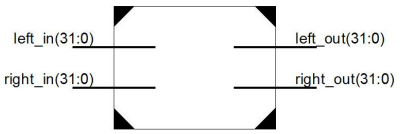
Blok girişine verilen şifresiz veriler saat işareti beklemeksizin hesaplanarak çıkışa verilmektedirler. Şifreleme bloğu FPGA üzerinde toplam 2048 adet dilim kaplamaktadır. Hat gecikmeleri ve mantık bloklarının gecikmelerinin toplamı yaklaşık olarak $290 * 10^{-9}$ saniyedir. Yani Bloğun girişine şifrelenmek üzere veri girdiğimizde blok bütün işlerini $290 * 10^{-9}$ saniyede bitirerek şifreli veriyi çıkışa vermektedir. Şifreleme bloğunun hat bağlantıları ve yerleştirme işlemi tamamlandıktan sonraki benzetim sonuçları Şekil 22'da görülmektedir. Girişlerden çıkışlara ne kadar gecikme olduğu Şekil 22'da açıkça görülmektedir.

6) Şifre Çözme Bloğu: Etiket kısmına yerleştirilecek olan şifreleme bloğunun tasarımı tamamlandıktan sonra okuyucu kısmı için de bu şifrelenmiş veriyi çözmek için şifre çözme bloğuna ihtiyaç duyulmaktadır. Şifre çözme donanımı için tasarım hedefi olarak çok hızlı bir sistem olması gerekmektedir. Bu doğrultuda şifre çözme bloğunun tasarımının saat işaretinde bağımsız tamamen kombinezonsal olmasına karar verilmiştir. Şifre çözme bloğunun tasarımı sırasında alt bloklar olarak toplama, çıkarma, mantıksal kaydırma ve özel veya blokları kullanılmıştır. Şifre çözme bloğu da şifreleme bloğunda olduğu gibi Şekil 7'daki turun 32 defa döngüye girmesiyle tamamlanmaktadır. Şifreleme tarafından farklı olarak anahtarların ters dönmesi ve ana kollardaki toplama bloklarının yerlerini çıkarma bloklarının almasıdır. Saat işaretsiz şifreleme bloğu tasarımında da olduğu gibi ilk olarak alt bloklar kullanılarak şifre çözme yapısının tek bir döngüsü tasarlanmıştır. Daha sonra bu tek bir döngünün öncekini çıkışı sonrakinin girişi olacak şekilde 32 adet peş peşe bağlanmıştır. Şekil 14de tek bir turun genel yapısı görülmektedir.

Bu bloğun içerisinde 4 adet mantıksal kaydırma, 6 adet toplama, 2 adet çıkarma ve 2 adet özel veya bloğu uygun şekillerde ara kablolarla birbirlerine bağlanmışlardır. Anahtar değeri bu blok içerisinde tutulmuştur. Okuyucu tasarımının tamamlandığı en üst bloğun içerisinde ise bu bloklardan 32 tanesi yan yana üretilerek uygun şekilde bağlama işlemleri yapılmıştır. En üst bloğun genel yapısı Şekil 15'da görülmektedir. Delta değişkeninin değerleri ise bir dizi yardımıyla en üst blokta tutularak her bir alt bloğa kendi kullanacağı delta değerleri Şekil 14'deki delta girişinden verilmektedir.



Şekil 14. Şifre çözme bloğunun bir döngüsü.



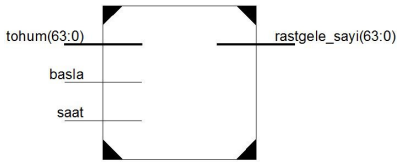
Şekil 15. Şifre çözme bloğu.

Şifre çözme bloğunun FPGA üzerine yüklenmesi benzetim sonuçları Şekil 23'de görülmektedir. Şifreleme bloğu için benzetim sonuçlarında girişlere $(00000000)_{16}$ değeri verildiğinde çıkışlarda $(9327C497)_{16}$ ve $(31B08BBE)_{16}$ şifreli metinleri elde edilmişti. Şifre çözme bloğundan beklediğimiz ise bu şifreleri verileri girişlere verdiğimiz zaman çıkışlardan şifresi çözülmüş verileri yani (00000000)16 elde etmemiz gerekmektedir. Şekil 23'de de görüldüğü gibi yapılan şifre çözme bloğu tasarımı tamamen doğru olarak çalışmaktadır. Şifre çözme donanımı FPGA üzerinde 2048 dilim kaplamaktadır. Aynı zamanda şifre çözme bloğu $296 * 10^{-9}$ saniye gibi çok kısa bir sürede şifre çözme işlemini tamamlamaktadır. Buradan da görüleceği üzere tasarımın başında hedeflenen okuyucu kısmı için çok hızlı şifre çözme donanımı tasarımı başarıyla gerçekleştirilmiştir.

7) Rastgele Sayı Üreteci Bloğu: Doğrulama mekanizması içerisinde okuyucu etiketin kimliğini tanımlama yapabilmesi için rastgele sayı üreterek etikete göndermesi gereklidir. Bundan dolayı okuyucu kısmı için rastgele sayı üreteci bloğu tasarlanmıştır. Tasarımın genel yapısı Şekil 16'da görülmektedir. Rastgele sayı üretecinin çalışma yapısı, basla girişi alçak seviyede iken bloğa 64 bit uzunluklu bir tohum değeri girilmektedir. Tohum değeri girildikten sonra her saat darbesinde bitler bir yüksek anlamlı bite atanır. Bunun sonucunda en sağ tarafta kalan en anlamsız bite ise 64. 62. 51. ve 50. bitlerin 'özel veya değil' işlemine sokulmasıyla elde edilen bit atanır. Tüm bu işlemler sonucunda rastgele sayı üreteci bloğu her saat darbesinde okuyucu tarafından kullanılmak amacıyla çıkışından 64 bit uzunluklu sözde-rastgele sayı üretir.

B. Yazılım Tasarımı

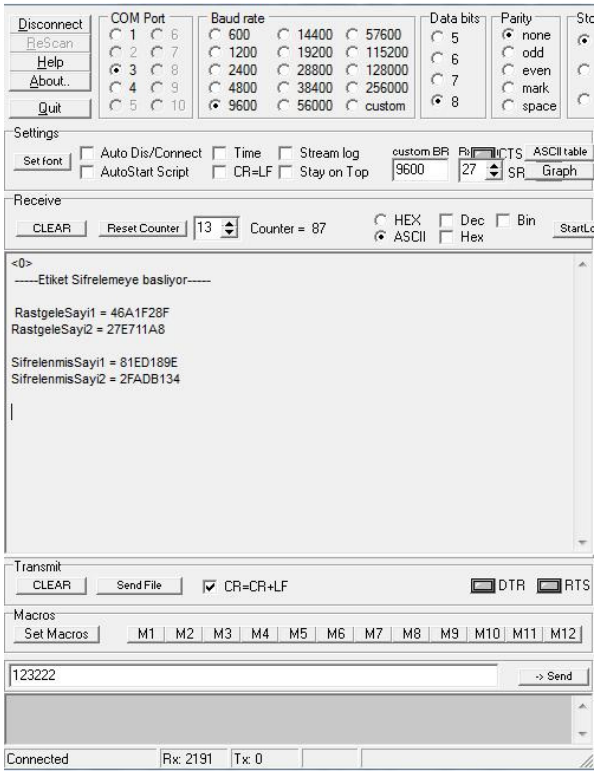
Etiket ve okuyucu tasarımı için gerekli olan şifreleme, şifre çözme ve rastgele sayı üreteci donanımları tasarımları yapıldıktan sonra bu donanımları ve tüm sistemi kontrol



Şekil 16. Rastgele sayı üreteci bloğu.

için FPGA üzerindeki Microblaze mikroişlemcisi kullanılmıştır. Etiket ve okuyucu kısımları için iki ayrı FPGA kartı kullanılmıştır. Mikroişlemci merkezli sistem tasarımı yapılırken öncelikle ISE programı aracılığıyla Microblaze temelli bir proje oluşturulmuştur. Daha sonra EDK programıyla donanımlar ile Microblaze arasındaki bağlantılar yapılmıştır. Ayrıca proje için gerekli olan seri Evrensel Eşzamansız Alıcı/Verici (Universal Asynchronous Receiver/Transmitter - UART) IP'leri de Microblaze ile yönetilebilir hale getirilmiştir. Son olarak tüm donanım bağlantıları tamamlandıktan sonra SDK programında yazılım tasarımı yapılarak proje tamamlanmıştır.

1) Etiket Kısmı: Etiket kısmı için gereken donanım parçası şifreleme bloğu olduğu için etiket kısmındaki Microblaze mikroişlemcisi şifreleme bloğuna erişmeli, onu kontrol etmelidir. Bu tasarım hedefi doğrultusunda projeye başlarken ISE programından projeye Microblaze mikroişlemcisi eklenmiştir. Projenin donanım Microblaze bağlantılarının yapmak üzere EDK programına geçilmiştir. EDK programı yardımıyla tasarımın gerçekleneceği kit seçilerek, sistemin saat işareti frekansı 50 Megahertz olarak belirlenmiştir. Ayrıca Microblaze'in hafızası 32 Kilobayt olarak seçilmiştir. Proje oluşturulduktan sonra şifreleme donanımı için kullanıcı IP'si oluşturur. Kullanıcı IP'si oluşturulurken Microblaze ile şifreleme donanımı arasındaki veri alış verişi sağlamak amacıyla 16 adet 32 bit uzunluklu kaydediciler kullanılmıştır. IP oluşturulduktan sonra şifreleme kısmı için yazılan donanım bloğunun en üst blok kodu EDK'nın kullanıcı IP'si için oluşturmuş olduğu en üst verilog kodunda çağrılmıştır. Kullanıcı IP'si için Verilog koduna gerekli eklemeler yapıldıktan sonra bu eklemelerin güncellenmesi için kullanıcı IP'si yeniden projeye eklenmiştir. IP'si oluşturulan şifreleme donanımının Microblaze ile olan bağlantısı PLB (Processor Local Bus) ile sağlanmıştır. Tüm bağlantıları da yapılan şifreleme donanımı için son olarak Microblaze hafızasında otomatik olarak adresi oluşturulmuştur. Microblaze ile şifreleme donanımı tamamen birleştirildikten sonra yazılım tasarımına geçmek üzere EDK projesi ISE aracılığıyla sentez ve gerçekleme aşamalarına girdirilmektedir. Bu aşamalar da tamamlandıktan sonra Microblaze'li donanım projesi yazılım tasarımı yapılmak üzere SDK programına gönderilir. Microblaze mikroişlemcisinin kontrol mekanizması SDK ara yüzünde C dili ile programlanmıştır. Yazılım aşamasında şifreleme donanımına ulaşmak için kullanıcı IP'sinin üst bloğunda Microblaze ile iletişim oluşturmak için aralarına koyulan kaydedicilerin adresleri kullanılmaktadır. Okuyucu tarafından etikete gelen rastgele sayının etiket tarafından şifrelenebilmesi için alınan sayı, şifreleme donanımının girişine bağlanan kaydediciye gönderilerek, çıkışına bağlanan kaydediciden de şifrelenmiş olan sayı okunmuştur. Oluşturulan sistemin FPGA üzerinde gerçekleme aşamalarını görmek için FPGA seri port aracılığıyla bilgisayara bağlanarak, FPGA'nın üzerinde akmakta olan işlemler bilgisayar ekranında gözlemlenmiştir. Okuyucu tarafından gönderilen ve etiket tarafından alınıp şifrelenen verinin FPGA üzerinde gerçekleştirilmesi Şekil 17'de görülmektedir.

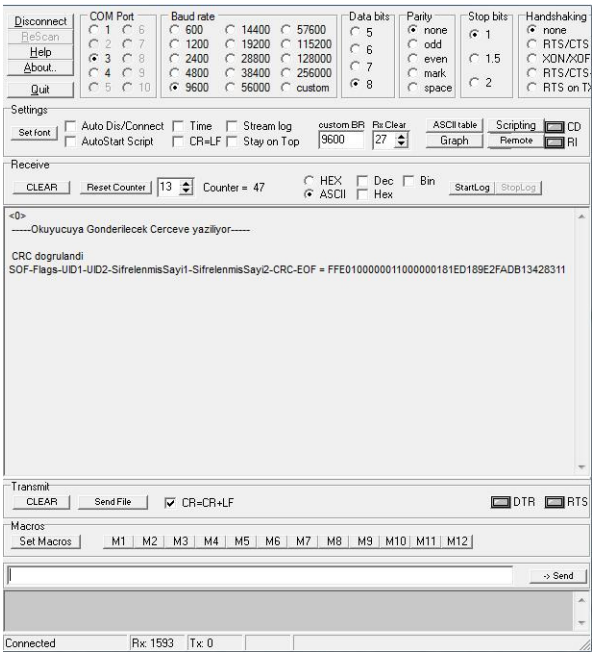


Şekil 17. Etiketin FPGA üzerinde gerçeklenmesine ilişkin ekran çıktısı.

Etiket tarafından oluşturulan şifrelenmiş veri okuyucu kısmına gönderilmesi için cevap çerçevesi içerisinde gönderilmelidir. Çerçeve içerisindeki SOF değeri $(FF)_{16}$, Bayraklar değeri $(E0)_{16}$, UID değeri $(1000000110000001)_{16}$ ve EOF değeri ise $(11)_{16}$ olarak belirlenmiştir. Okuyucu kısmına gönderilecek verinin hatalı olup olmadığını kontrol etmek amacıyla çerçeveye eklenmesi gereken CRC değeri yazılım içerisinde hesaplanarak $(0283)_{16}$ olarak hesaplanarak çerçeveye eklenmiştir. Okuyucuya gönderilen veri çerçevesinin değerleri Şekil 18'de gösterilmektedir.

Protokolün başlangıcında etiket Microblaze içinde sakladığı sayıcı değerini, şifrelenmiş sayıyı gönderdiği gibi aynı çerçevede göndermektedir. Etiketin göndermiş olduğu sayıcıyı alıp kendi sayıcısı ile kontrol eden okuyucu etikete bu rastgele sayısını göndermektedir. Daha sonra Şekil 18'de ki çerçeve okuyucuya gönderilerek etiket görevini tamamlamıştır.

2) Okuyucu Kısmı: Okuyucu kısmı için tasarlanan şifre çözme ve rastgele sayı üreteci donanımları Microblaze mikroişlemcisi ile yönetilmesi gerekmektedir. Bu doğrultuda okuyucu kısmını yönetmek için ISE programı kullanılarak Microblaze temelli sistem oluşturuldu. Etiket kısmında

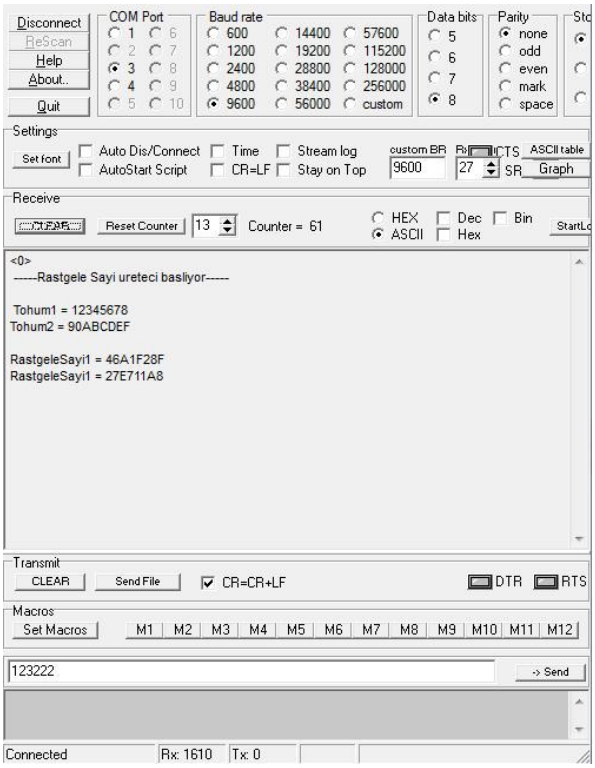


Şekil 18. Okuyucuya gönderilen çerçeveye ilişkin ekran çıktısı.

olduğu gibi sistem birimlerini ve bağlantılarını oluşturulan için EDK programı kullanılmıştır. EDK programında, daha önce tasarlanmış olan rastgele sayı üreteci ve şifre çözme bloklarının Microblaze ile bağlantısını yapmak için kullanıcı IP'si oluşturulmuştur. İki donanım bloğu aynı IP içerisinde yönetilmiştir. Bu iki donanım bloğunu yönetmek için Microblaze ile donanımlar arasına IP içerisinde donanımlara erişmek amacıyla 32 bit uzunluklu kaydediciler konulmuştur. Donanımlara erişmek için kullanılacak olan kaydedici sayısı donanımları giriş ve çıkışlarının toplamı kadar olmalıdır. Rastgele sayı üreteci donanımına erişmek için 5 adet ve şifre çözme bloğuna erişmek için ise 4 adet kaydedici kullanılmıştır. IP'si oluşturulan kullanıcı donanımlarının Microblaze'in veri yoluna bağlanması PLB ile sağlanmıştır. Bu haliyle Microblaze rastgele sayı üretecine ve şifre çözme bloğuna kendi komutlarıyla ulaşabilir hale getirilmiştir.

Sisteme eklenen Microblaze ile donanım tasarımı tamamlanan okuyucunun yazılım tasarımını yapmak için ISE aracılığıyla sentezleme ve gerçekleme işlemleri yapılarak SDK programına geçilmiştir. Okuyucu kısmının yazılım tasarımı da C dili ile yapılmıştır. Okuyucunun görevi doğrultusunda öncelikli olarak etiketten aldığı sayıcı bilgisini kontrol eder, eğer bu veri kendi içerisinde tutmuş olduğu sayıcı ile aynı değilse protokol gerçeklenmeden etiketle iletişimi kesmektedir. Eğer sayıcı bilgisi doğru ise protokolün ikinci aşaması olan rastgele sayı üretecine tohum değeri gönderir ve oradan bir rastgele sayı alarak uygun çerçeve ile etiket kısmına göndermektedir. Daha sonra etiket kısmından gelen şifrelenmiş sayının şifresini çözmek üzere şifre çözme donanımına gönderir. Bunun sonucunda şifresi çözülen sayıyı göndermiş olduğu rastgele sayı ile karşılaştırarak eğer doğruysa protokolü başarılı bir şekilde tamamlar.

Microblaze'in rastgele sayı üretecine göndermiş olduğu tohum değeri ve bunun sonucunda elde ettiği rastgele sayıya ilişkin FPGA üzerinde gerçekleme görüntüleri Şekil 19'de görülmektedir. Bu değerin Şekil 17'de görülen etikete giden veri olduğu bilinmektedir.

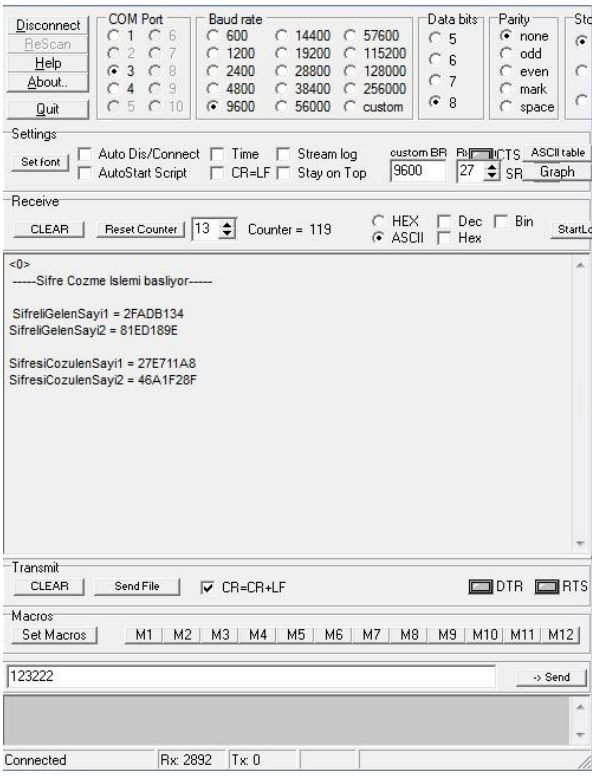


Şekil 19. Rastgele sayı üretecinin Microblaze ile FPGA üzerinde gerçeklenmesine ilişkin ekran çıktısı.

Rastgele sayının üretilip aynı sayının etiket tarafından şifrelenmesi Şekil 17'de görülmüştü. Etiket tarafından şifrelenen verilerin okuyucu tarafına gönderilecek şifresinin çözülmesine ilişkin ekran görüntüsü Şekil 20'te görülmektedir. Şifresi çözülmüş olan rastgele sayı ile okuyucu tarafından üretilip etikete gönderilmiş olan rastgele sayının eşit olduğu Şekil 19 ve Şekil 20 karşılaştırılarak çok net bir biçimde görülmektedir. Bu iki verinin aynı olması okuyucunun ve etiketin protokolü başarılı bir biçimde tamamladığını göstermektedir.

V. Sonuçlar

Bu çalışmada donanım ve yazılım içeren güvenli bir RFID sistem tasarlanmıştır. Çalışmada bahsedilen protokol ilk defa FPGA üzerinde gerçeklenerek donanım ve yazılım ortak çalışması gösterilmiştir. Etiket ve okuyucu kısımları tasarımlarının sorunsuz şekilde çalışır halde olduğu UART aracılığıyla bilgisayarla haberleştirilerek kanıtlanmıştır. Protokolde kimlik doğrulama mekanizmasını sağlamak üzere şifreleme algoritması olarak ilk defa TEA kullanılmıştır. TEA gerek yer bakımından gerek hız bakımından daha önce gerçeklenmiş olan algoritmalara kıyasla, RFID sistemlere daha verimli çalışma imkanı sağlamıştır. TEA şifreleme algoritmasının saat işaretli olarak FPGA üzerinde sadece 175 dilim işgal edilerek tasarlanması başarılmıştır.



Şekil 20. Okuyucuya ait şifre çözme bloğunun FPGA üzerinde gerçeklenmesine ilişkin ekran çıktısı.

Ayrıca okuyucu kısmı için tasarlanan şifre çözme bloğunun $290 * 10^{-9}$ saniye gibi kısa bir sürede şifre çözme işlemini yapması sağlanmıştır.

Çalışma neticesinde TEA şifreleme algoritması kullanılarak daha hızlı ve daha az yer kaplayan RFID sistemlerin tasarlanması mümkün hale getirilmiştir. Gelecek çalışmalarda daha gelişmiş mikroişlemciler ve daha efektif şifre algoritmaları kullanılarak daha verimli sistemlerin tasarlanabileceği tespit edilmiştir.

Kaynaklar

[1] K. Finkenzeller, *RFID-Handbook*, 2nd ed. John Wiley & Sons, Ltd., 2003.

[2] M. Feldhofer, "An authentication protocol in a security layer for RFID smart tags," in *Proceedings of The 12th IEEE Mediterranean Electrotechnical Conference (MELECON)*, vol. 2. Dubrovnik, Croatia: IEEE, May 2004, pp. 759 – 762.

[3] A. Kavas, "Radyo frekans tanımlama sistemleri," *Elektrik Mühendisliği*, vol. 430, pp. 74 – 80, Nisan 2007.

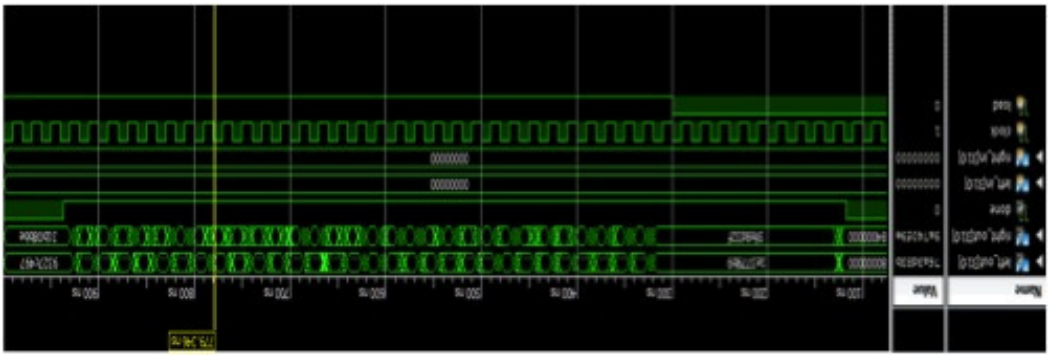
[4] H. Lehpamer, *RFID Design Principles*. New York, NY, USA: Artech House, 2008.

[5] I. I. O. for Standardization), *ISO/IEC 18000-3:2010 Information technology – Radio frequency identification for item management – Part 3: Parameters for air interface communications at 13,56 MHz*, 2010.

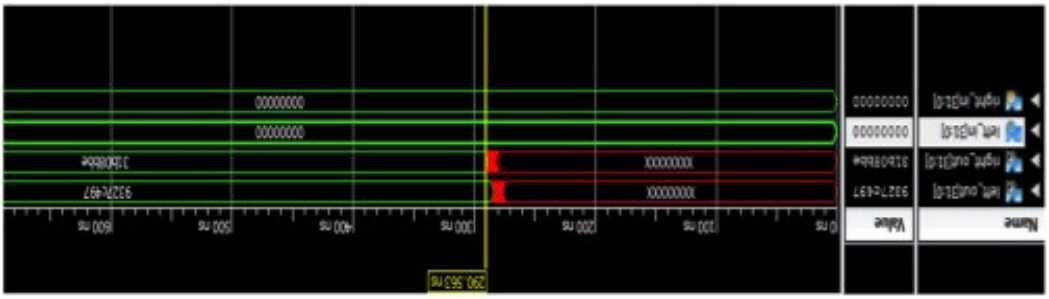
[6] D. J. Wheeler and R. M. Needham, "TEA, a tiny encryption algorithm," in *Proceedings of the Second International Workshop Fast Software Encryption (FSE)*, ser. Lecture Notes in Computer Science. Leuven, Belgium: Springer, 1994, pp. 363 – 366.

[7] V. R. Andem, "A cryptanalysis of the tiny encryption algorithm," Master's thesis, The University of Alabama, Alabama, USA, March 2003.

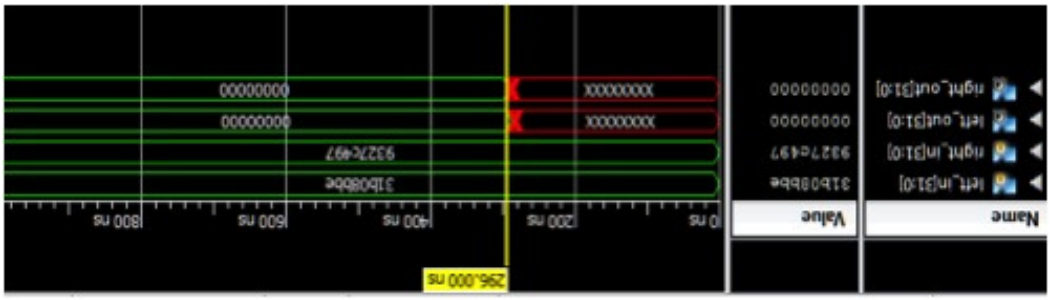
[8] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1997.



Şekil 21. Saat işaretli şifreleme bloğu.



Şekil 22. Saat işaretli şifreleme bloğu.



Şekil 23. Saat işaretli şifreleme bloğu.

FPGA Üzerinde IPv4 Ethernet Haberleşme Uygulaması

Mehmet Salih Ocak, Evren Cesur, Nerhun Yıldız, Vedat Tavşanoğlu
Yıldız Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Esenler, 34220, İstanbul
e-posta: mehmetsalih.ocak@gmail.com, ecesur@yildiz.edu.tr

Özetçe

Optik fiber kabloların hızlı gelişimi ile birlikte 10 Gbps mertebelerine çıkan iletim hızları uç birimlerde verinin işlenmesinin çok daha hızlı olması ihtiyacını doğurmuştur. Uç birimlerde veri işleme fonksiyonları büyük oranda genel amaçlı mikroişlemciler ile gerçeklenmektedir. Fakat iletim tarafındaki hızın giderek artması bazı durumlarda işlemci kapasitelerinin yetersiz kalmasına ve veri iletiminde gecikmelere sebep olmaktadır. Bu noktada daha özelleşmiş donanımlara ihtiyaç duyulur. Bu özel donanımlardan biri ise alanda programlanabilir kapı dizileri(Field Programmable Gate Arrays, FPGA) dir Bu çalışma kapsamında internet üzerinden paket haberleşmesi yapabilen bir sistemin FPGA üzerinde tasarımı ve gerçeklenmesi amaçlanmıştır.

1. Giriş

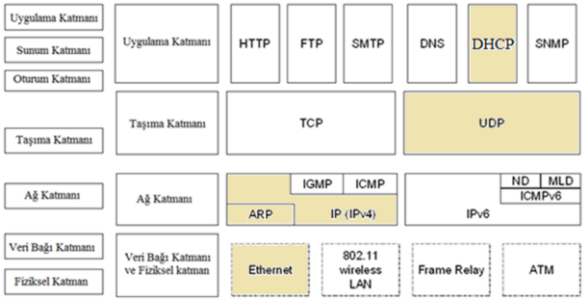
Paralel işlem kapasitesi ve esnek tasarım kabiliyeti ile çok yüksek veri işleme hızlarına ulaşabilen FPGA üzerinde şu ana kadar bir çok ağ ara yüz sistemi gerçeklenmiştir. Fakat bu gerçekleştirilen sistemlerin genel özelliği, uygulama katmanındaki fonksiyonları ya bir işlemciyle veya bir sürücü aracılığıyla genel amaçlı bir bilgisayar üzerinde gerçekleştirmesidir. Yapılan bu çalışmada ise sistem, gerekli olan tüm fonksiyonları başka hiçbir ek donanıma ihtiyaç duymadan tek bir FPGA tümdevresi üzerinde gerçekleştirecek şekilde tasarlanmıştır.

TCP/IP referans modelinde tam bir hiyerarşik yapı vardır. Hiyerarşinin en üstünde kullanıcıların çalıştırdığı uygulama programları ve en altta ise verinin bit düzeyinde aktarılması vardır. Ara katmanlar bu iki katman arasında gerekli uyarlamaları içerir. Her katman bir üst katmana hizmet sunarken bir alt katmandan kendisi için hizmet ister.
Bu çalışmada TCP/IP referans modelinin tüm katmanlarının fonksiyonlarını gerçekleştirebilecek bir sistem tasarımı yapılmıştır. Sistem blokları temelde veri paketleme ve ayırma esasına dayalı çalışır. Her katmandaki blokların özel görevleri vardır. Alıcı kısımlar veri filtreleme görevlerini de üstlenirken gönderici kısımlarda iletim hattına gönderilecek gerekli cevapları, paketleri oluşturma görevini üstlenir. Gerçeklenen sistem gerçek zamanlı uygulamalara imkan vermek amacıyla gelen paketleri filtreleme işlemini paketin tamamının gelmesini beklemeden hızlandırarak yapmaktadır, sistem ileriye yönelik geliştirilmeye açık olması için bloklar arasında ortak bir ara yüz planlanmış ve ek bloklar eklenmesi kolaylaştırılmıştır.
Gerçekleme sırasında VHDL(Very High Speed Integrated Circuit Hardware Description Language) dili kullanılmıştır. Bazı blokların çalışması Modelsim veya Isim yazılımlarında simülasyonla gözlenirken çoğunun çalışması Digilent firmasına ait spartan3E kiti[1] üzerinde gerçeklenmiştir. Ethernet fiziksel ara yüz işlemlerini gerçekleştirmesi için SMSC firmasının 83C185 tümdevresi[2] kullanılmıştır.

.

2. Sistem Tasarımı

TCP/IP referans modeli [3] 5 katmandan oluşmaktadır. Bu katmanlar fiziksel, veri bağı, internet, taşıma ve uygulama katmanları olarak adlandırılır. Her bir katmanda farklı protokoller çalışmaktadır. Her bir katmanda bulunan protokoller ve bu çalışmada hangilerinin kullanıldığı Şekil 1'de görülmektedir[4].



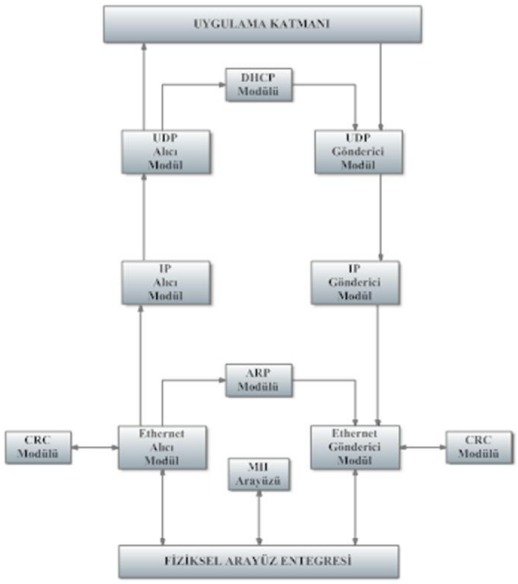
Şekil 1: TCP/IP referans modeli ve kullanılan protokoller

Fiziksel katmanda hatta gönderilecek farksal işaretleri oluşturma işini SMSC firmasına ait 83C185 tümdevresi gerçekleştirmektedir. Veri bağı katmanında ise kablolu haberleşme kullanıldığından Ethernet protokolü seçilmiştir. Ethernet protokolü fiziksel hattan gelen verileri paketlerden ayırma ve MAC adresi filtreleme görevini yerine getirerek bir üst katmana filtrelenmiş IP paketini göndermektedir[5].

İnternet katmanında IPv4 [6] kullanılmıştır. Ayrıca IP(Internet Protocol) [6] ve MAC(Media Access Control) [7] adresi dönüşümlerini yapması için bir de ARP(Address Resolution Protocol) [8] bloğu bulunacaktır. ARP protokolü TCP/IP referans modeline göre internet katmanında çalışan bir protokoldür. Ağa gelen pakette bulunan MAC adresinin hangi alt IP ye ait olduğu bilinmelidir çünkü gelen paketlerde sadece ağın IP si bulunmaktadır cihazın ise MAC adresi bulunmaktadır. Bu sorunun çözümü ARP protokolü ile sağlanmaktadır. Bu protokolün görevlerini de sistemde bulunan olan ARP bloğu gerçekleştirecektir.

Uygulama katmanında en önemli iki protokol TCP(Transmission Control Protocol) [9] ve UDP(User Datagram Protocol) [10] üzerinde araştırma yapılıp UDP kullanılmasına karar kılınmıştır. TCP bağlantı tabanlı ve güvenlikli bir protokol olup çok fazla ek kontrol paketleri ve ekstradan kontrol bitleri içermektedir. UDP ise bağlantısız olup ekstra güvenlik bitleri ve paketleri içermemektedir. Bu noktada güvenli iletim ile hızlı iletim arasında bir tercih yapılması gerekmektedir. Hem sistem tasarımında sağladığı kolaylık açısından hem de sistemin ileriye yönelik video aktarımı planları olduğundan UDP protokolünün kullanımına karar verilmiştir. Çünkü video paketleri aktarılırken arada kaybolacak küçük paketler görüntüde gözle görülür bir bozukluk oluşturmaz, bu da bizim güvenli iletimden fedakarlık edebilmemizi sağlar.

Sistemin sabit bir IP si olmadığı için herhangi bir ağa bağlandığında yeniden IP alması gerekmektedir. Bu görevi DHCP(Dynamic Host Configuration Protocol) [11] protokolü yerine getirecektir. Tasarlanacak bu blok gerekli DHCP paket trafiğini yöneterek sistemin başlangıç konfigürasyonunu gerçekleştirecektir. Tasarlanan sistemin bir blok diyagramı Şekil 2'de görülmektedir.



Şekil 2: Sistemin blok diyagramı

3. Sistemin Uygulaması

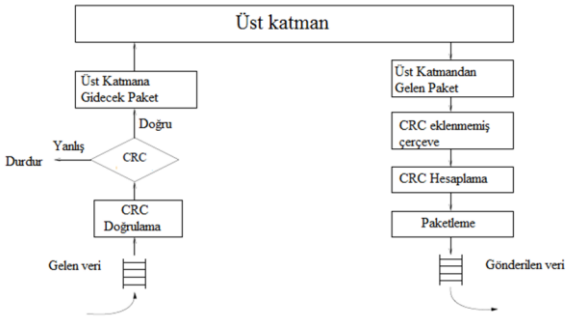
Sistem genel bakışta 2 ayrı kısımdan oluşmaktadır. Bunlar; FPGA içi ve FPGA dışı birimlerdir. FPGA dışındaki fonksiyonları SMSC firmasına ait 83C185 tümdevresi gerçekleştirirken, FPGA içerisinde 5 katmandaki protokollerin tümünü gerçekleyen bloklar ve fiziksel tümdevrenin konfigürasyon ara yüzünü oluşturan bir blok bulunmaktadır.

3.1 Fiziksel Arayüz Tümdevresi

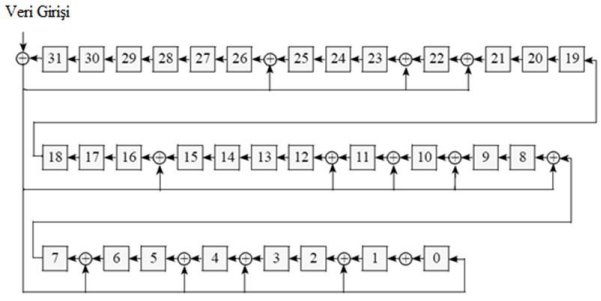
SMSC 83C185 tümdevresi FPGA dışında bulunan tek birimdir. FPGA içerisinde gerçeklenen MII(Media Independent Interface) arayüzü ile gerekli konfigürasyon bilgilerini aldıktan sonra çalışmaya başlar[2]. Dış dünyadan gelen sinyalleri 4 bitlik parçalar halinde Ethernet alıcı bloğa gönderir. Ethernet gönderici bloktan kendine 4 bitlik parçalar halinde gelen verileri ise MLT-3 kodlama yaparak dış dünyaya gönderir. Tümdevre 10 Base-T ve 100 Base-TX veri hızı modlarını desteklemektedir. 4 bitlik bir veri yolu vardır, özel durumlarda 5 bite çıkarılabilmektedir. Bu projede tümdevre 100 Base-TX ile çalışmaktadır ve bu modda tüm bu gönderme işlemleri 25 MHz ile 4 bit olarak gönderilmektedir. Böylece 100Mbps lik bir veri aktarım hızı elde edilir.

3.2 CRC Bloğu

Sistemde hata algılamak için Ethernet protokolünde standart hata algılama algoritması olarak kullanılan CRC-32 kullanılmıştır. Sistemde bulunan CRC bloğunun çalışması Şekil 3'te açıklanmıştır[12]. Algoritmanın FPGA için devre karşılığını polinomal yaklaşım ile elde etmek mümkündür. Gerçeklenen devre yapısı Şekil 3'teki gibidir[13]. Şekilde + işaretleri mantıksal xor işlemini simgelerken kare sembolleri bir bitlik birer yazmaç simgelemektedir. İşlem sonunda yazmaçlardaki değer verinin CRC-32 karşılığını verecektir.



Şekil 3: CRC bloğunun çalışması

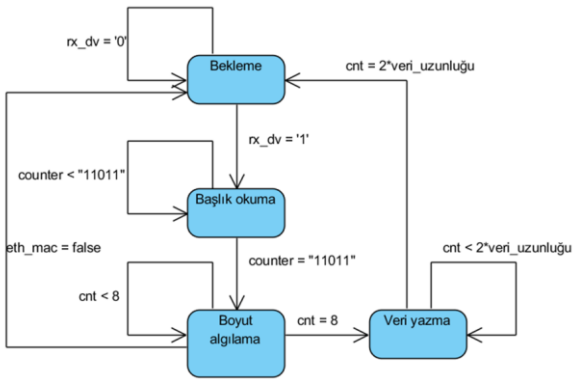


Şekil 4: CRC-32 algoritmasının devre karşılığı

3.3 Ethernet Alıcı Blok

Fizikse arayüzden gelen 4 bitlik parçaları birleştiren ve üst katmana ileten kısımdır. Bunu yaparken Ethernet başlığını veriden ayırır, gelen paket boyutunu hesaplar, MAC adresi filtrelemesini gerçekler. Eğer paket yayım(broadcast) veya kendi MAC adresine gelmemişse paketi durdurur üst katmana geçmesini engeller. Ayrıca gelen paketin IP yada ARP paketi olmasının ayrımı burada yapılır ve ona göre hangi üst katmana göndereceğine karar verir. Bu blokta durum makineleri kullanılmıştır. Temelde 4

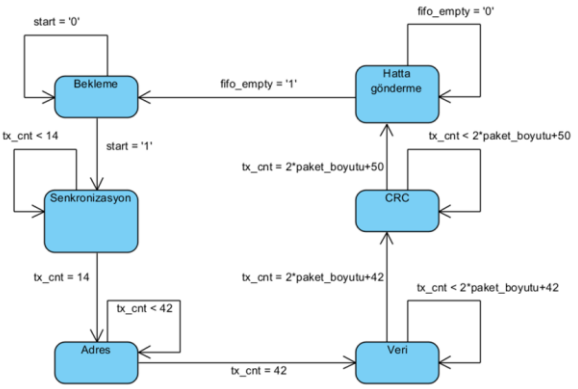
ayrı durum ile çalışmaktadır. Ethernet alıcı bloğun durum diyagramı Şekil 5'te görülmektedir.



Şekil 5: Ethernet alıcı blok durum diyagramı

3.4 Ethernet Gönderici Blok

IP bloğundan gelen datagramları(IP katmanındaki paketleri verilen isim) paketleme görevini üstlenir, diğer bloklardan gelen uzunluk ve adres bilgilerine göre paketleme işlemini yapar, bunu yaparken eş zamanlı olarak CRC değerini hesaplayarak paketin sonuna ekler. Bu blok bekleme, senkronizasyon, adres, veri, CRC ve hatta gönderme durumu olmak üzere 6 ayrı durum ile çalışmaktadır. Ethernet gönderici bloğun durum diyagramı Şekil 6'da görülmektedir.



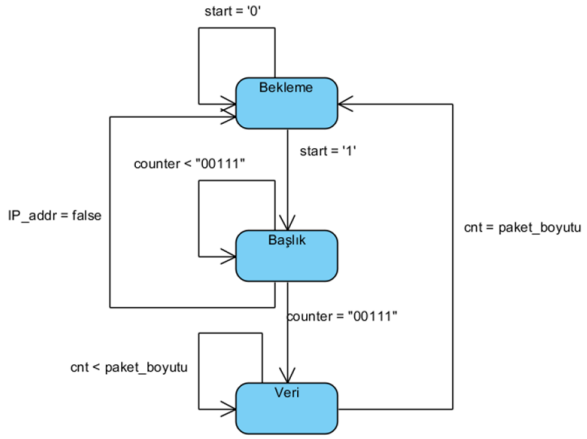
Şekil 6: Ethernet gönderici blok durum diyagramı

3.5 ARP Bloğu

Bu blok sadece sisteme ARP paketi geldiği zaman aktif olur. Sistem ARP paketi geldiğini ethernet alıcı blokta belirler ve ARP bloğuna bildirir. Blok kendi hafıza birimine veri yazılmaya başladığı anda okumaya başlar. Kendisine ARP isteği gönderen MAC adresine gönderilmek üzere kendi MAC adresini içeren paketi ethernet gönderici bloğun hafıza birimine yazar. Hafıza birimine yazılan paketi hatta göndermek üzere ethernet gönderici blok çalışmaya başlar.

3.6 IP Alıcı Blok

Bu blokta gelen IP paketlerinin başlık ve veri kısımları birbirinden ayrılarak bir üst katmana gitmesi gereken paket oluşturulur. Bu blok 3 temel durum ile çalışmaktadır. IP alıcı bloğun durum diyagramı Şekil 7'de görülmektedir.

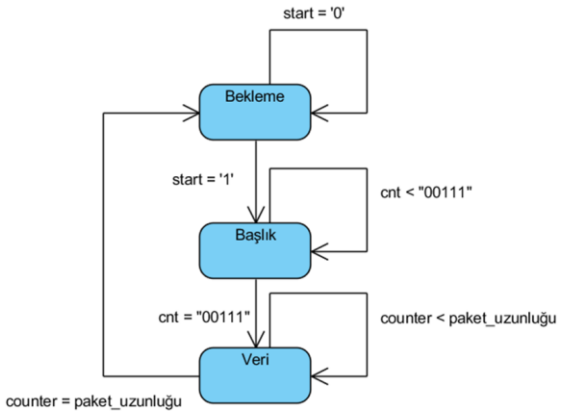


Şekil 7: IP alıcı blok durum diyagramı

3.7 IP Gönderici Blok

Uygulama katmanından almış olduğu IP verilerini kullanarak üst katmandan gelen paketleri IP datagram oluşturacak şekilde paketleme görevini üstlenir. IP datagram haline getirdiği veriyi ethernet gönderici bloğa gönderir ve hatta gönderilmesi için başlama sinyalini verir.

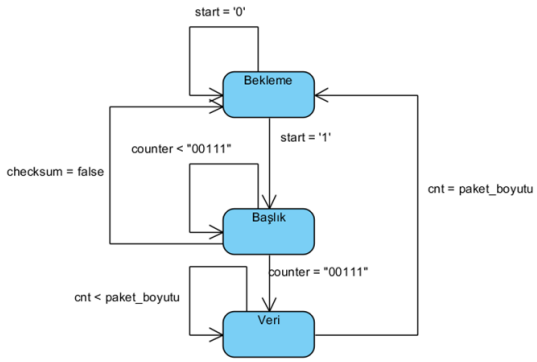
Başlama sinyalini alan blok bekleme durumundan çıkıp başlık bilgilerini ethernet gönderici bloğun hafıza birimine yazar ve ardından kendi hafıza biriminden okuduğu verileri başlığın arkasına ekler. IP gönderici bloğun durum diyagramı Şekil 8'de gösterilmiştir.



Şekil 8: IP gönderici blok durum diyagramı

3.8 UDP Alıcı Blok

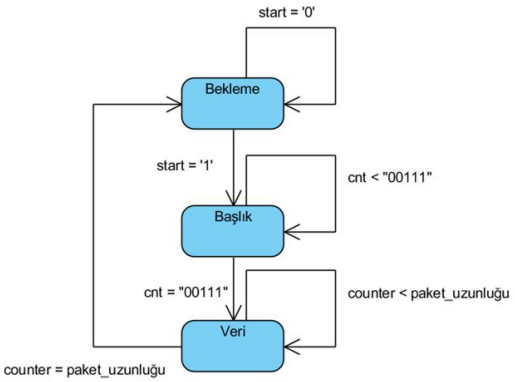
IP alıcı bloğu ile benzer durum makinası kullanan bu blok port numaralarına göre uygulama katmanına veri göndermektedir. Eğer port numarası 67 ise DHCP bloğuna göndermekte, Başka bir numara ise ait olduğu uygulamaya göndermektedir. UDP alıcı bloğun durum diyagramı Şekil 9' da gösterilmiştir.



Şekil 9:UDP alıcı blok durum diyagramı

3.9 UDP Gönderici blok

IP gönderici bloğu ile benzer durum makinası ile çalışır. Uygulama katmanından gelen paket isteğine göre uygun paketlemeyi durum makineleri kullanarak yapar ve IP gönderici bloğunun hafıza birimine yazar. Yazma işlemi bittiği zaman IP gönderici bloğuna başlama sinyalini gönderir. UDP gönderici bloğun durum diyagramı şekil 10'dagösterilmiştir.

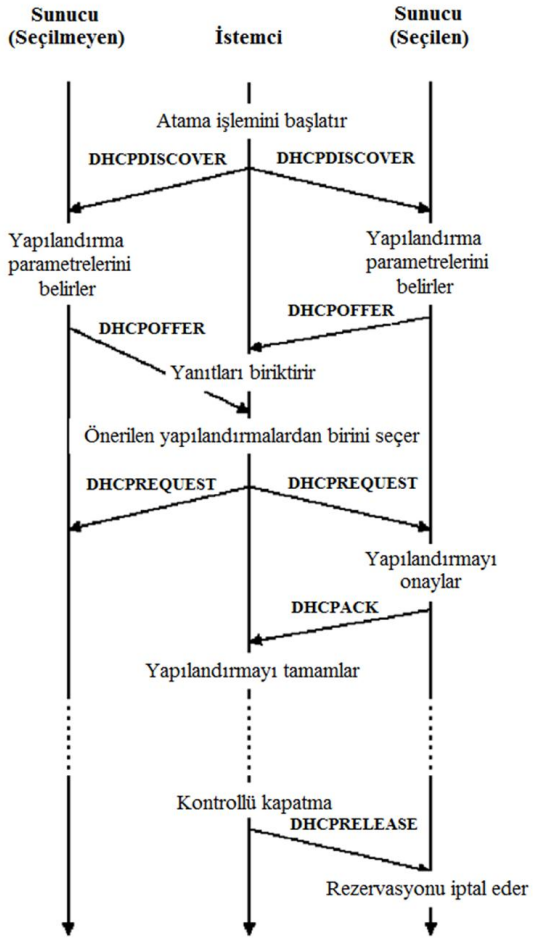


Şekil 10:UDP gönderici blok durum diyagramı

3.10 DHCP Bloğu

TCP/IP protokolünün kullanıldığı ağlardaki her bilgisayarın benzersiz (unique) bir IP adresi olmalıdır. IP adresi ve onunla birlikte ilişkili alt ağ maskesi, hem ana bilgisayarı hem de onun bağlı olduğu alt ağı belirler. Bir bilgisayar farklı bir alt ağa taşındığında IP adresinin değiştirilmesi gerekir. DHCP, bir istemci için bir DHCP sunucusunun IP adresi havuzundan dinamik olarak bir IP adresi atanmasını sağlar.

Bir istemciye IP adresi ve diğer yapılandırma parametrelerinin atanması sırasında DHCP sunucusu ve istemcisi arasında gerçeklesen tipik mesajlaşma adımları Şekil 11'deki akış diyagramında gösterilmiştir[14].

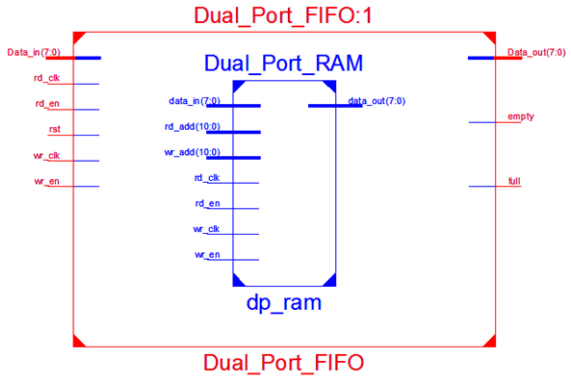


Şekil 11: DHCP protokolü işaretleşme trafiği

3.11 Dual Port Asenkron FIFO(First In First Out) Bloğu

Blokların kendi arasında ve dış dünya ile arasında olan senkronizasyon problemini gidermek için her bloğa bir FIFO entegre edilmiştir. FIFO(First In First Out) hafıza birimleri tampon görevi görmektedir. Bu FIFO ların tasarlanmasının amacı ise sistemi platformdan bağımsız bir hale getirmektir.

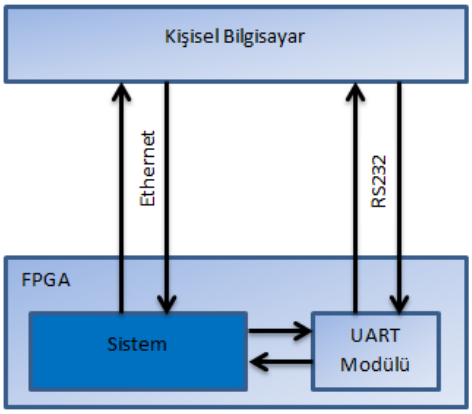
Tasarlanan FIFO aynı anda hem okumaya hem de yazmaya izin vermektedir ve bu işlemler asenkron bir biçimde gerçekleşmektedir. İç yapısında bir adet çift portlu blok RAM(Random Access Memory) ve 2 adet sayıcı yapısı bulunmaktadır. Sayıcılar RAM için adres bilgileri oluştururken dışarıdan gelen okuma veya yazma işaretlerine göre adresleme yapılmaktadır. Ayrıca okuma ve yazma adres sayıcılarının durumlarına göre FIFO nun dolu veya boş olduğu tespit edilebilmektedir. Bu sayede sistem içinde çeşitli kontrol işlemleri gerçekleştirilebilmektedir. Tasarlanan FIFO nun şematik bir çizimi Şekil 12'de görülmektedir. Karışıklık olmaması açısından iç yapıda sadece RAM bloğu gösterilmiştir



Şekil 12: Dual port asenkron FIFO şematiği

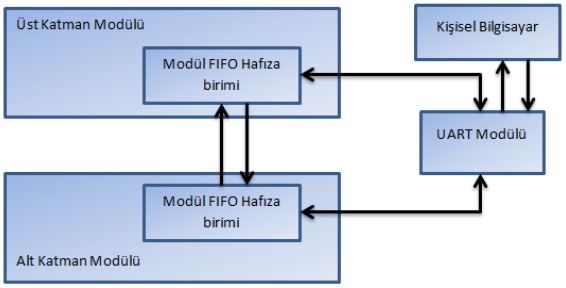
4. Sistem Geliştirme Ortamı

Sistem geliştirme aşamasında FPGA üzerinde bir UART alıcı-verici blok tasarlanarak kapalı döngü bir test ortamı sağlanmıştır. Ayrıca bu tasarlanan UART bloğu sistemi dışarıdan parametrik kontrole izin verir hale getirmektedir. Test ortamının genel bir bakışı Şekil 13'de görülmektedir.



Şekil 13: Sistem test mekanizması blok diyagramı

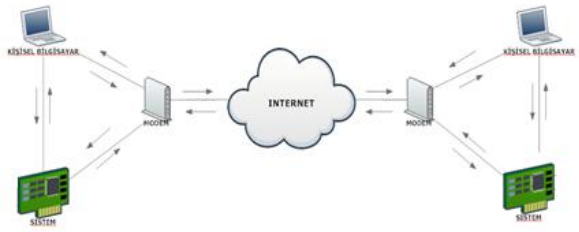
Sistem içerisine dahil olan UART arayüzü daha önce bahsedilen, bloklar arasında arayüz sağlayan FIFO birimlerinin içeriğini okuyup kişisel bilgisayara aktararak veri akışı sırasında paketlerin durumunu ve doğruluğunu test imkanı sağlamakta ve tasarımı kolaylaştırmaktadır. Test bloğunun çalışma yapısı Şekil 14'de görülmektedir.



Şekil 14: Test bloğu çalışma yapısı

5. Sonuç

Bu çalışma kapsamında FPGA üzerinde çalışan bir ağ istemci mimarisi oluşturulmuştur. Bu mimari FPGA tabanlı sistemleri internete bağlamak, kontrol ve veri aktarımı görevini yerine getirmek için pratik bir çözüm olarak sunulmuştur. Sistem ilk aşamada internet üzerinden video aktarımı için tasarlanmış olsa da tasarlanan mimari aynı zamanda günümüzde kişisel bilgisayarlarda kullanılan NIC(Ağ Arayüzü Kartı) mimarisinin bir iskeleti halindedir ve geliştirmeye açıktır. Sistemin istenen son çıktısı Şekil 15'te görülmektedir.



Şekil 15: İstenen son sistem çıktısı

Önerilen sistem TCP/IP protokol mimarisinin tüm katmanlarının görevlerini yerine getirebilmektedir. Her bir katmanda gerekli olan protokolleri gerçekleyen blok hiyerarşik bir yapıda çalışmaktadır. Her bir katman bir üst katmana gerekli hizmeti verirken bir alt katmandan gerekli hizmeti almak ister.

Tasarlanan mimaride diğer çalışmalardan farklı olarak 3 özellik öne çıkmaktadır. Bunlardan birincisi sistemin tüm katmanların fonksiyonlarını yerine getirerek hiçbir destek ünitesine(bilgisayar, işlemci) ihtiyaç duymadan tek başına çalışabilmesidir. Böylece sistemin taşınabilirliği artmakta ve bağımsız çalışabilme yeteneği kazanmaktadır. İkinci fark ise paket filtreleme esnasında karar verme aşaması paketin tümünün gelmesini beklemeden sadece başlık bilgileri alındıktan sonra yapılmaktadır, bu özellik sisteme hız kazandırmakta ve üst katmanları meşgul edecek gereksiz paket depolama işlemlerini yok etmektedir. Üçüncü fark ise sistemin tüm bloklarının ortak bir FIFO arayüzüne sahip olmasıdır. Bu özellik bloklar arası veri aktarımında senkronizasyon problemini sıfıra indirirken sistemin bütününü çok esnek bir hale getirmektedir.

VHDL ile tasarlanan sistem Digilent firmasına ait spartan 3E deneme kiti üzerinde çalıştırılmış olsa da mimari tasarım aşamasında platformdan bağımsız olacak şekilde yapıldığı için herhangi bir FPGA platformu üzerinde çalışması da mümkündür. Haberleşme hızı kit üzerinde bulunan SMSC 83C185 tümdevresinin olanak verdiği maksimum hız olan 100 Mbps olmuştur fakat daha gelişmiş bir tümdevre kullanılması halinde sistemin haberleşme hızı çok daha yüksek noktalara ulaşabilir. Sistem şu anda 4. katmana kadar olan görevleri yerine getirmektedir. Fakat DHCP ve ARP bloğu henüz tasarım ve test aşamasındadır. DHCP ve ARP bloğu tamamlandığı zaman istenilen noktaya ulaşılacaktır.

Şu anda UART protokolü ile dış birimlerle harici kontrol ve haberleşme yapabilen sistem FPGA mimarisinin ve yapılandırılan bloklar arası FIFO arayüzü sayesinde tüm endüstriyel protokoller ile kontrol edilebilme ve haberleşebilme altyapısına sahiptir. Ayrıca sistem üst

katmanlar kullanılmadan veri bağı katmanında çalışarak çok bloklu sistemlerde bloklar arası paket haberleşmesi sağlayabilen ana işlemci görevi görebilecek bir yapıya sahiptir.

Çalışma kapsamında ileride düşünülen video konferans sistemi için uygulama katmanına bir video ve kamera arayüzü eklenip internet üzerinden bir video konferans sistemi çalıştırılabilir. Ayrıca sistemin esnek yapısı sayesinde NIC(Ağ arayüz kartı) iskeleti içeren bu mimariye gerekli katmanlarda gerekli protokolleri desteklemesi için ek bloklar eklenerek gerçek zamanlı çalışan bir NIC kartı tasarlanabilir. FPGA üzerinde yapılan bu sistem ASIC(Application Specific Integrated Circuit-Uygulamaya Özgü Tümdevre) haline getirilerek maliyeti ve boyutu küçültülmüş bir NIC gerçeklenebilir. Tüm bunlar ileriki zamanlarda devam edilmesi öngörülen çalışmalardır.

6. Kaynaklar

[1] http://www.xilinx.com/support/documentation/spar3e-sk.htm [30.09.2012]

[2] SMSC 83C185 Datasheet

[3] IETF RFC 1180

[4] Cisco, Interconnecting Cisco Network Devices, CA Cisco System, Inc., 1999.

[5] Forouzan, B., (2007), Data Communication and Networking, McGraw-Hill Education, New York.

[6] IETF RFC 791

[7] Kurose, J. ve Ross K., (2008), Computer Networking A Top-Down Approach, Pearson Education Inc., Boston.

[8] IETF RFC 826

[9] IETF RFC 793

[10] IETF RFC 768

[11] IETF RFC 2131

[12] Lu, V., (2003) "Designing TCP/IP Functions In FPGAs", Yüksek Lisans Tezi, Delft University Of Technology Computer Engineering Department.

[13] http://www.hackersdelight.org/crc.pdf [30.09.2012]

[14] Ebil, F.,(2006) "DHCP'de Kullanıcı Kimliği ve Şifre Tabanlı Doğrulama ve Servis Seçme İşlemlerinin CHAP ve RADIUS Protokolleri Kullanılarak Gerçeklenmesi", Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.

Gömülü Sistemler ve Uygulamaları Sempozyumu - GÖMSİS 2012
İstanbul Teknik Üniversitesi, 29-30 Kasım 2012, İstanbul.

Analog ve Karışık İşaret Gözcü Tabanlı Doğrulamada Halelerin Kullanımı

Doğan Ulus
Boğaziçi Üniversitesi
Elektrik-Elektronik Mühendisliği Bölümü
Bebek 34342, İstanbul
e-posta: dogan.ulus@boun.edu.tr

Alper Şen
Boğaziçi Üniversitesi
Bilgisayar Mühendisliği Bölümü
Bebek 34342, İstanbul
e-posta: alper.sen@boun.edu.tr

***Özetçe*—Bu çalışmada analog ve karışık işaret (AKİ) *(Analog and mixed signal)* devre tasarımlarında kullanılmak üzere gözcü tabanlı bir doğrulama *(Assertion-based verification)* çözümü sunulmuştur. Bu çözüm çerçevesinde, analog işaretleri kendi tolerans ve saçılma değerleri ile birlikte ifade edebilmek için analog işaretlerin halesi kavramı öne sürülmüştür. Analog işaret haleleri, analog işaretler için onların etrafında kabul edilebilir bir alan tanımlamaktadır. Halelerin gözcü tabanlı doğrulamada kullanılması, analog işaretler için eşdeğerliğin de dahil olduğu yeni karşılaştırma işleçleri tanımlanmasına izin vermektedir. Tanımlanan bu yeni işleçler, gözcü tabanlı doğrulama dillerinin analog katmanına yerleştirilmiştir. Bu yaklaşımın faydaları, programlanabilir bir anahtar devresi ve gerilim kontrollü bir salıngaç *(Voltage-controlled oscillator)* üzerinde denenmiştir.**

I. GİRİŞ

Elektronik sektöründeki güncel yönelimler daha küçük cihazlarda daha çok işlevi zorunlu kılarken, elektronik sistemlerin sürekli artan karmaşıklığının, elektronik tasarım otomasyonu (ETO) olmadan, üstesinden gelmek artık imkansızdır. Ancak ETO alanındaki tüm gelişmelere rağmen bugün tasarlanan sistemler hatalara her zamankinden daha açıktır. Bu yüzden bu sistemlerin doğrulanmasında pek çok farklı yöntem öne sürülmüştür. Model ve eşdeğerlik kontrolü *(model- and equivalence-checking)*gibi biçimsel yöntemler, bu sistemlerin tam olarak doğruluğunu ispatlamaya çalışır ancak bu yöntemlerin çok büyük sistemler için kullanımı sınırlıdır. Bu yüzden gözcü tabanlı doğrulama gibi yarı-biçimsel yöntemler, endüstride kendine önemli bir yer bulmaktadır.

Bugün analog ve karışık işaret (AKİ) devrelerinin doğrulanmasında genellikle otomatik olmayan ve devreye özgü çözümler kullanılmaktadır. Ancak bu tarz doğrulama yöntemleri, –yongaların bünyesindeki AKİ birimlerinin sayısı ve karmaşıklığı artarken– günümüz yongalarının piyasaya çıkış zamanı ve ilk seferde çalışma gereksinimlerini karşılamada yetersiz kalmaktadır. Bu yüzden AKİ tasarımlarının doğrulanmasında gözcü tabanlı doğrulama (GTD) gibi daha yapısal ve biçimsel yöntemlere ihtiyaç vardır. GTD yöntemi, tasarım belirtimlerini biçimsel olarak ifade ederek ve benzetim sonuçlarının değerlendirilmesini otomatikleştirerek doğrulama süreçlerinin verimliliğini ve yeniden kullanılabilirliğini arttırmaktadır. Bu otomasyonu gerçekleştirebilmek için sistem özellikleri öncelikle biçimsel bir dilde ifade edilmeli, sonrasında ise benzetim sonuçlarının (biçimsel olarak ifade edilmiş) bu özelikleri sağlayıp sağlamadığı kontrol edilmelidir.

Bir sonraki kısımda konuyla ilgili, önceden yapılmış çalışmalar özetlenmiştir. Üçüncü kısımda AKİ gözcü dillerinin genel karakteristikleri ve bu dillerin yapısı kısaca açıklanmıştır. Dördüncü kısımda analog işaretler için geliştirdiğimiz hale kavramı verilmiştir. Analog işaretlerin gerçek zaman kavramı üzerinden mantıksal işaretlere dönüştürülmesi işlemi beşinci kısımda anlatılırken altıncı kısımda ise, programlanabilir anahtar ve gerilim kontrollü salıngaç devreleri üzerinde yaptığımız deneyler açıklanmıştır.

II. İLGİLİ ÇALIŞMALAR

Literatürde analog ve karışık işaret doğrulamada için pek çok yöntem önerilmiştir. Bu yöntemleri, [1], [2], [3], [4], [5], [6] gibi biçimsel yaklaşımlar ve [7], [8], [9], [10] gibi yarı-biçimsel yöntemler olarak iki ana dala ayırabiliriz. Bu yaklaşımların matematiksel kökenleri daha eski çalışmalara uzansa da bunların akademideki ve endüstrideki uygulamaları daha emekleme aşamasındadır.

Signal Temporal Logic (STL) [11], [12] Maler ve Nickovic tarafından gerçek zamanlı analog işaretlerin zamansal özellikleri gözlemek için ortaya atılmıştır. Bu çalışma *Metric Interval Temporal Logic* (MITL)'ı [13] gerçek zamanlı işaretlere doğru genişletirken analog özellikleri benzetim sonuçlarından kontrol etmektedir. Bu yaklaşımın analog özelliklerin benzetim devam ederken kontrol eden sürümü [14]'de sunulmuştur. Bu çalışmalar, *Analog Monitoring Tool* (AMT) [15] ile uygulanmıştır. Bu araç STL'de tanımlanan özellikleri benzetim sonuçlarından kontrol ederken analog işaretleri sadece karşılaştırma işleçleriyle mantıksal işaretlere dönüştürmektedir. Analog işaretlerde bulunan rasgele değişimler, analog işaretlerin toleransları dikkate alınmazsa genel doğrulama kalitesini düşürmektedir.

Öte yandan, daha büyük bir çalışmanın parçası olarak Lammermann ve diğerleri [8] SystemC-AMS tasarımları için zamansal kontrol birimi önermişlerdir. Yukarıdaki çalışmadan farklı olarak bu çalışmada gerçek zamanlı bir yaklaşım yerine saat-bazlı bir yaklaşım kullanılmış ve analog ifade-edebilirliği arttırmak için yeni analog işleçler tanımlamıştır.

III. Analog ve Karışık İşaret Gözcü Dilleri

Varolan gözcü tabanlı doğrulama (GTD) süreçlerinde, doğrulama sistem gereksinimleri ile başlamaktadır. Doğrulama mühendisleri bu gereksinimleri biçimsel bir dilde ifade ederek, benzetim sonuçlarının bu gereksinimleri karşılayıp karşılamadığını otomatik olarak denetler. Literatürde, analog ve karışık işaret (AKİ) devrelerin zamansal özellikleri için geliştirilmiş gözcü dilleri [7], [8], [9] genellikle *Linear Temporal Logic (LTL)* [16] tabanlıdır ve LTL'in ifade-edebilirliğini gerçek zamana doğru genişletmektedirler.

Bu gibi AKİ gözcü dilleri birkaç katmandan oluşurlar ve bu açıdan *Property Specification Language (PSL)*'den [17] etkilenmişlerdir. Böyle AKİ gözcü dilinin grameri Tablo I'de gösterilmiştir. Katmanlar dikey olarak yerleştirilmiş ve bilgiler en alt katmandan en üsttekine doğru sırayla aktarılmaktadır. PSL katmanlarına ek olarak AKİ gözcü dillerine analog özellikleri ifade edebilmek için bir Analog katman eklenmiştir. Bu Analog katman analog işaretlerdeki bilgiyi işleyip bir üstteki mantıksal katmana iletmektedir. Örneğin, analog bir işaret 5V'dan büyük mü diye sorulduğunda Analog katman analog işaretin 5V'dan büyük olduğu anları gösteren maktıksal bir işaret döndürmektedir. Bu şekilde analog işaretler maktıksal işaretlere dönüştürüldükten sonra mantıksal ve zamansal işlemler bu işaretlerin üzerine uygulanabilir.

Analog katmanda analog işaretleri mantıksal işaretlere dönüştürme işlemi pek çok şekilde yapılabilir. Analog işaretleri diğer analog işaretlerle veya sabit değerlerle karşılaştırmak en basit ve en çok kullanılan yöntemdir. Bu yüzden AKİ tasarımları için ilk ortaya çıkan GTD yöntemleri sadece karşılaştırma işleçlerini içermektedir [7], [18], [19], [20], [21]. Daha sonraki çalışmalar, analog işaretlerin frekans özelliklerinden de mantıksal işaretler döndürebilmektedirler [8]. Bu çalışmada AKİ gözcü dillerinin analog katmanı farklı bir yönde genişletilmiştir. Analog işaretlerin tolerans ve saçılma değerlerini ifade edebilmek için hale kavramı ortaya atılmıştır. Tablo I'de bizim bu çalışmada sunduğumuz gramer gösterilmiş ve bizim eklediğimiz özellikler taralı olarak belirtilmiştir.

Tablo I'de farklı semboller farklı işleçleri göstermektedir. Bu sembollerin açıklaması aşağıdadır.

- Zamansal İşleçler, $\circledcirc \in \{G, F\}$
- İkili Mantıksal İşleçler, $\bullet \in \{\wedge, \vee, \Rightarrow\}$
- Mantıksal Değil İşleci, $\neg$
- Hale Karşılaştırma İşleci, $\boxdot$
- Hale Oluşturma İşleci, $\mathcal{H}$
- Aritmetik İşleçler, $\odot \in \{+, -, *, /\}$

Gramerimizde zamansal ve mantıksal işleçler kendi anlamlarını korurken hale işleçlerinini anlamları ilerleyen kısımlarda anlatılacaktır. Kendi gramerimizi kullanarak yazdığımız bir özellik aşağıda verilmiştir.

$$G((p \wedge q) \Rightarrow F(G(\mathcal{H}(x)\ SGT\ \mathcal{H}(y + c))))$$

Bu özellikte p ve q mantıksal işaretlere, x ve y analog işaretlere, c ise sabit bir değere karşılık gelmektedir. Ayrıca

Tablo I
Sunulan AKİ Gözcü Dili Grameri

Zamansal Katman	ZmnIfd	::=	$\circledcirc$ MntkIfd
		|	$\circledcirc$ ZmnIfd
		|	ZmnIfd $\bullet$ ZmnIfd
		|	$\neg$ ZmnIfd
Mantıksal Katman	MntkIfd	::=	HaleIfd $\boxdot$ HaleIfd
		|	MntkIfd $\bullet$ MntkIfd
		|	$\neg$ MntkIfd
Analog Katman	HaleIfd	::=	$\mathcal{H}$(IsrtIfd, parametreler)
		|	IsrtIfd
	IsrtIfd	::=	AnalogIsaret
		|	Sabit
		|	IsrtIfd $\odot$ IsrtIfd

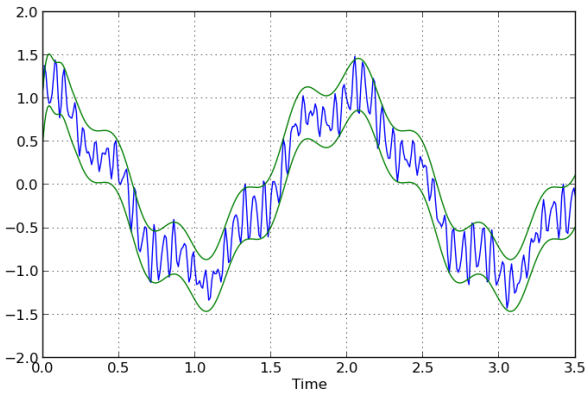
$\circledcirc$ Zamansal İşleçler — $\odot$ Aritmetik İşleçler
$\bullet$ İkili Mantıksal İşleçler — $\boxdot$ Hale Karşılaştırma İşlecleri
$\neg$ Mantıksal Değil İşleci — $\mathcal{H}$ Hale Oluşturma İşleci

SGT hale karşılaştırma işleci iken $\mathcal{H}$ işleci hale oluşturmak için kullanılmaktadır.

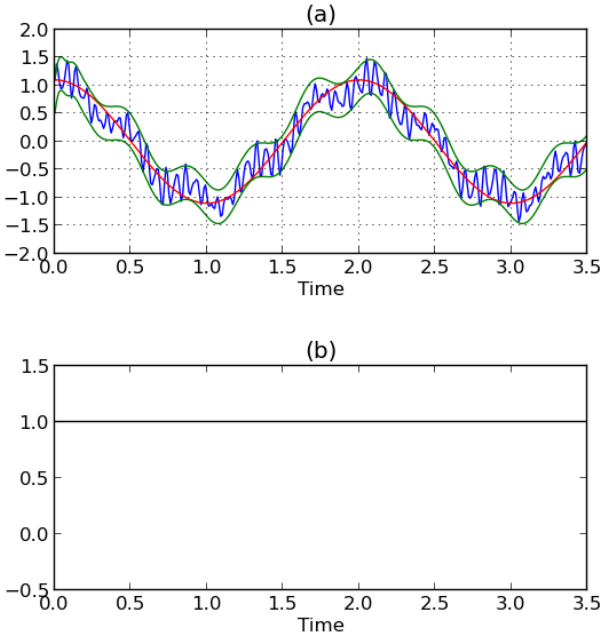
IV. Analog İşaret Halesi Kavramı

Astronomideki hale terimi güneşin veya ayın çevresindeki ışık halkasını tanımlamakta kullanılır. Fazla zor olmayan bir yorumla, bir nesnenin halesini, o nesnenin etki alanını gösteren bir sınır olarak düşünebiliriz. Bu yorumdan hareketle bu bildiride analog işaret halesini de analog işaretin tolerans ve saçılma değerlerini gösteren alt ve üst sınırlar olarak tanımlamaktayız. Bildirinin kalanında analog işaret halesi kısaca *hale* olarak, halenin analog işaret için tanımladığı sınırlar arasında kalan bölgeyi de *hale bölgesi* olarak isimlendireceğiz. Bu kavramı gözümüzde daha iyi canlandırabilmek için bir analog işaret ve onun halesi Şekil 1'de gösterilmiştir.

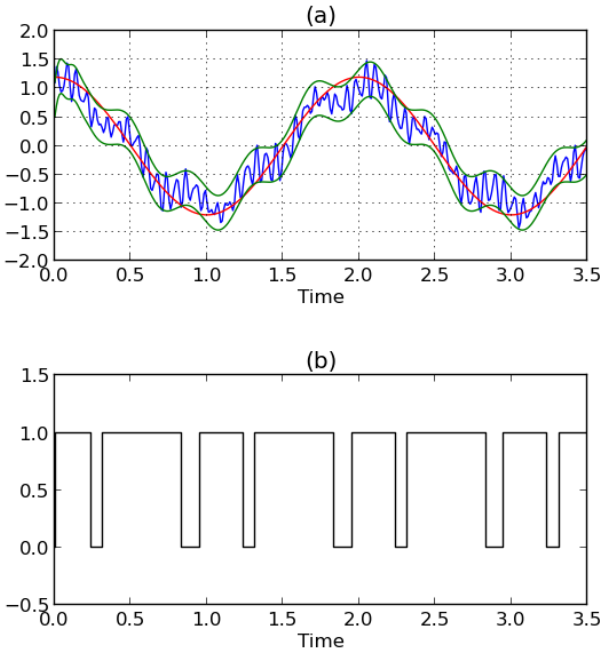
Haleler analog işaretlerin etrafında kabul edilebilir bir alan tanımlamaktadır. Bu sayede haleler, karşılaştırma gibi işlemlerde işaretler arasındaki küçük farklılara göz yummayı sağlarken analog özellikleri daha doğal bir biçimde ifade etmeye olanak verir. Örneğin, iki analog işaretin birbirine eşdeğer olduğuna karar vermek bir analog tasarımcı için oldukça kolaydır. İki işaretin genel dalga biçimine bakarak



Şekil 1. Bir analog işaret (mavi) ve onun halesi (yeşil).



Şekil 2. Bir referans işaretinin diğer bir işaretin hale bölgesinde olup olmadığının kontrolü. Referans işareti her zaman hale bölgesinde olduğu için döndürülen mantıksal işaret her zaman doğrudur.



Şekil 3. Bir referans işaretinin diğer bir işaretin hale bölgesinde olup olmadığının kontrolü. Döndürülen mantıksal işaretin değeri, referans işaretinin hale bölgesinde olduğu anlarda doğrudur.

bunların birbirine eşdeğer olup olmadığını söyleyebilir. Ancak bir bilgisayar için aynı sonuca varmak o kadar kolay değildir. İki işaretin eşdeğerliğine karar vermek için algoritmanın küçük farklılıklara göz yumabilmesi yani işaretlerin tolerans değerlerini dikkate alması ve aynı zamanda işaretlerin genel dalga şekillerinin aynı olduğunu kontrol etmesi gereklidir. Analog işaretler üzerindeki gürültü ve saçılım gibi rasgele değişkenler böyle bir karşılaştırma yapmayı zorunlu kılmaktadır. Halelerin gözcü dillerinde kullanılması bu ihtiyacın bir sonucudur.

İki örnek daha halelerin gözcü tabanlı doğrulamada kullanımını göstermek için Şekil 2 ve Şekil 3'de verilmiştir. Bu örneklerin a) kısmında maviyle çizilen analog işaret bir adet ana ve birkaç yan frekans bileşeninden oluşmaktadır. Kırmızıyla gösterilen referans işareti ise aynı ana frekansta yer alan bir sinüs dalgasıdır. Ancak genliği 1.1 katsayısı ile yükseltilmiştir. Bu iki işareti karşılaştırmak istediğimizde bilgisayar bu iki işaretin eşdeğer olduğuna karar vermekte yetersiz kalmaktadır. Oysaki analog bir tasarımcı, iki işaretin belli tolerans değerleri içinde eşdeğer olduğunu söyleyecektir. Analog işaretler etrafında haleler (yeşil ile çizilmiş) oluşturarak yapılan karşılaştırma işlemi böyle bir değerlendirme imkanı sağlayacaktır. Şekil 2'de referans işareti her zaman hale bölgesi içinde kalmaktadır. Bu yüzden b) kısmında döndürülen mantık işareti her zaman doğrudur. Bu da bize iki işaretin eşdeğer olduğu sonucuna götürür. Öte yandan, Şekil 3'de referans işareti zaman zaman hale bölgesinin dışına çıkmıştır, dolayısıyla b) kısmında döndürülen mantık işareti her zaman doğru değildir. Bu yüzden iki işaretin eşdeğerliğini söyleyemeyiz.

Böylece işaret karşılaştırması kullanıcı tarafından tanımlanan tolerans değerleri çerçevesinde yumuşatılmıştır. Bu yaklaşım analog tasarımcıların devreleri doğrularken kullandıkları yaklaşımla paraleldir.

V. Halelerin Karşılaştırılması

Halelerin birbiriyle karşılaştırılması ile mantıksal işaretler elde edilmektedir. Ancak analog işaretlerin doğrudan birbiriyle karşılaştırılmasından farklı olarak halelerin karşılaştırılmasında altı farklı durum oluşabilir. Bu altı durum için altı farklı işleç sunulmuş ve bu işleçler *tamamen-büyüktür* (sgt), *neredeyse-büyüktür* (ngt), *kapsar* (cvr), *kapsanmıştır* (cvd), *neredeyse-küçüktür* (nlt) ve *tamamen-küçüktür* (slt) olarak isimlendirilmiştir.

$$sgt = (u_1 \geq u_2) \wedge (u_1 \geq l_2) \wedge (l_1 > u_2) \wedge (l_1 > l_2)$$

$$ngt = (u_1 \geq u_2) \wedge (u_1 \geq l_2) \wedge (l_1 \leq u_2) \wedge (l_1 > l_2)$$

$$cvr = (u_1 \geq u_2) \wedge (u_1 \geq l_2) \wedge (l_1 \leq u_2) \wedge (l_1 \leq l_2)$$

$$cvd = (u_1 \leq u_2) \wedge (u_1 \geq l_2) \wedge (l_1 \leq u_2) \wedge (l_1 \geq l_2)$$

$$nlt = (u_1 < u_2) \wedge (u_1 \geq l_2) \wedge (l_1 \leq u_2) \wedge (l_1 \leq l_2)$$

$$slt = (u_1 < u_2) \wedge (u_1 < l_2) \wedge (l_1 \leq u_2) \wedge (l_1 \leq l_2)$$

Bu altı işleç, biçimsel olarak yukarıdaki şekilde ifade edilmektedir. Bu denklemlerde l_1 ve u_1, birinci halenin alt

ve üst sınır işaretleri iken l_2 ve u_2, ikinci halenin alt ve üst sınır işaretleridir.

Şekil 4'te düz ve kesik çizgilerle temsil edilen iki adet hale gösterilmiştir. Şekil 5'de ise bu iki hale, hale karşılaştırma işleçleri ile karşılaştırılmış ve bu karşılaştırmalarından döndürülen mantıksal işaretler gösterilmiştir.

VI. Deneyler

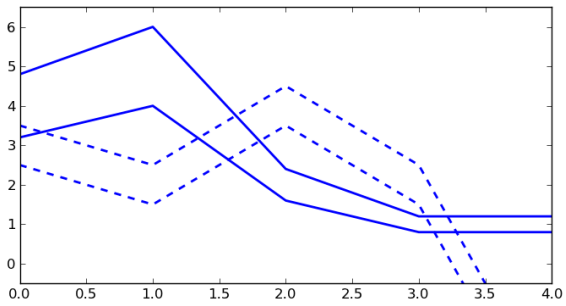
İki tane örnek devre üzerinde yaptığımız deneyler yaklaşımımızın uygulanabilirliğini göstermektedir. Birinci deneyde 1-bit yerel bellek içeren programlanabilir anahtar devresi kullanılmıştır. Bu devrenin yapısı gereği yerel bellek değeri maktıksal bir ise anahtar açıktır ve çıkış değeri giriş değerini takip etmelidir. Bu deneyde devrenin bu özelliği sağlayıp sağlamadığı doğrulanacaktır.

Analog işaretler, $a : in$, $a : out$ ve $a : on$ sırasıyla giriş, çıkış ve yerel belleke değerini ifade etmektedirler. Bu işaretler Şekil 6'in ilk üç çiziminde gösterilmiştir. Üçüncü çizimde $a : on$ kesik kırmızı çizgi ile gösterilen eşik değeri $(V_{dd}/2)$ ile karşılaştırılmış, döndürülen mantıksal işaret $b : on$, dördüncü çizimde gösterilmiştir. Çıkış işaretinin giriş işaretini takip ettiğini kontrol etmek için giriş işareti ertrafında bir hale oluşturulmuştur. Eğer $b : on$ değeri mantık bir değerindeyken çıkış işareti her zaman bu halenin içindeyse bu özellik doğrulanacaktır. Bu özellik biçimsel olarak şöyle yazılmıştır.

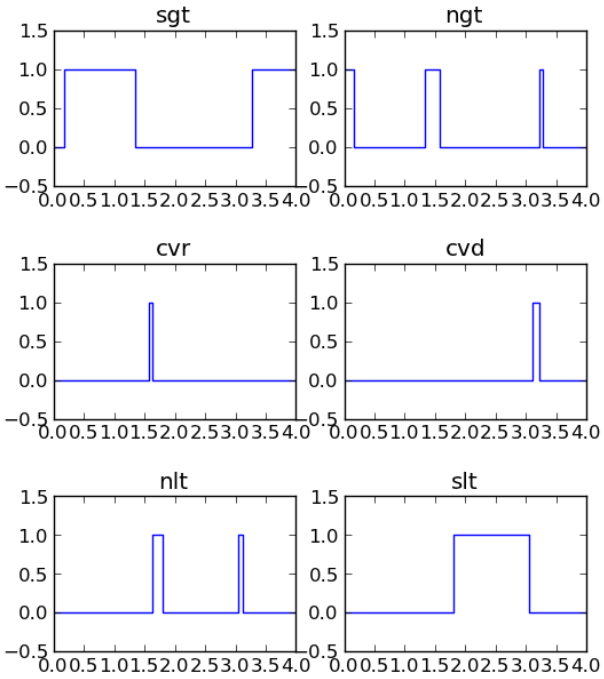
$$G(b : on \Rightarrow (a : out\ CVD\ \mathcal{H}(a : in)))$$

Bu özellikte G işleci zamansal mantığın *her-zaman* işlecine, CVD işleci öne sürdüğümüz hale karşılaştırma işleçlerinden *kapsanmıştır* işlecine ve $\mathcal{H}$ işleci hale oluşturma işlecine karşılık gelmektedir. Şekil 6'nin beşinci çiziminde ise $a : in$ (mavi), $\mathcal{H}(a : in)$ (yeşil) ve $a : out$ (kırmızı) birlikte ve yakınlaştırılmış bir biçimde gösterilmiştir. Bu çizimde çıkış işaretinin giriş işaretinin halesini tarafından kapsandığı görülmektedir. Şekildeki geri kalan çizimler özelliğin doğrulanmasının kalan aşamalarını göstermektedir.

İkinci deneyde ise birinci, ikinci ve üçüncü dereceden kazanç parametrelerine sahip gerilim kontrollü salıngaç (GKS) Verilog-AMS modeli kullanılmıştır. İkinci ve üçüncü derece



Şekil 4. Düz ve kesik çizgilerle gösterilmiş iki hale



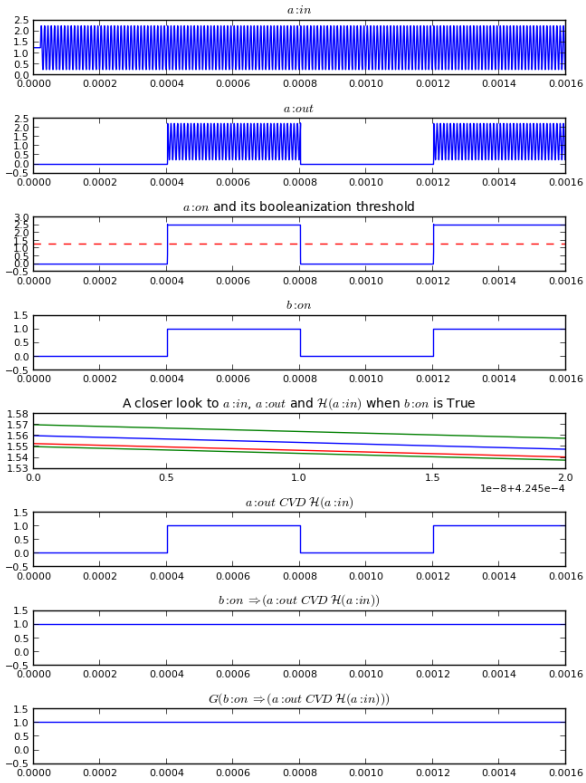
Şekil 5. Şekil 4'deki halelerin karşılaştırma sonuçları

kazanç parametreleri doğrusal olmayan karakteristiği modellemektedir. İlk önce GKS modelinin sadece birinci dereceden kazanç parametresi ile benzetimi yapılmış, ve bu benzetim sonucu, referans işaretimizi oluşturmuştur. Daha sonra ikinci ve üçüncü dereceden kazanç parametreleri üzerinde uygulanan Monte Carlo benzetimi ile çıkış işareti için bir hale oluşturulmuştur. Sonrasında referans işareti ile çıkış işaretinin eşdeğerliği haleler kullanarak doğrulanacaktır. Şekil 7'in ilk çiziminde GKS'nin giriş işareti, ikinci çizimde ise referans işaretinin halesi ile çıkış işaretinin halesi görülmektedir. Bu deneyde iki farklı eşdeğerlik denklemi denenmiştir. Üçüncü ve dördüncü çizimlerde $((out\ CVR\ ref) \vee (out\ CVD\ ref))$ ve $((out\ CVR\ ref) \vee (out\ CVD\ ref) \vee (out\ NLT\ ref) \vee (out\ NGT\ ref))$ eşdeğerlik durumlarının döndürdüğü mantıksal işaretler gösterilmiştir. Bu deneyde süreç saçılımlarının devrenin çıkışında yarattığı sapmaların istenen tolerans aralığında olup olmadığı kontrol edilmiştir.

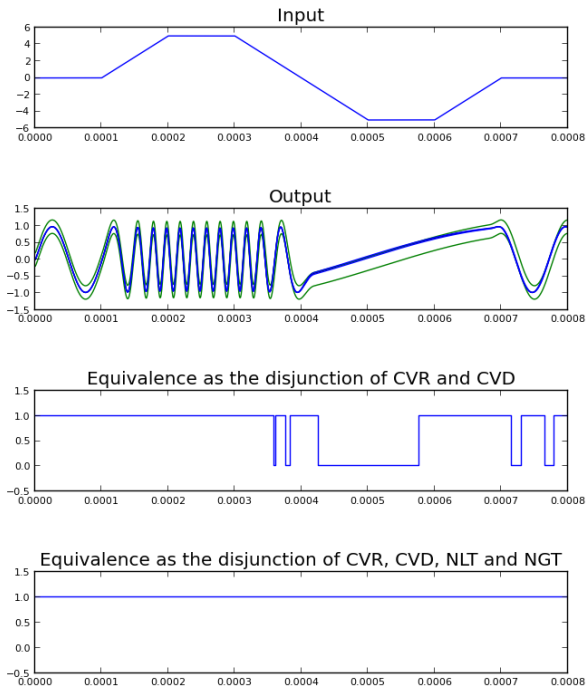
VII. Sonuç

Bu çalışmada analog işaretler için hale kavramı sunulmuştur. Haleler analog işaretlerdeki tolerans ve saçılım değerlerini ifade etmeyi sağlamaktadır. Halelerin gözcü tabanlı analog doğrulamada kullanılması analog özellik kontrollünde küçük farklılıklara göz yumarak daha doğal bir karşılaştırma imkanı yaratır. Bu analog tasarımcıların doğrulamada kullandığı yaklaşımlarla benzerdir.

Haleleri oluşturmak bu çalışmada dört farklı yöntem önerilmiştir. Bunun sayısı ve niteliği değişikliğe açıktır.



Şekil 6. Deney 1'deki özelliğin adım adım doğrulanması



Şekil 7. Deney 2'deki özelliğin doğrulaması

Oluşturulan haleleri, gözcü tabanlı doğrulamada kullanmak için hale karşılaştırma işleçleri tanımlanmış ve iki adet örnek devre üzerinde yapılan deneyler sunulmuştur.

Sonraki çalışmalarda ise analog işaretleri, mantıksal işaretlere dönüştürürken oluşan sorunlar ele alınacak, gözcü dillerinin analog ifade-edebilirliği zaman ekseninde arttırılmaya çalışılırken frekans ve diğer başka eksenlerdeki özelliklerin de bu çerçeve içinde doğrulanabilmesi için yöntemler geliştirilecektir. Son olarak ise daha fazla deney ile bu çalışmaların analog doğrulamada sağladığı faydalar gösterilecektir.

VIII. KAYNAKÇA

[1] S. Steinhorst, "Formal verification methodologies for nonlinear analog circuits," Ph.D. dissertation, Goethe-University of Frankfurt am Main, 2011.

[2] S. Steinhorst and L. Hedrich, "Model checking of analog systems using an analog specification language," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2008, pp. 324–329.

[3] M. Zaki, "Techniques for the formal verification of analog and mixed-signal designs," Ph.D. dissertation, Concordia University, 2008.

[4] G. Al-Sammane, M. Zaki, and S. Tahar, "A symbolic methodology for the verification of analog and mixed signal designs," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2007, pp. 249–254.

[5] E. Barke, D. Grabowski, H. Graeb, L. Hedrich, S. Heinen, R. Popp, S. Steinhorst, and Y. Wang, "Formal approaches to analog circuit verification," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2009, pp. 724–729.

[6] S. Little, D. Walter, K. Jones, C. Myers, and A. Sen, "Analog/mixed-signal circuit verification using models generated from simulation traces," *International Journal of Foundations of Computer Science*, vol. 21, no. 2, pp. 191–210, 2010.

[7] D. Ničković, "Checking timed and hybrid properties: Theory and applications," Ph.D. dissertation, Joseph Fourier University, 2008.

[8] S. Lämmermann, J. Ruf, T. Kropf, W. Rosenstiel, A. Viehl, A. Jesser, and L. Hedrich, "Towards assertion-based verification of heterogeneous system designs," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2010, pp. 1171–1176.

[9] R. Mukhopadhyay, S. K. Panda, P. Dasgupta, and J. Gough, "Instrumenting AMS assertion verification on commercial platforms," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 14, no. 2, 2009.

[10] J. Havlicek and S. Little, "Realtime regular expressions for analog and mixed-signal assertions," in *Proceedings*

of the Conference on the Formal Methods in Computer Aided Design (FMCAD). IEEE, 2011, pp. 155–162.

[11] O. Maler and D. Ničković, "Monitoring temporal properties of continuous signals," in *Proceedings of the Conference on Formal Modelling and Analysis of Timed Systems (FORMATS)*, 2004, pp. 152–166.

[12] O. Maler, D. Ničković, and A. Pnueli, "Real time temporal logic: past, present, future," in *Proceedings of the Conference on Formal Modelling and Analysis of Timed Systems (FORMATS)*, 2005, pp. 2–16.

[13] R. Alur, T. Feder, and T. Henzinger, "The benefits of relaxing punctuality," *Journal of the ACM (JACM)*, vol. 43, no. 1, pp. 116–146, 1996.

[14] O. Maler, D. Nickovic, and A. Pnueli, "Checking temporal properties of discrete, timed and continuous behaviors," *Pillars of computer science*, pp. 475–505, 2008.

[15] D. Ničković and O. Maler, "AMT: A property-based monitoring tool for analog systems," in *Proceedings of the Conference on Formal Modelling and Analysis of Timed Systems (FORMATS)*, 2007, pp. 304–319.

[16] A. Pnueli, "The temporal logic of programs," in *18th Annual Symposium on Foundations of Computer Science*. IEEE, 1977, pp. 46–57.

[17] H. Foster, E. Marschner, and Y. Wolfsthal, "IEEE 1850 PSL: The next generation," in *Proceedings of Design and Verification Conference and Exhibition (DVCON)*, 2005.

[18] S. Mukherjee and P. Dasgupta, "Incorporating local variables in mixed-signal assertions," in *TENCON 2009-2009 IEEE Region 10 Conference*. IEEE, 2009, pp. 1–5.

[19] H. Anand, J. Havlicek, and H. Miller, "Assertion based analog mixed signal verification," Freescale, Tech. Rep., 2008.

[20] D. J. O'Riordan and P. K. Bhattacharya, "PSL/SVA Assertions in SPICE," in *Proceedings of The Design & Verification Conference & Exhibition*, 2012.

[21] G. Nunn, F. Delguste, A. Khan, A. Verma, and B. Geden, "Using Digital Verification Techniques on Mixed-Signal SoCs with CustomSim and VCS," Synopsys, Tech. Rep., 2011.

Kriptoloji Uygulamalarına Özel Bir İşlemcinin Tasarlanarak FPGA Üzerinde Gerçeklenmesi

Onur Şahin, Berna Örs Yalçın
İstanbul Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Maslak, 34469, İstanbul
e-posta: {sahinonur2, siddika.ors}@itu.edu.tr

Özetçe—**Bu uygulamada genel amaçlı komutlarının yanı sıra gelişmiş şifreleme standardı (Advanced Encryption Standard-AES) algoritmasının hızlı ve kolay bir şekilde gerçekleştirilmesini sağlayacak olan komutları da komut setinde barındıran bir işlemci tasarlanarak FPGA üzerinde gerçeklenecektir. AES algoritmasına özel komutların eklenmesi ile işlemcinin veri şifreleme/şifre çözme işlemleri için performansının artırılması amaçlanmaktadır.**

Bu çalışmada ilk olarak AES komutlarını gerçekleştirecek olan donanımsal yapılar oluşturulmuştur. Bu komutların ekleneceği genel amaçlı işlemci de mimari yapısı oluşturulduktan sonra yüksek hızlı tümleşik devre tanımlama dili (Very high speed integrated circuit Hardware Description Language-VHDL) kullanılarak gerçekleştirilmiştir.

I. GIRIŞ

İşlemciler artık hemen hemen bütün elektronik cihazlar (cep telefonları, PDA, kameralar, duyarga düğümleri…) üzerinde bulunmaktadır ve kullanılan işlemcilerin özellikleri ve türü uygulamadan uygulamaya değişmektedir [1]. Bu tip uygulamaların geliştirilebilmesi açısından kullanılan iki temel yaklaşım bir genel amaçlı işlemci (General Purpose Processor, GPP) kullanmak ve ya uygulamaya özel tümleşik devre (Application Specific Integrated Circuit, ASIC) ile gerekli fonksiyonların gerçekleştirilmesini sağlamaktır [2].

Genel amaçlı işlemciler kolay programlanabilme özelliklerinden dolayı uygulama alanları çok geniştir fakat kullanılacağı uygulamaya göre bazı dezavantajları da birlikte getirebilmektedir ve bu tip işlemcilerin kaynakları yetersiz kalabilmektedir [2]. Örneğin, kriptoloji algoritmalarının bu tip işlemciler üzerinde yazılımsal olarak gerçekleştirilmesi esneklik açısından kolaylık sağlasa da, veri işleme gücü, bellek ve enerji bakımından kısıtlı olan bu işlemciler performans açısından oldukça yetersiz kalmaktadır [3]. Diğer bir olası çözüm olan ASIC tasarımı ve üretimi de oldukça maliyetli bir iştir. Bu ikilemin, son yıllarda kullanımlarında önemli bir artış gösteren uygulamaya özel komut setli işlemciler (Application Specific Instruction Set Processor, ASIP) ile çözümü yoluna gidilmektedir. Bu tip işlemcilerde geliştirilen uygulamalar, genellikle standart yazılım uygulamalarına göre daha yüksek performans sağlamasının yanı sıra yardımcı işlemci (co-processor) gibi sadece belirli bir uygulamaya adanmış donanım bloklarına göre de daha az silikon alanı kaplamaktadır [4]. ASIP'ler yazılımın sağladığı esneklik ile donanımların sağladığı performans avantajlarından birlikte yararlanılabilmesini sağlayan işlemcilerdir.

Ayrıca, günümüzde çoğu modern elektronik araçlar (PC, cep telefonları, ağ yönlendiricileri, akıllı kartlar, ağ duyargaları…) önemli verilere erişmekte, bu verileri işleyip depolamakta veya başka birimlerle haberleşme kanallarıyla iletmektedir. Bu durum dolayısıyla oluşan güvenlik ihtiyaçların [3] karşılanması için kullanılan yöntemlerden biri, bu çalışma da gerçekleştirildiği üzere, kriptografi uygulamalarına özgü fonksiyonlar gerçekleyebilen ve güvenlik unsuru ön planda tutulan bir ASIP işlemcinin gerçekleştirilmesidir.

Bildirinin ikinci bölümünde geliştirilen işlemciyi oluşturan blokların tasarlanması anlatılmıştır. Üçüncü bölümde AES (Advanced Encryption Standard) [4] algoritması hakkında bilgi verilirken dördüncü kısımda şifreleme biriminin tasarlanması irdelenmiştir. Beşinci kısımda tasarlanan bloklar ile işlemci mimarisinin oluşturulması açıklanmıştır. Altıncı kısımda tasarlanan işlemci üzerinde AES algoritmasının gerçekleştirilmesi anlatılmış ve yedinci bölümde çalışma ile ilgili sonuçlara yer verilmiştir.

II. TEMEL MIKROIŞLEMCI BLOKLARININ TASARLANMASI

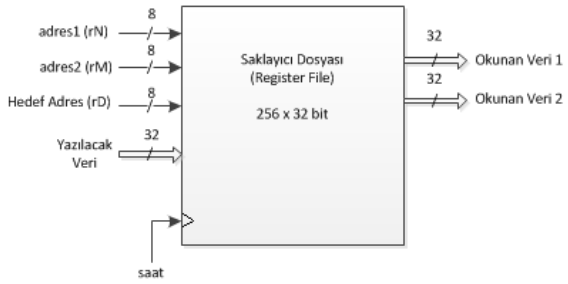
Bu çalışmada tasarlanan işlemci 32 bitlik bir işlemcidir. Bu yüzden verileri tutacak iç saklayıcıları ve veri yolları 32 bit genişliğindedir. Ayrıca tasarımda akümülatör benzeri saklayıcılar kullanılmamıştır çünkü yapılan işlemler doğrudan saklayıcı dosyasındaki veriler üzerinden yürütülebilmektedir.

Ayrıca işlemci RISC mimarisi baz alınarak geliştirilmiştir [5]. Bu mimari 1980'lerin başında da bilgisayar tasarımcılarının az sayıda buyruğu bulunan ve sık sık belleğe erişmeyen makinelerin mikroişlemci birimi içerisinde daha hızlı olduklarını tavsiye etmesi ile ortaya çıkmıştır [5].

A. Saklayıcı Dosyası

Gerçekleştirilen saklayıcı dosyasının genel görünümü Şekil-1'de verilmiştir.

Oluşturulan saklayıcı dosyası yapısında yazma işlemi saat işaretine göre senkron bir şekilde yapılırken, okuma işlemi saat işaretinden bağımsız olarak asenkron bir şekilde gerçekleştirilmektedir.



Şekil 1: Saklayıcı Dosyası

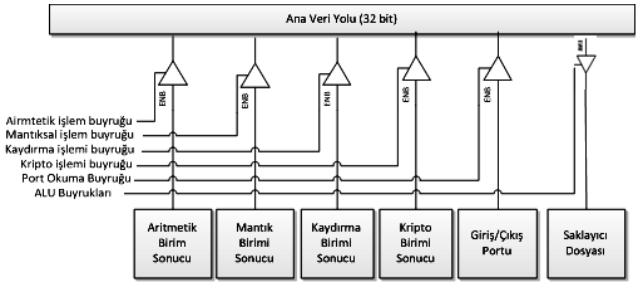
Saklayıcı dosyasına ilişkin gelişitilen komutlar Tablo-1'de verilmiştir. Oluşturulan komutlar ile saklayıcıların düşük ve yüksek anlamlı 16 bitlerine ayrı ayrı erişilmektedir. Bu şekilde komut uzunluğu daha kısa hale getirilebilmiştir.

Tablo1 : Saklayıcı Dosyası Komutları

Makine Kodu (hex)	Sembolik İfade	İşlem
80kkkknn	MovH rN,<k>	rN = <k> (Yüksek)
90kkkknn	MovL rN, <k>	rN = <k> (Düşük)

B. *Veri Yolu Düzeni*

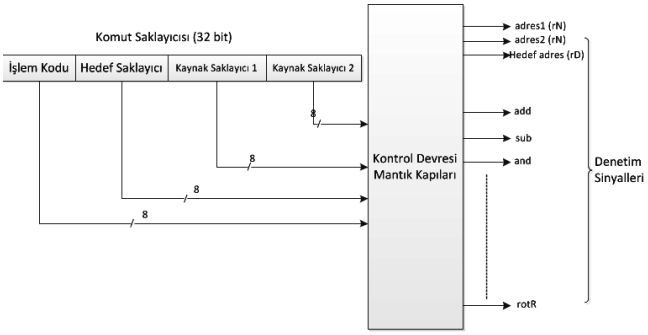
Tasarladığımız işlemcide 3 temel veri yolu bulunmaktadır; bir ana veri yolu ve 2 yardımcı veri yolu. Bütün veri yolları 32 bit kapasitesindedir. Yardımcı veri yolları saklayıcı dosyasından okunan verileri tutmaktadır ve busN ve busM olarak isimlendirilmişlerdir. Ana veri yoluna ise icra edilen komut sonucunda elde edilen veriler yazılmaktadır. Ana veri yolu aynı zamanda saklayıcı dosyasının girişine bağlı olduğundan, bu yoldaki veri saklayıcı dosyasına yazılabilmektedir. Ana veri yolu üstüne bağlı yapılar Şekil-2'de gösterilmiştir.



Şekil 2 : Ana Veri Yoluna Bağlı Birimler

C. *Komut Çözücü ve Kontrol Birimi*

Tasarlanan bu işlemcide komutlar bir saat darbesinde tamamlanmaktadır. Dolayısıyla her saat darbesinde yeni bir komut alınmakta ve kontrol biriminde zamanlama amaçlı olarak bir sayıcının kullanılması gerekmemektedir. Saat darbesinin yükselen kenarında, program belleğinden alınan komut, komut saklayıcısına yazılmaktadır. Kontrol birimi bu komut saklayıcısını okumakta ve bellekten okunacak verileri, hangi işlemin yürütüleceğini ve ana veri yoluna hangi birimin yazacağını belirleyen kontrol sinyallerini üretmektedir. Bu işlemleri gerçekleştiren kontrol biriminin genel görünümü Şekil-3'de gösterilmiştir.



Şekil 3 : Kontrol Biriminin Genel Görünümü

D. *Aritmetik Birim*

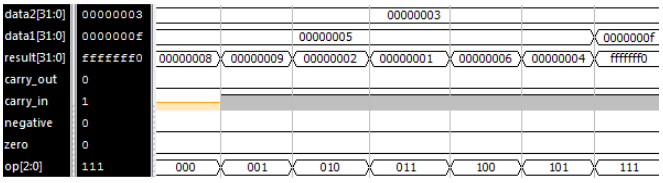
Aritmetik işlemleri icra edecek olan donanım birimi aritmetik birimdir. Bu işlemci için tasarlanan aritmetik birim 8 farklı işlem gerçekleştirilebilmektedir. Aritmetik birimin girişine kaynak saklayıcıların değerlerini tutan busN ve busM yardımcı veri yolları bağlanmıştır. 8 farklı işlemi kodlamak üzere 3 bitlik seçim girişi bulunmaktadır ve uygulanan seçim girişine göre yapılacak aritmetik işlem belirlenmektedir.

Aritmetik birimde gerçekleştirilen işlemler, onaltılık tabandaki makine kodu karşılıkları ile birlikte Tablo-2'de verilmiştir.

Tablo 2 : Aritmetik Birim Komutları

Makine Kodu	Sembolik İfade	İşlem
D0ddmmnn	ADD rN, rM, rd	rd = rN + rM
D1ddmmnn	ADDC rN, rM, rd	rd = rN + rM + c
D2ddmmnn	SUB rN,rM,rD	rd = rN - rM
D3ddmmnn	SUBC rN,rM,rD	rd = rN - rM - c
D40000nn	INC rN	rN = rN + 1
D50000nn	DEC rN	rN = rN - 1
D600mmnn	CMP rN,rM	rN ↔ rM
D70000nn	COM rN	rN = rN'

Tablo-2'de verilen komutları sırayla gerçekleştiren bir test kodu ile elde edilen sinyallerin durumu Şekil-4'de verilmiştir. Bu şekilde "data1" ve "data2" olarak adlandırılan saklayıcılar üzerinde Tablo 2'de verilen komutlar sırasıyla gerçekleştirilmektedir ve elde edilen sonuçların saklandığı "result" adı verilmiş olan 32 bitlik saklayıcı incelendiğinde işlemlerin başarıyla gerçekleştirildiği görülmektedir. "carry_in" elde girişini ve "op" sinyali de işlem seçim girişini göstermektedir.



Şekil 4 : Aritmetik Birim Simülasyon Sonucu

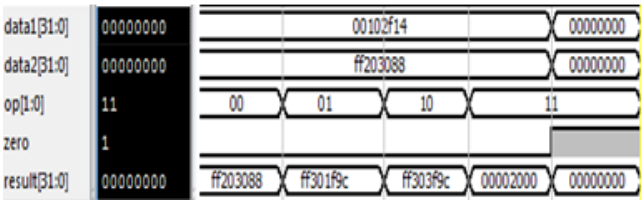
E. Mantık Birimi

Mantık birimi 3 temel mantıksal işlemi ve veri aktarma işlemini gerçekleştirmektedir. Veri aktarma komutu bir saklayıcıdaki değerin başka bir saklayıcıya da aktarılmasını sağlamaktadır. Mantık birimi komutlar onaltılık tabandaki makine kodları ile birlikte Tablo-3'de verilmiştir.

Tablo 3 : Mantık Birimi Komutları

Makine Kodu	Sembolik İfade	İşlem
C0dd00nn	Mov rN, rd	rd = rN + rM
C1ddmmnn	Xor rN, rM, rd	rd = rN xor rM
C2ddmmnn	Or rN, rM, rd	rd = rN or rM
C3ddmmnn	And rN, rM, rd	rd = rN and rM

Tablo-3'deki işlemleri verilen sırayla örnek bir değer için test eden kodun yazılması sonucunda elde edilen çıkış sinyalleri Şekil-5'te gösterilmiştir. "data1" ve "data2" kaynak saklayıcı üzerinde sırayla aktarma, "özel veya", "veya", "ve" işlemlerinin başarıyla gerçekleştirildiği, sonuçların yazıldığı "result" saklayıcısı ile gösterilmiştir.



Şekil 5 : Mantık Birimi Simülasyon Sonucu

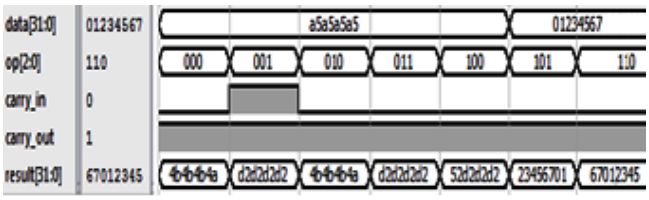
F. Kaydırma Birimi

Kaydırma birimine ilişkin komutlar Tablo-4'de verilmiştir.

Tablo 4 : Kaydırma Birimi Komutları

Makine Kodu	Sembolik İfade	İşlem
F00000nn	rotL rN	Sola Eldeli Öteleme
E10000nn	rotR rN	Sağa Eldeli Öteleme
E20000nn	Asl rN	Aritmetik Sola Kaydırma
E30000nn	Asr rN	Aritmetik Sağa Kaydırma
E40000nn	Lsr rN	Sağa Mantıksal Kaydırma
E50000nn	RotLB	Sola Bir Bayt Öteleme
E60000nn	RotRB	Sağa Bir Bayt Öteleme

Verilen komutların sırayla örnek bir veri için test edilmesi ile elde edilen simülasyon sonucu Şekil-6'da verilmiştir. "data" saklayıcısı üzerinde Tablo 4'de gösterilen işlemler farklı "op" girişlerinin verilmesi gerçekleştirilmiş ve sonuçların doğruluğu gözlenmiştir.



Şekil 6 : Kaydırma Birimi Simülasyon Sonucu

III. SIFRELEME BIRIMININ TASARLANMASI

Tasarlanan bu işlemciyi standart genel amaçlı işlemcilerden farklı yapan en önemli kısmı şifreleme birimidir. Bu birim ile işlemciye genel amaçlı kullanıma uygun komutların yanı sıra, kriptoloji uygulamalarına özel işlemleri gerçekleştirecek donanımların da eklenmesi sağlanmıştır. Tasarlanan bu şifreleme birimi, AES algoritmasını bu işlemcide en verimli şekilde gerçekleştirecek şekilde düşünülmüştür. İşlemcide AES algoritmasına özel komutların bulunması ile veri şifreleme ve ya şifreli veriyi çözme işlemleri çok daha hızlı ve verimli bir şekilde yapılabilmektedir.

A. AES Şifreleme Algoritması

Belçikalı iki kriptografi uzmanı Joan Daemen ve Vincent Rijmen tarafından geliştirilen Rijndael algoritması 2000 yılında Amerika Devlet Standartlar Enstitüsü (NIST) tarafından Gelişmiş Şifreleme Standardı (AES) olarak isimlendirilerek elektronik veri güvenliğinin sağlanması amacıyla veri şifreleme standardı olarak ortaya konmuştur [6]. Rijndael algoritması, anahtar uzunluğuna göre farklı sayıda döngü içeren ve her döngüde belirli işlemlerin, yenilenen anahtar da kullanılarak şifrelenecek veriye uygulanması ile gerçekleştirilen bir blok şifreleme algoritmasıdır

Rijndael algoritmasında, şifrelenecek ve ya şifresi çözülecek veriler 128 bit uzunluğundadır. Bu 128 bitlik veri, her bir elemanı 1 bayt yani 8 bit'e karşılık gelen, 4 satır ve 4 sütundan (4x4) oluşan bir matris ile ifade edilir. Bu matrise "durum" adı verilmektedir.

AES algoritmasının sözde kodu Şekil 6'da gösterilmiştir.

```
Cipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
   byte  state[4,Nb]

   state = in

   AddRoundKey(state, w[0, Nb-1])

   for round = 1 step 1 to Nr-1
      SubBytes(state)
      ShiftRows(state)
      MixColumns(state)
      AddRoundKey(state, w[round*Nb, (round+1)*Nb-1])
   end for

   SubBytes(state)
   ShiftRows(state)
   AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1])

   out = state
end
```

Şekil 6 : AES Algoritması [6]

Şekil 6'da verilen algoritma incelendiğinde her turda durum matrisi üzerinde bayt değiştirme, satır kaydırma, sütun karıştırma, tur anahtarı ekleme işlemlerinin gerçekleştirildiği görülmektedir. Satır kaydırma işlemi matrisin satırları üzerinde işlem yaparken sütun kaydırma işlemi matrisin sütunları üzerinde işlem gerçekleştirmektedir. Görüldüğü üzere son turda sütun karıştırma işlemi gerçekleştirilmemektedir.

B. Satır Sütun Dönüştürme Problemi

32 bit işlemcilerde AES algoritmasının gerçeklenmesi için durum matrisinin satır ve ya sütunları işlemcinin 32 bitlik saklayıcılarında tutulmalıdır. Fakat AES algoritması işlemlerinden satır kaydırma işlemi matrisin satırları üstünde işlem yaparken, sütun karıştırma işlemi matrisin sütunları üzerinde işlem yapmaktadır. Örnek olarak, işlemci saklayıcılarında satırları tuttuğumuzu düşünecek olursak sütun karıştırma işlemine sokulacak sütunu oluşturmak için 4 satırın da birer baytının okunarak bir sütun oluşturulması gerekir. Bu sütun oluşturma işlemini her bir sütun için gerçekleştirmemiz gerektiğini ve algoritma en az 10 tur içerdiğinden dolayı bu işlemin her turda tekrarlanacak olması gerektiğini düşünecek olursak, algoritmanın yazılım ile bu tip bir yaklaşım kullanılarak gerçekleştirilmesinin ciddi bir ek maliyet getireceği açıktır.

AES algoritmasını gerçekleştirmek üzere öne sürülmüş birçok komut yapısı bulunmaktadır. Uygun yöntemin seçilmesi için, tasarlanan genel amaçlı işlemcinin bir saat işareti çevriminde komutları icra ediyor olması ve 32 bit işlem yapabiliyor olmasının yanında gerçekleştirilecek mimariye uygun olması unsurları göz önünde bulundurulmuştur.

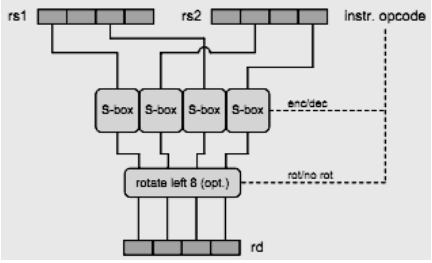
Satır-sütun dönüştürme problemine çözüm oluşturan ve tasarladığımız işlemci yapısına da uygun olarak görülen sbox4s, isbox4s, isbox4r, mixcol4s, ve imixcol4s komutlarının işlemcimizin komut setine eklenmesine karar verilmiştir [3]. Komutları icra edecek olan donanım birimleri de gene VHDL ile gerçekleştirilmiştir.

C. AES Komutları

Eklenen komutlar sütunlara dayalı işlem yapmaktadırlar ve iki kaynak saklayıcı gerektirmektedirler. Bu komut yapılarının sağladığı en belirgin avantaj ise satır kaydırma işlemini dolaylı olarak icra etmeleridir. AES algoritmasını bu komutlar ile gerçeklerken satır kaydırma işlemine gerek duymamaktayız.

1) sbox4s - isbox4s – sbox4r

Bu komutlar AES algoritmasının her turunda gerçekleştirilen bayt değiştirme ve ters bayt değiştirme işlemini gerçekleştirmektedir. Komutun yapısı Şekil-7'de gösterilmiştir.



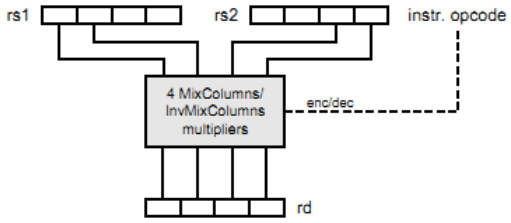
Şekil 7 : sbox4s, isbox4s ve sbox4r Komutlarının Yapısı [7]

Şekil-7'den görüldüğü üzere girişte iki adet kaynak saklayıcı bulunmakta ve bu saklayıcıların şekilde gösterilen baytları alınarak S-kutusundan geçirilmektedir. S-kutusu çıkışlarının bağlı olduğu bayt kaydırma birimi bu donanım bloğunun anahtar üretimi için de kullanılmasını sağlamaktadır ve sbox4r komutu ile bu işlem gerçekleştirilmektedir. Sbox4s komutu şifreleme işlemi için kullanılırken isbox4s komutu şifre çözme için kullanılmaktadır ve S-kutuları yerine ters S-kutuları kullanılması ile elde edilmiştir.

Sbox4s komutunun kullanılışına ilişkin simülasyon sonuçları bir sonraki kısımda açıklanan mixcol4s komutu ile birlikte verilmiştir. Bu şekilde AES algoritmasının bir turunda gerçekleştirilen bayt değiştirme, satır kaydırma ve sütun karıştırma işlemlerinin sadece anlatılan iki komut ile gerçeklenebildiği gösterilmiştir.

2) mixcol4s – imixcol4s

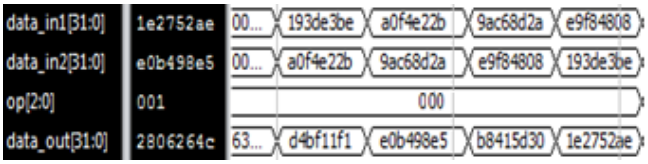
Sütun karıştırma ve ters sütun karıştırma işlemlerinin gerçekleştirilmesini sağlayan bu komutların yapısı Şekil-8'de gösterilmiştir.



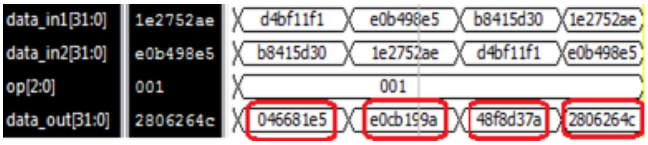
Şekil 8 : Mixcol4s ve imixcol4s Komutlarının Yapısı [7]

Bu komut yapısı gereğince, matrisin iki farklı sütununu tutan iki kaynak saklayıcısının Şekil 8'de gösterilen baytları alınarak tasarlanan sütun çarpıcı modülünden geçirilmektedir. Mixcol4s ve imixcol4s komutları arasındaki tek fark, imixcol4s için ters sütun çarpıcı modülünün kullanılmasıdır.

AES algoritmasının bir turununa karşılık gelen işlemler dizisinin sbox4s ve mixcol4s komutları ile gerçekleştirilmesine ilişkin simülasyon sonuçları Şekil-9 ve Şekil-10'da verilmiştir. Bu şekillerde "data_in1" ve "data_in2" kaynak saklayıcılarına karşılık düşmekte olup sırasıyla sbox4s ve mixcol4s işlemleri gerçekleştirilmektedir. Kutucuklar içinde gösterilen sonuçlar algoritmanın bir turunun başarı ile gerçekleştirildiğini göstermektedir.



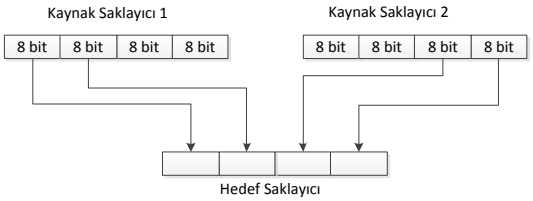
Şekil 9 : Sbox4s İşlemi Örneği



Şekil 10 : Mixcol4s İşlemi Örneği

3) Sütun Düzeltme (FixCol)

AES algoritmasının son turunda sütun karıştırma işlemi gerçekleştirilmemektedir. Ancak kullandığımız yapıda mixcol4s komutu aynı zamanda satır kaydırma işleminin dolaylı olarak gerçekleştirilmesinde rol almaktadır. Satır kaydırma işleminin, sütun karıştırma işlemi yapmadan gerçekleştirilebilmesi için FixCol komutu tanımlanmıştır. Bu komutun gerçekleştirdiği işlem Şekil-11'de gösterilmiştir.



Şekil 11 : FixCol Komutunun Yapısı

Şekil 11'deki komut yapısı ile Şekil 8'de verilen mixcol4s komut yapısı karşılaştırıldığında, FixCol komutunun tek farkının, verinin sütun çarpma bloğundan geçirilmemesi olduğu görülmektedir.

IV. TASARLANAN BLOKLAR ILE IŞLEMCI MIMARISININ OLUŞTURULMASI

Bu kısımda önceki bölümlerde tasarlanan yapılar da kullanarak oluşturulan Harvard mimarisindeki [8] RISC işlemcinin mimari yapısı ortaya konulacaktır.

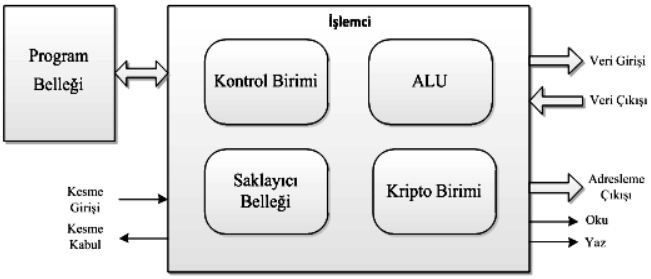
A. Işlemcinin Genel Görünümü

Tasarlanan işlemci komutları harici bir program belleğinden okumaktadır. İşlemci program belleğini adresleme amaçlı olarak toplam 12 adres hattına sahiptir. Bu da 4 kB uzunluğuna kadar komut adresleme imkânı sağlamaktadır. Program belleği işlemciye harici olarak bağlandığından yazılacak olan program uzunluğunu göre boyutu küçültülebilmektedir.

İşlemciye çevre donanımlarının bağlanabilmesi ve bu donanımlara erişim sağlanabilmesi açısından genel amaçlı bir giriş/çıkış iskelesi (port) tanımlanmıştır.

Bunun yanında, kesme kaynaklarının işlemciye bağlanabilmesini sağlayan bir adet maskelenebilir kesme girişi de bulunaktadır.

İşlemcinin genel dış görünümü Şekil 12'de gösterilmiştir.

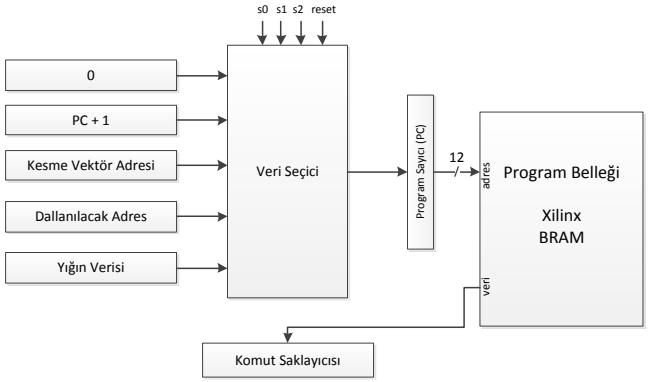


Şekil 12 : İşlemcinin Genel Görünümü

B. Işlemcinin Donanım yapısı ve Senkronizasyonu

1) Program Sayma İşlemi

Program sayıcı işlemciye sıfırlama (reset) sinyali geldiğinde sıfırlanmakta ve program belleğinin başına işaret etmektedir. Bu giriş ile işlemcinin yazılan programın tekrar başına dönmesi sağlanabilmektedir. Program sayıcı kesme girişi aktif olmadığında ve dallanma komutları dışındaki komutları yürütürken bir artarak devam etmektedir. Bu şekilde program belleğinden sıralı olarak komutlar okunarak yürütülmektedir. Dallanma komutları icra edilirken, eğer test koşulu varsa ve bu koşul sağlanıyorsa program sayıcıya dallanacağı adres yüklenmektedir. Şartsız dallanma komutları için program sayıcı komutda verilen adres ile yüklenmektedir. Sistemin genel blok diyagramı Şekil-13'de verilmiştir.



Şekil 13 : Program Okuma Birimi

Şekil 13'de verilen blok diyagramında bir veri seçici bloğuna bağlı farklı adresler gösterilmiştir. Kontrol birimi tarafından belirlenen seçim girişlerine göre ilgili adres program sayıcıya yüklenmekte ve program belleğinden o adrese karşılık gelen komut okunmaktadır. Okunan bu komut da komut saklayıcısına yerleştirilmektedir.

Program sayıcı, giriş/çıkış işlemleri için diğer komutlardan farklı olarak bir saat darbesi süresince bekletilmektedir. Xilinx'in blok ram çekirdeklerini harici bellek olarak kullanırken, okuma işlemlerinin işlemci ile senkron olması için okuma işlemi yürütülürken ekstra bir saat darbesi için buyruğun değişmemesi gerekmekte olduğundan, program sayıcı bekletilerek aynı buyruğu

okuması sağlanmıştır. Senkronizasyonu sağlayacak birim donanımsal olarak gerçeklendiğinden, kullanıcı bu durumdan habersiz olarak okuma işlemini başarılı bir şekilde gerçekleştirebilmektedir.

2) Yığın Yapısı

Tasarlanan işlemcide yığın yapısı bulunmaktadır ve temel görevi alt programlara dallanırken işlemcinin geri dönüş adresinin saklanmasıdır. Dallanma işlemi gerçekleştirilirken, bir sonraki buyruğun adresi, yani program sayıcının bir fazlası, yığına yazılmakta ve yığın göstergesi bir artırılmaktadır. Geri dönüş adresinin yanı sıra, bayrakların alt programda değiştirilmesine karşı tedbir olarak elde, sıfır ve negatif bayrakları da yığına yazılmaktadır.

3) Veri Yolu Düzeni

Oluşturulan yapıda toplam üç adet veri yolu bulunmaktadır. Bunlardan ikisi işleme tabi tutulacak verileri tutmakta olan 32 bitlik yardımcı veri yollarıdır, busN ve busM olarak isimlendirilmişlerdir. Bu iki veri yolu işlem birimlerine giriş olarak gelmektedirler. BusN ve busM veri yollarının dışında, bir adet de ana veri yolu bulunmaktadır. Ana veri yolu yapılan işlemlerin sonucunun yazıldığı veri yoludur. Ana veri yoluna birçok birim bağlıdır ve hangi birimin bu yola erişim sağlayabileceği kontrol birimi tarafından belirlenir. Anlatılan bu veri yolu düzeni ile birlikte işlemcinin genel donanım şeması Şekil-14'de verilmiştir.

Bu donanım şemasında işlem birimlerinin ve giriş/çıkış biriminin saklayıcı dosyası ile veri yolları kullanılarak oluşturulan bağlantısı gösterilmiştir. Komut saklayıcısından okunan değerlere göre kontrol birimi sinyaller üretmekte ve yapılacak işlemi belirlemektedir. Bu yüzden kontrol biriminin diğer birimler ile bağlantıları bulunmaktadır. Bayraklar ve program sayıcının o anki değeri alt program çağrılarında şemada da gösterildiği üzere yığın yapısında saklanmaktadır.

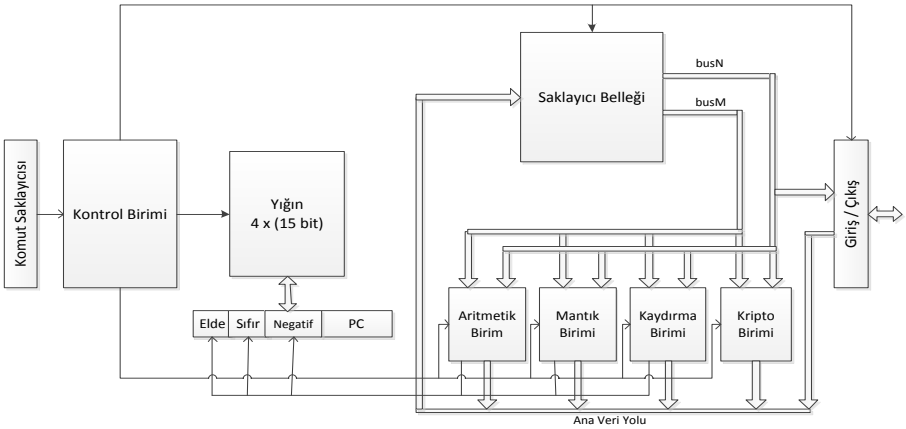
4) Giriş/Çıkış Organizasyonu

İşlemciye çevre donanımların bağlanabilmesi ve veri giriş-çıkışı yapabilmesi için bir adet iskele bulunmaktadır. Şekil-14'de de gösterildiği üzere iskeleye yazılacak veriler busN veri yolu üzerinden sağlanmakta, okunan veriler ise ana veri yoluna aktarılmaktadır. Bu işlemcinin iskelesinin 18 bitlik adresleme hattına sahip olması sebebiyle, birçok harici birim adres kod çözücüler kullanılarak işlemciye bağlanabilecektir.

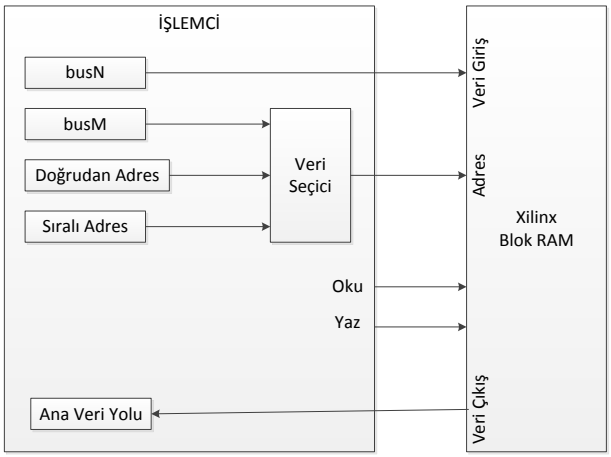
Bu giriş/çıkış iskelesi işlemciye birçok artı özellik kazandırmaktadır. Bu özelliklerin en önemlisi işlemciye dolaylı adresleme yeteneği kazandırmasıdır. Tasarlanan bu işlemci, komutları bir saat darbesinde gerçekleştirmektedir ve dolayısıyla doğrudan adresleme ile çalışmaktadır. Ancak birçok uygulamada saklayıcılarda verinin kendisinin değil adresinin saklanması gerekebilmektedir. Bu iskele sayesinde işlemci saklayıcılarındaki değer adres olarak çıkışa bağlı bir bellek elemanına gönderilebilmektedir. Bu adres gönderme işlemi busM veri yolu üzerinden yapılmaktadır. Dolaylı adresleme imkanının yanında, sıralı bellek gözlerinin tek komut ile iskele üzerinden okunmasını sağlayan özellik de okuma işlemlerinde kolaylık sağlayabilmektedir.

Örnek bir şema oluşturması açısından iskeleye bir blok ram bağlandığı durum için oluşan blok şema Şekil-15'da gösterilmiştir.

Blok şemada busN veri yolunun veri çıkışı, ana veri yolunun ise veri girişi için kullanıldığı görülmektedir. Ayrıca veri seçici yapısına bağlı 3 farklı adres ile okunacak verinin adresi belirlenebilmektedir. Görüldüğü üzere busM veri yolu adres girişi olarak kullanılabilmektedir ve bu da dolaylı adresleme imkanı sağlamaktadır.



Şekil 14 : İşlemcinin Donanım Yapısı



Şekil 15 : Giriş/Çıkış Organizasyonu

C. Komut Kümesi

Tablo-1,2,3,4'de saklayıcı dosyası, aritmetik birim, mantık birimi ve kaydırma birimine ilişkin komutlar listelenmişti. İşlemcinin diğer komutları bu kısımda listelenmiştir. Program akışının belirli şartlar altında ve ya koşulsuz olarak başka bir adrese dallanmasını sağlayan dallanma komutları ve alt program çağırma komutları Tablo-5'de listelenmiştir.

Tablo 5 : Dallanma Komutları

Makine Kodu	**Sembolik**	**İşlem**
40000ddd	Bne <dest>	Dallan Eşit Değilse
44000ddd	Beq <dest>	Dallan Eğer Eşitse
48000ddd	Bgt <dest>	Dallan Eğer Büyükse
4C000ddd	Blt <dest>	Dallan Eğer Küçükse
50000ddd	Bge <dest>	Dallan Eğer Büyük Ve Ya Eşitse
54000ddd	Ble <dest>	Dallan Eğer Küçük Ve Ya Eşitse
00000ddd	Call <dest>	Alt Program Çağrısı
10000ddd	Goto	Koşulsuz Dallanma
20000000	Ret	Alt Programdan Dönüş
24000000	Reti	Kesme Programından Dönüş

Kesme girişlerinin aktif edilmesi ve ya engellenmesi için Tablo-6'da verilen komutlar kullanılmaktadır.

Tablo 6 : Kesme İşlemleri

Makine Kodu	**Sembolik**	**İşlem**
28000000	di	Kesmeleri Engelle
2C000000	ei	Kesmelere İzin Ver

İskele üzerinden okuma yazma işlemlerinin gerçekleştirilmesi için kullanılabilen komutlar da Tablo 7'de tanıtılmıştır. Komutların verilen makine kodu karşılıkları on altılık tabandadır.

Tablo 7 : İskele Okuma/Yazma Komutları

Makine Kodu	**Sembolik İfade**	**İşlem (İskele)**
3000mmnn	Mov (rM), rN	Dolaylı Adrese Yazma
340000nn	Mov (++), rN	Sıralı Adrese Yazma
38kkkknn	Mov (<k>),rN	Doğrudan Adrese Yazma
F000mmnn	Mov rN, (rM)	Dolaylı Adresden Okuma
F40000nn	Mov rN, (++)	Sıralı Adresden Okuma
F8kkkknn	Mov rN,(<k>)	Doğrudan Adresi Okuma

Tablo-8'de ise AES algoritmasını gerçeklemek üzere komut setine eklenen özel komutlar gösterilmiştir.

Tablo 8 : AES Komutları

Makine Kodu	**Sembolik İfade**	**İşlem**
D8ddmmnn	Sbox4x rN,rM,rD	Bayt Değiştirme
D9ddmmnn	Mixcol4s rN,rM,rD	Sütun Karıştırma
DAddmmnn	isbox4s rN,rM,rD	Ters Bayt Değiştirme
DBddmmnn	imixcol4s rN,rM,rD	Ters Sütun Karıştırma
DCddmmnn	RotWord rN,rM,rD	(rN = rM) rd = sbox4r(rN)
DDddmmnn	FixCol rN,rM,rD	Sütun Düzeltme

V. AES ALGORITMASININ TASARLANAN IŞLEMCI ILE GERÇEKLEŞTIRILMESI

Komut kümesindeki komutlar ile AES algoritmasını gerçekleyecek olan program yazılarak tasarlanan işlemci üzerinde çalıştırılmıştır.

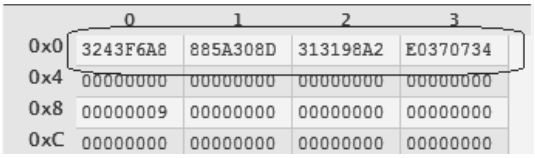
Program öncelikle sembolik dil kullanılarak oluşturulmuş, sonrasında komutlar makine kodu karşılıklarına çevrilerek işlemcinin program belleğine yüklenmiştir. Yazılan program Tablo-9'da verilmiştir.

Tablo 9 : AES Algoritması İçin Yazılan Program

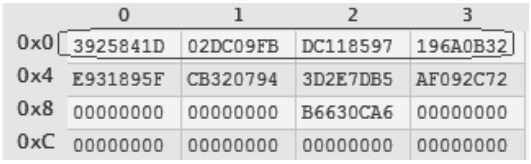
Etiket	Komut	Açıklama	Makine Kodu
basla	MOVH R0, 3243h	Şifrelenecek veri saklayıcılara yazılıyor	80324300
	MOVL R0, F6A5h		90f6a800
	MOVH R1, 885Ah		80885a01
	MOVL R1, 308Dh		90308d01
	MOVH R2, 3131h		80313102
	MOVL R2, 98A2h		9098a202
	MOVH R3, E037h		80e03703
	MOVL R3, 0734h		90073403
İlk_tur	MOVH R8, 0000h	Dongu sayacı = 9	80000008
	MOVL R8, 0009h		90000908
	MOV R10, <00>	İskeleden 00 adresini oku	f800000a
	XOR R0,R10,R0	İskeleden bir sonraki veriyi al	c100000a
	MOV R10,(++)		f400000a
	XOR R1,R10,R1		c101010a
	MOV R10,(++)		f400000a
	XOR R2,R10,R2		c102020a
	MOV R10,(++)		f400000a
	XOR R3,R10,R3		c103030a
Dongu	Sbox4s R0,R1,R4	Sbox4s İşlemi	d8040100
	Sbox4s R1,R2,R5		d8050201
	Sbox4s R2,R3,R6		d8060302
	Sbox4s R3,R0,R7		d8070003
	Mixcol4s R4,R6,R0	Mixcol4s İşlemi	d9000604
	Mixcol4s R5,R7,R1		d9010705
	Mixcol4s R6,R4,R2		d9020406
	Mixcol4s R7,R5,R3		d9030507
	MOV R10,(++)		f400000a
	XOR R0,R10,R0		c100000a
	MOV R10,(++)		f400000a
	XOR R1,R10,R1		c101010a
	MOV R10,(++)		f400000a
	XOR R2,R10,R2		c102020a
	MOV R10,(++)		f400000a
	XOR R3,R10,R3		c103030a
	DEC R8	Sayacı Azalt	d5000008
	CMP R8,#00	0 olmuş mu?	d6000809
	BNE Dongu	Olmamışsa bir sonraki tura geç	40000012
	Sbox4s R0,R1,R4		d8040100
	Sbox4s R1,R2,R5		d8050201
	Sbox4s R2,R3,R6		d8060302
	Sbox4s R3,R0,R7		d8070003
	Fixcol R4,R6,R0		dd000604
	Fixcol R5,R7,R1		dd010705
	Fixcol R6,R4,R2		dd020406
	Fixcol R7,R5,R3		dd030507
	MOV R10,(++)		f400000a
	XOR R0,R10,R0		c100000a
	MOV R10,(++)		f400000a
	XOR R1,R10,R1		c101010a
	MOV R10,(++)		f400000a
	XOR R2,R10,R2		c102020a
	MOV R10,(++)		f400000a
	XOR R3,R10,R3		c103030a
SON	MOV R0,R0	İşlem yok	c0000000
	BRA SON		10000035

Programın işlemcide çalıştırılması için makine koduna çevrilen program, program belleği olarak kullanılan BRAM'in içine yazılmıştır. Blok ram'e bu verilerin yazılması için ".coe" uzantılı BRAM başlatma dosyası (Bram Initialization File) oluşturulmuştur. Bu şekilde yazılan kodlar kolayca program belleğine yerleştirilebilmektedir.

Şekil 16'da şifrelenecek olan örnek bir verinin [5] saklayıcı dosyasındaki görünümü ve Şekil 17'de bu verinin şifrelenmiş hali gösterilmiştir.



Şekil 16 : Şifrelenecek Veri



Şekil 17 : Şifrelenmiş Veri

VI. SONUÇ

Bu çalışmada komut seti genişletilmesi yöntemi ile kriptoloji uygulamarında kullanılabilecek, hızlı ve kolay bir şekilde veri şifreleme/çözme işlemlerini gerçekleştirebilirken genel amaçlı kullanıma da olanak sağlayabilecek komutları bulunan bir işlemci donanımsal olarak ortaya konmuştur.

Çalışmanın başında temel aritmetik, mantıksal, giriş-çıkış ve bellek aktarım komutlarını gerçekleştirebilen bir işlemci VHDL kullanılarak gerçekleştirilmiştir. Genel amaçlı işlemci tamamlandıktan sonra, işlemcimizin yapısına en uygun olan, etkin ve kolay bir şekilde AES algoritmanın programlanmasını sağlayacak komutlar seçilerek işlemcinin komut listesinde bulunan boşluklara eklenmiş, yeni donanımlar da işlemci ile bütünleştirilmiştir. AES için eklenen komutlar ile satır kaydırma işlemine gerek duyulmadan şifreleme ve şifre çözme işlemleri yapılabilmektedir. Bu da üçüncü kısımda belirtilen satır-sütun dönüştürme problemini ortadan kaldırabilmemizi sağlamaktadır.

Oluşturulan bu işlemcinin genel amaçlı ve AES'e özel komutları birlikte kullanılarak örnek bir şifreleme işlemi için AES algoritmasını gerçekleştirecek program kodu yazılmış ve işlemci üzerinde koşturularak başarılı ve doğru bir şekilde şifreleme işleminin gerçekleştiği gözlenmiştir.

Oluşturulan işlemcinin FPGA üzerinde kapladığı alan yapılan sentezleme işlemi sonucunda elde edilen ve Şekil-18'de verilen sentez raporunda gosterilmistir. Spartan 6 - xc6slx75t FPGA modeli için yapılan sentezleme sonucunda bütünüyle kullanılan LUT-FF oranı %5 ve tampon olarak kullanılan BUFG/BUFGCTRL'lerin ise kullanım oranının %25 olduğu görülmektedir. Ayrıca işlemcinin çalışma

Logic Utilization	Used	Available	Utilization
Number of Slice Registers	144	93296	0%
Number of Slice LUTs	1765	46648	3%
Number of fully used LUT-FF pairs	96	1813	5%
Number of bonded IOBs	4	348	1%
Number of Block RAM/FIFO	2	172	1%
Number of BUFG/BUFGCTRLs	4	16	25%

Şekil 18 : FPGA Kaynak Kullanım Oranları

frekansının 78.812 MHz'e kadar çıkabileceği görülmüştür.

Şekil 19'da SPARC-V8 mimarisine sahip Leon-2 işlemcisi üzerine bu makalede belirtilen AES komutlarının eklenmesiyle elde edilen sonuçlar verilmiştir. Görüldüğü üzere bu işlemcide AES komutlarının eklenmesi ile şifreleme/çözme işlemleri C dilinde 458, makine dilinde ise 219 saat darbesi sürmektedir [3].

	Key exp.	Encr. perf.		Code size	
Implementation	Cycles	Cycles	Speedup	Bytes	Rel. change
No extensions (pure SW)	739	1,637	1.00	2,168	0.0%
sbox4s & mixcol4s (C)	316	458	3.57	568	-73.8%
sbox4s & mixcol4s (ASM)	316	219	7.47	412	-81.0%
sbox4s & mixcol4s, unrolled	316	196	8.35	896	-58.7%

Şekil 19 : Leon-2 İşlemcisi için Elde Edilen Sonuçlar [3]

Bizim tasarladığımız işlemcide ise şifreleme/çözme işlemleri için yazılan programlar, şifrenin harici bir bellek biriminde olduğu durum için 224, şifrenin saklayıcı dosyasında tutulduğu durum için ise sadece 142 saat darbesinde tamamlanmaktadır.

Oluşturulan bu işlemci, esnek ve basit bir yapıya sahip olmasından ve komut setinde başka komutlar için uygun boşluklar yer almasından dolayı geliştirilmeye açıktır. Farklı şifreleme algoritmalarını gerçekleyecek komutlar da tasarlanıp eklenerek işlemcinin kriptoloji alanı için işlevselliği artırılabilecektir.

VII. REFERANSLAR

[1] Koushanfar F., Prabhu V., Potkonjak M., Rabaey J., Processors for Mobile Applications**,** *Proceedings 2000 International Conference on Computer Design*, 603-8, September, 2000

[2] Gour D., Jain M., ASIP Design Space Exploration: Survey and Issues. *(IJCSIS) International Journal of Computer Science and Information Security*,Vol. 9, No. 4, 2011

[3] Tillich S. and Großschädl J., Instruction Set Extensions for Efficient AES Implementation on 32-bit Processors. In L. Goubin and M. Matsui, editors, *Workshop on Crypto-graphic Hardware and Embedded Systems CHES 2006*, volume LNCS 4249, pages 270–284, Yokohama, Japan, October 10–13 2006. Springer-Verlag

[4] Tillich S. and Großschädl J. Accelerating AES Using Instruction Set Extensions for Elliptic Curve Cryptography. *In International Workshop on Information Security & Hiding (ISH 05), in conjunction with International Conference on Computational Science & Its Applications (ICCSA 2005)*, LNCS 3481, pp. 665–675. Springer, 2005.

[5] Mano, M.M., 2007. Bilgisayar Sistemleri Mimarisi, Literatür Yayıncılık, İstanbul

[6] FIPS 197, 2001. Advanced Encryption Standard, National Institute of Standarts and Technology.

[7] Tillich S., 2008, Instruction Set Extensions for Support of Cryptography on Embedded Systems, *PhD Thesis*, Graz University of Technology, Avusturya

[8] Kong, J.H.; Ang, L.-M.; Seng, K.P., "Minimal Instruction Set AES Processor using Harvard Architecture," *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on* , vol.9, no., pp.65-69, 9-11 July 2010

Gömülü Sistemler ve Uygulamaları Sempozyumu - GÖMSİS 2012
İstanbul Teknik Üniversitesi, 29-30 Kasım 2012, İstanbul.

Gerçek Zamanlı Bir Hücresel Sinir Ağı Emülatörü Gerçeklemesi

Nerhun Yıldız, Evren Cesur ve Vedat Tavşanoğlu
Yıldız Teknik Üniversitesi
Elektronik ve Haberleşme Mühendisliği Bölümü
Davutpaşa, Esenler, 34220, İstanbul
e-posta: {nerhuny,ecesur,tavsanav}@yildiz.edu.tr

***Özetçe*—Bu bildiride yeni bir gerçek zamanlı Hücresel Sinir Ağı (HSA, Cellular Neural Networks, CNN) emülatörünün matematiği ve mimarisi ele alınmıştır. Tasarlanmış olan yapı biri ucuz ve diğeri gelişmiş olmak üzere iki farklı FPGA (Field Programmable Gate Array, Alanda Programlanabilir Kapı Dizisi, APKD) üzerinde gerçeklenmiştir. Gerçeklenen prototiplerin her ikisi Full–HD 1080p@60 (1920×1080 çözünürlük, 60 Hz çerçeve hızı, 124.4 Mpiksel/s piksel hızı) standardındaki hareketli görüntüler üzerinde çalışabilmektedir. Ayrıca sistemin birçok kısmı çalışma esnasında programlanabilecek veya sentezleme öncesinde konfigürasyonu yapılabilecek şekilde tasarlandığından dolayı sistemin esnekliği, ölçeklenebilirliği ve tekrar kullanılabilirliği çok yüksektir.**

I. GIRIŞ

Standart Hücresel Sinir Ağları (HSA), veya İngilizce adıyla Cellular Neural Networks (CNN), iki boyutlu bir sinir ağı yapısıdır [1]. Sürekli zamanlı gerçeklemelerde nöron hücreleri düzlemsel bir ızgaranın düğümlerine dizilmiştir ve her hücre zamansal bir analog bellek içerir. Hücreler arası bağlantılar nöron ağının ağırlık matrisini ve dolayısıyla da HSA'nın uzay zamansal dinamiğini belirler. HSA'nın çok kullanıldığı alanlar: Görüntü işleme, yapay görme uygulamaları ve kısmi türevli diferansiyel denklemlerin çözülmesi olarak sıralanabilir.

Sürekli zamanlı HSA gerçeklemeleri her ne kadar çok yüksek çerçeve hızlarında çalışabilseler de günümüze kadar gerçeklenebilmiş en yüksek çözünürlük 176×144 ile EYE–RIS'tir [2]. Çözüm olarak her ne kadar büyük görüntüler parçalar halinde ilgili yapılar üzerinde işlenebilse de gerçeklemelerin giriş ve çıkışlarının bant genişliklerinin düşük olması bu tür yapıların yavaşlamasına yol açar.

Sürekli zamanlı HSA'nın diferansiyel denklemlerindeki türev ifadeleri ayrıklaştırılarak Ayrık Zamanlı HSA'nın (AZ–HSA'nın) fark denklemleri elde edilir. Fark denklemleri de sayısal olarak çözülebildiğinden dolayı HSA yapısı sayısal sitemler üzerinde de gerçeklenebilir. Programlanabilir sayısal sistemlere DSP'ler (sayısal işaret işleyiciler), GPU'lar (grafik işlemciler) ve FPGA'ler (Field Programmable Gate Array, Alanda Programlanabilir Kapı Dizileri, APKD'ler) örnek verilebilir. Bu çalışma kapsamında yapılan gerçeklemelerde hem esnek olmaları, hem de en yüksek performansı sunmaları açısından FPGA'ler tercih edilmiş, yazım dili olarak da VHDL kullanılmıştır.

İkinci bölümde HSA'nın matematiksel modeli verilmiştir. Üçüncü bölümde tasarlanan yapının mimarisi ele alınmıştır. Dördüncü bölümde gerçeklenen prototipler anlatılmıştır. Beşinci bölümde ise sonuçlara yer verilmiştir.

II. HSA'NIN MATEMATIKSEL MODELI

$K \times L$ boyunda dikdörtgen bir yapıya sahip, m komşuluklu, konumdan bağımsız ve sürekli zamanlı bir HSA'nın hücre durum ve çıkış denklemleri

$$\dot{x}_{ij}(t) = -x_{ij}(t) + \sum_{k,l=-m}^{m} \big(a_{kl} y_{i+k\,j+l}(t) + b_{kl} u_{i+k\,j+l}\big) + z, \tag{1}$$

$$y_{ij}(t) = f(x_{ij}(t)) = 0.5\left(|x_{ij}(t)+1| - |x_{ij}(t)-1|\right), \tag{2}$$

çifti ile tanımlanabilir [3]. Burada (i,j), $i \in \{1,2,...K\}$, $j \in \{1,2,...L\}$ hücrenin uzamsal kartezyen koordinatı, $x_{ij}(t)$ hücrenin t anındaki durumu, u_{ij} hücrenin sabit değerli girişi, a_{kl} and b_{kl}, $k,l \in \{-m,...0,...m\}$, $m \in \mathbb{N}$ sırasıyla sabit değerli geri besleme ve giriş ağırlık katsayıları, z ise eşik değeridir. (1) denklemi şablon formunda yazılırsa

$$\dot{x}_{ij}(t) = -x_{ij}(t) + \boldsymbol{A} \circledast \boldsymbol{Y}_{ij}(t) + \boldsymbol{B} \circledast \boldsymbol{U}_{ij} + z \tag{3}$$

elde edilir. Burada $\circledast$ *şablon nokta çarpımı* operatörüdür ve $m = 1$ için

$$\boldsymbol{A} = \begin{bmatrix} a_{-1\,-1} & a_{-1\,0} & a_{-1\,1} \\ a_{0\,-1} & a_{00} & a_{01} \\ a_{1\,-1} & a_{10} & a_{11} \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} b_{-1\,-1} & b_{-1\,0} & b_{-1\,1} \\ b_{0\,-1} & b_{00} & b_{01} \\ b_{1\,-1} & b_{10} & b_{11} \end{bmatrix},$$

$$\boldsymbol{X}_{ij}(t) = \begin{bmatrix} x_{i-1\,j-1}(t) & x_{i-1\,j}(t) & x_{i-1\,j+1}(t) \\ x_{i\,j-1}(t) & x_{ij}(t) & x_{i\,j+1}(t) \\ x_{i+1\,j-1}(t) & x_{i+1\,j}(t) & x_{i+1\,j+1}(t) \end{bmatrix},$$

$$\boldsymbol{U}_{ij} = \begin{bmatrix} u_{i-1\,j-1} & u_{i-1\,j} & u_{i-1\,j+1} \\ u_{i\,j-1} & u_{ij} & u_{i\,j+1} \\ u_{i+1\,j-1} & u_{i+1\,j} & u_{i+1\,j+1} \end{bmatrix}.$$

şeklindedir.

(1) denklemindeki türev ifadesine ileri Euler yaklaşıklığı uygulanırsa AZ–HSA'nın fark denklemi

$$x_{ij}(n+1) = x_{ij}(n) + T_s\big(-x_{ij}(n) + \boldsymbol{A} \circledast \boldsymbol{Y}_{ij}(n) + \boldsymbol{B} \circledast \boldsymbol{U}_{ij} + z\big). \tag{4}$$

şeklinde elde edilir. Her ne kadar (4) kullanılarak bir sayısal gerçekleme yapılabilse de HSA'nın FSR (Full Signal Range, tüm sinyal aralığı) modelinin gerçeklenmesi daha kolaydır [4]. HSA'nın FSR modelinin hücre durum denklemi (4)'te $x_{ij}(n) = y_{ij}(n)$ alınarak

$$x_{ij}(n+1) = (1-T_s)y_{ij}(n) + T_s\boldsymbol{A} \circledast \boldsymbol{Y}_{ij}(n) + T_s\boldsymbol{B} \circledast \boldsymbol{U}_{ij} + T_s z,$$

şeklinde elde edilir. Bu denklem

$$x_{ij}(n+1) = \bar{\boldsymbol{A}} \circledast \boldsymbol{Y}_{ij}(n) + \bar{\boldsymbol{B}} \circledast \boldsymbol{U}_{ij} + \bar{z}, \qquad (5)$$

olarak düzenlenebilir. Burada yeni şablonlar ve eşik

$$\bar{a}_{kl} = \begin{cases} (1-T_s) + T_s a_{kl} & k,l = 0, \\ a_{kl} & \text{otherwise}, \end{cases}$$
$$\bar{b}_{kl} = T_s b_{kl}, \qquad \bar{z} = T_s z.$$

şeklinde tanımlıdır. Son olarak (2) ve (5) birleştirilirse

$$y_{ij}(n+1) = f\big(\bar{\boldsymbol{A}} \circledast \boldsymbol{Y}_{ij}(n) + \bar{\boldsymbol{B}} \circledast \boldsymbol{U}_{ij} + \bar{z}\big). \qquad (6)$$

elde edilir.

III. Gerçeklenen Yapının Mimarisi

(6) numaralı denklemde

$$g_{ij} = \bar{\boldsymbol{B}} \circledast \boldsymbol{U}_{ij} + \bar{z} \qquad (7)$$

kısmı ayrık zaman değişkeni n'den bağımsızdır ve dolayısıyla bir kerede hesaplanabilir. Öte yandan geri kalan kısım

$$y_{ij}(n+1) = f\left(\bar{\boldsymbol{A}} \circledast \boldsymbol{Y}_{ij}(n) + g_{ij}\right) \qquad (8)$$

şeklindedir ve ancak yineleme ile hesaplanabilir.

Gerçeklenen yapı [5]'te önerilen mimarinin geliştirilmesi ile elde edilmiştir [6], [7]. Buna göre (7)'de görülen sabit kısım BPU (B–template Processing Unit) adı verilen bir işlemciyle işlenirken, (8) için her biri bir iterasyon sonucunu hesaplamaktan sorumlu N sayıda APU (A–template Processing Unit) ardı ardına dizilir. Böylece toplam N adımlık bu yineleme tek bir işlemci ile döngüsel olarak N iterasyon adımında işlenmek yerine iş hattı kurularak tek adımda gerçekleştirilmiş olur.

Gerçeklenen sistemin blok diyagramı Şekil 1a'da verilmiştir. Sistemin işlem akışı sıralı paketlenmiş bir hareketli görüntünün alınması ile başlar, HSA'nın sonucunun hesaplanması ile devam eder ve hesaplanmış sonucun sıralı olarak paketlenerek dış dünyaya verilmesi ile sona erer.

FPGA üzerinde yapılmış olan gerçeklemenin basitleştirilmiş blok diyagramı Şekil 1b'de verilmiştir. Video giriş ve çıkış blokları sırasıyla hareketli görüntünün çözülmesi ve yeniden oluşturulması için tasarlanmıştır. UART seri arayüzü; şablonların, eşiğin ve diğer bazı parametrelerin çalışma esnasında programlanmasında kullanılır. HSA emülatörü ise sistemin çekirdeğini oluşturur

HSA emülasonu renkli görüntünün gri seviyeye dönüştürülmesi ile başlar (Şekil 1c). Daha sonra elde edilen gri görüntü programlanabilir bir griden siyah/beyaza dönüştürme bloğu üzerinden kullanılan şablonların görevi ve ihtiyacına göre ya dönüştürülür, ya da değiştirilmeden bir sonraki bloğa iletilir. BPU bloğu bu veriyi alarak (7) denklemini hesaplar, sonucu ilk APU'ya iletir. APU'lar ardı ardına (8)'deki yineleme sonucunu hesaplar. Son olarak gerekli görüldüğü durumlarda devreye alınan bir karşıtlık keskinleştirme bloğu ile sonucun gözlenebilirliği arttırılır.

Şekil 1c'deki APU ve BPU işlemcileri yapısal olarak birbirine çok benzediğinden dolayı xPU adı verilen ve APU veya BPU olarak programlanabilen tek bir HSA işlemcisi tasarlanmıştır (Şekil 1d). Bu blok sınır koşullarını üretmekten, girişindeki verileri tamponlamaktan ve bir şablon nokta çarpımı ile bir toplama işlemi yapmaktan sorumludur [7].

Gerçeklenen sistemde [5]'tekinin aksine merkezi bir kontrol biriminin kullanılması tercih edilmemiştir. Bunun nedeni merkezi kontrol biriminin en ufak değişikliklerden etkilenerek yeni bir tasarım süreci gerektirmesi ve dolayısıyla tasarlanmış olan diğer blokların tekrar kullanılabilirliğini bozmasıdır [6]. Bu nedenle tasarlanan tüm blokların içine satır ve çerçeve bilgisini içeren *hframe* ve *vframe* yerel kontrol işaretleri ile kontrol edilen birer yerel kontrol birimi gömülmüştür. Yerel kontrol biriminin birinci görevi *hframe* ve *vframe*'den ait olduğu bloğun iç kontrol işaretlerini üretmek, ikinci görevi ise bir sonraki bloğun ihtiyaç duyduğu *hframe* ve *vframe* işaretlerini üretmektir. Bir başka deyişle her blok kendisinden bir önceki blok tarafından kontrol edilir ve bir sonraki bloğu kontrol eder. Sonuçta herhangi bir bloğun işlem zincirine eklenmesi veya çıkartılması kontrol mantığını değiştirmez. Dolayısıyla tasarlanan mimarinin yeniden kullanılabilirliği çok yüksektir.

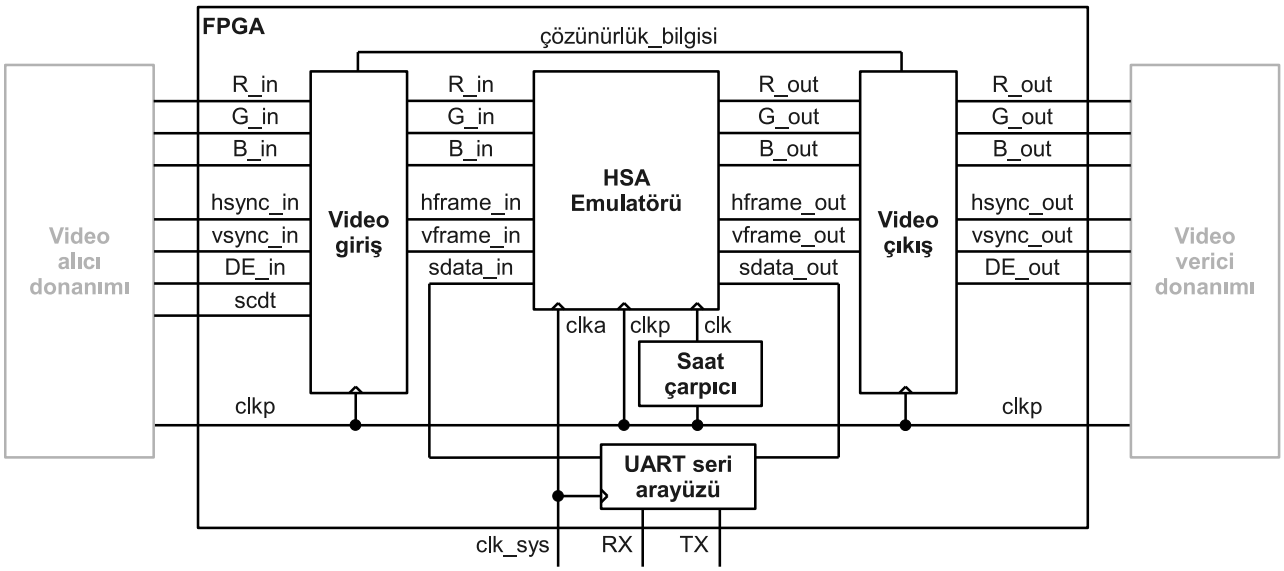
Tasarlanan mimaride sabit noktalı aritmetik kullanılmıştır. Sistemin esnekliğini ve tekrar kullanılabilirliğini arttıran bir özelliği de bit genişliklerinin, bellek derinliklerinin, vb. birçok parametrenin sentezlemeden önce değiştirilebilir olmasıdır. Ayrıca tüm şablon değerleri, eşik, baypas modu, vb. birçok parametre veya özellik çalışma esnasında programlanabilecek şekilde tasarlanmıştır. Tüm bu özellikler sistemin daha esnek ve kullanılabilir olmasını sağlamaktadır.

IV. Gerçeklenen Prototipler

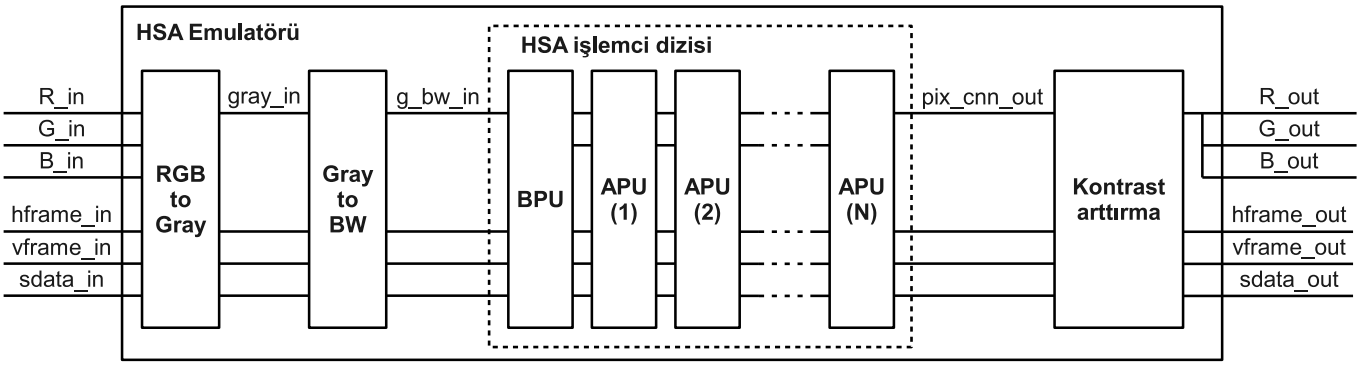
Gerçeklenmiş olan prototipte Şekil 1c'deki CNN emülatörü bloğunun iç yapısı iki HSA işlemci dizisi içerecek şekilde değiştirilmiştir (Şekil 2). Burada iki ayrı kanaldan işlenen görüntü çoğullanarak çıkışa aktarılır. Bu teknikle ekranın istenilen bir bölümü farklı, kalan bölümü farklı bir algoritma ile işlenebilir. Ayrıca istendiğinde HSA işlemci dizilerinden biri baypas edilerek ekranın bir kısmı işlenmeyebilir. Çıkış çoğullama bloğu ekrandaki bir dikdörtgenin içine birinci, dışına ise ikinci kanaldan gelen verileri aktarır. Bu dikdörtgenin köşe koordinatları programlanabildiği gibi hareket etmesi de sağlanabilir. Böylece HSA sonucunun gözlemlenebileceği bir ortam elde edilir.

[7]'ta sonuçları verilmiş olan sistemde FPGA olarak yüksek performanslı bir Altera Stratix IV GX 230

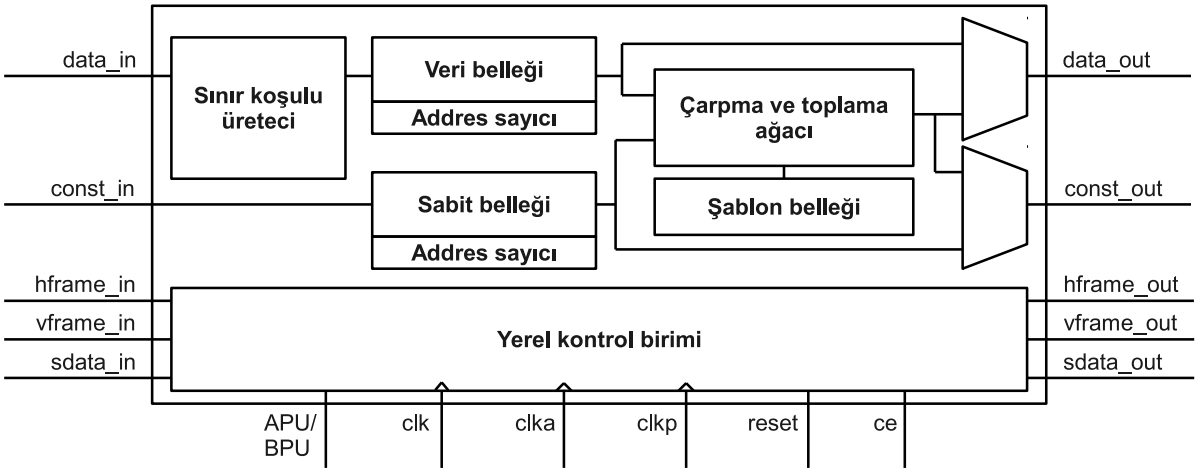
(a) Sistemin basitleştirilmiş blok diyagramı



(b) FPGA üzerinde gerçeklenen sistemin basitleştirilmiş blok diyagramı

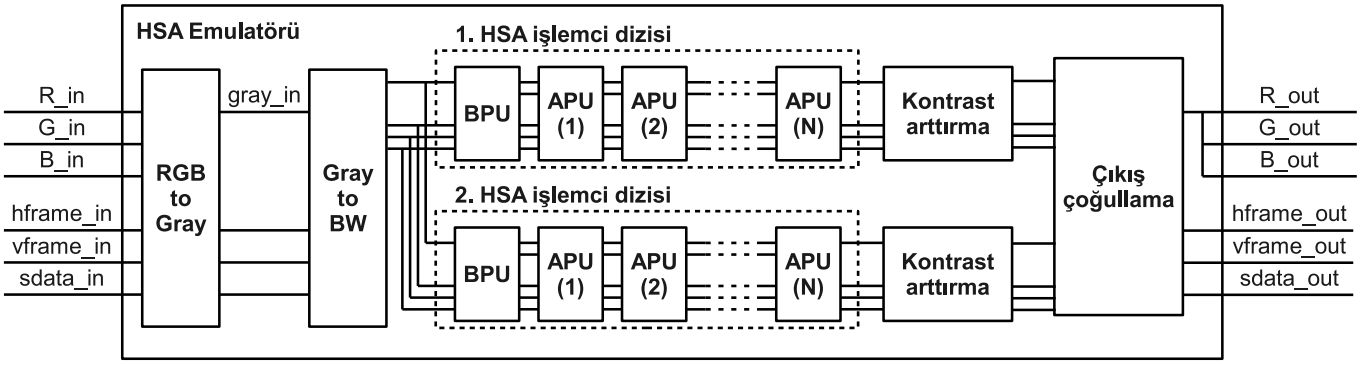


(c) *CNN Emülatörü* bloğunun basitleştirilmiş blok diyagramı



(d) APU veya BPU olarak programlanabilen xPU işlemcisinin basitleştirilmiş blok diyagramı

Şekil 1: Sistemin değişik seviyelerinin basitleştirilmiş blok diyagramları



Şekil 2: Gerçeklenen prototipteki HSA emülatörünün blok diyagramı

seçilmiştir. İlgili FPGA'in içine 150 xPU işlemcisi sığmaktadır. Her bir işlemcinin 9 çarpma ve 9 toplama işlemi yaptığı göz önüne alındığında Full–HD 1080p@60 (1920×1080 çözünürlük, 60 Hz çerçeve hızı ve 124,4 Mpiksel/s piksel hızı) için toplam saniyede 335.9 Giga çarpma ve toplama işlemi yapılmaktadır. Bu da sistemin işlem gücünü ortaya koyar.

Öte yandan aynı yapı, hiçbir ek kodlama yapılmadan, yalnızca VHDL kodlarında birkaç parametrenin değiştirilmesi ile ucuz bir Altera Cyclone III C 25 FPGA'ine gömülmüştür. Ölçekleme nedeniyle bu sisteme 8 xPU işlemcisi sığmaktadır. Ancak kenar belirleme, keskinleştirme, vb. birçok filtrenin HSA ile gerçeklenmesinde 2–3 iterasyon zaten yeterli olmaktadır. Test edilen en yüksek çözünürlük yine Full–HD 1080p@60'tır. Dolayısıyla küçültülen sistemde de yine saniyede 124,4 Mpiksel görüntü işlenebilmektedir.

V. Sonuçlar

Sonuçta yüksek performanslı ve yüksek çözünürlükteki hareketli görüntüleri işleyebilen bir HSA emülatörü FPGA üzerinde gerçeklenmiştir. Test edilen en yüksek çözünürlük 124,4 Mpiksel/s görünür piksel hızına sahip Full–HD 1080p@60'tır. Ancak burada sınırlayıcı kullanılan giriş/çıkış birimi olduğundan uygun bir giriş/çıkış birimi ile 300 Mpiksel/s mertebesindeki hızlara herhangi bir optimizasyona gerek kalmadan ulaşılabileceği öngörülmektedir.

Mimarinin kodlanmasında VHDL dili kullanılmıştır. Mimarinin birçok kısmı sentezleme öncesinde konfigürasyonu yapılabilecek veya çalışma esnasında programlanabilecek şekilde tasarlandığından dolayı sistemin esnekliği, tekrar kullanılabilirliği ve ölçeklenebilirliği çok yüksektir.

Tasarlanan yapı biri gelişmiş, bir de ucuz olmak üzere iki farklı FPGA'e gömülerek başarıyla test edilmiştir. Bu da sistemin ölçeklenebilirliğini ve esnekliğini kanıtlar. Ayrıca yerel kontrol birimlerinin kullanılması tasarıma yeni birimlerin eklenebilmesini veya mevcut birimlerin çıkartılmasını kolaylaştırır ve tasarlanan blokların tekrar kullanılabilir olmasını sağlar.

VI. Teşekkür

Bu çalışma 108E023 numaralı TÜBİTAK projesi kapsamında desteklenmiştir.

VII. Kaynakça

[1] L. Chua and L. Yang, "Cellular neural networks: theory," *Circuits and Systems, IEEE Transactions on*, vol. 35, no. 10, pp. 1257–1272, oct 1988.

[2] AnaFocus Inc., sep 2012. [Online]. Available: http://www.anafocus.com/

[3] L. Chua and T. Roska, *Cellular Neural Networks and Visual Computing: Foundation and Applications*. Cambridge University Press, 2002.

[4] S. Espejo, A. Rodriguez-Vazquez, R. Dominguez-Castro, and R. Carmona, "Convergence and stability of the FSR CNN model," in *Cellular Neural Networks and their Applications, 1994. CNNA-94., Proceedings of the Third IEEE International Workshop on*, dec 1994, pp. 411–416.

[5] K. Kayaer and V. Tavsanoglu, "A new approach to emulate CNN on FPGAs for real time video processing," in *Cellular Neural Networks and Their Applications, 2008. CNNA 2008. 11th International Workshop on*, july 2008, pp. 23–28.

[6] N. Yildiz, E. Cesur, and V. Tavsanoglu, "A new control structure for the pipelined CNN processor arrays," in *Cellular Nanoscale Networks and Their Applications (CNNA), 2010 12th International Workshop on*, feb. 2010, pp. 1–4.

[7] E. Cesur, N. Yildiz, and V. Tavsanoglu, "Architecture of the next generation real time CNN processor: RTCNNP-v2," in *Nonlinear Theory and its Applications (NOLTA), 2010 International Symposium on*, september 2010.

YSA Uygulamaları İçin FPGA Tabanlı Softmax Transfer Fonksiyonunun Gerçeklenmesi

İsmail Koyuncu
Düzce Üniversitesi
Kontrol ve Otomasyon Teknolojisi Bölümü
Uzunmustafa, 81100, Düzce
e-posta: ismailkoyuncu@duzce.edu.tr

İbrahim Şahin
Düzce Üniversitesi
Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümü, 81620, Konuralp, Düzce
eposta: ibrahimsahin@duzce.edu.tr

Özetçe— **Günümüzde yapay sinir ağları kontrol, optimizasyon, tıp, sinyal ve görüntü işleme gibi bir çok alanda kullanılmaktadır. Yapay sinir ağlarında genel olarak doğrusal ve doğrusal olmayan transfer fonksiyonları bulunmaktadır. Doğrusal olmayan transfer fonksiyonları üstel işlemler içerdiğinden bunların geniş değer aralıkları içerisinde donanımsal olarak gerçeklenmeleri oldukça zordur. Bu çalışmada doğrusal olmayan transfer fonksiyonlarından biri olan softmax transfer fonksiyonu FPGA tabanlı olarak tasarlanmıştır. Tasarım bir donanım tanımlama dili olan VHDL'de kodlanmış ve 32-bit IEEE 754-1985 kayan noktalı sayı standardı kullanılmıştır. Çalışmada 2, 3 ve 4 girişli olmak üzere 3 farklı softmax transfer fonksiyonu modellenmiştir. Modellenen transfer fonksiyonu Xilinx'in ISE 13.1 aracı kullanılarak Virtex-6 FPGA çipi için sentezlenmiş ve test edilmiştir. Test sonuçlarına göre tasarlanan transfer fonksiyonunun çalışma frekansı 266.429MHz'dir. Ayrıca tasarım bir milyon veri takımını 4.743ms gibi çok kısa bir sürede hesaplayabilmektedir.**

I. GİRİŞ

Günümüzde Yapay Sinir Ağları (YSA (Artificial Neural Networks)) pek çok alanda yaygın bir şekilde kullanılmaktadır. Bu alanlara sinyal ve görüntü işleme [1, 2] elektrik motorlarının kontrolü [3], optimizasyon [4,5] örnek olarak verilebilir. YSA'ların gerçeklenmesi birkaç farklı donanım ile yapılabilmektedir. Bunlar yazılım tabanlı çalışan bilgisayar programları ve donanımsal yapılardır. Yazılım gerçeklemeleri genellikle yeterli performans verememektedir. Donanımsal olarak YSA'ların gerçeklenmesi için literatürde Application Specific Integrated Circuits (ASIC) [6], Digital Signal Processing (DSP) [7], ve Field Programmable Gate Array (FPGA) [8-10] gibi farklı yapılar bulunmaktadır. ASIC tabanlı uygulamalarda oldukça yüksek performans elde edilmektedir. Ancak ASIC tabanlı uygulamaların en önemli dezavantajı tasarım aşamasının uzun sürmesi ve herhangi bir hata durumunda yapının tekrar kullanılamamasıdır. ASIC tabanlı seri üretimde tasarım sırasında yapılacak bir hata oldukça yüksek maliyet ve uzun zaman kaybına neden olmaktadır. YSA'lar genel yapısından dolayı paralel olarak çalışmakta ancak DSP çipleri ise seri olarak işlem yapmaktadırlar. FPGA sistemleri hem yeniden programlanabilir olmalarından dolayı esnek bir yapıya hem de paralel işlem yapabilme özelliğine sahiptirler. YSA'larda genel olarak doğrusal ve doğrusal olmayan transfer fonksiyonları olmak üzere iki çeşit transfer fonksiyonu bulunmaktadır. Doğrusal olmayan transfer fonksiyonları üstel işlemler içerdiğinden bunların donanım tabanlı gerçeklenmeleri oldukça zordur. Bu nedenle bu fonksiyonların donanım tabanlı olarak modellenmelerine ihtiyaç duyulmaktadır. Bu amaçla çalışmada doğrusal olmayan transfer fonksiyonlarından biri olan softmax transfer fonksiyonu FPGA tabanlı olarak tasarlanmıştır. Tasarım 32-bit IEEE 754-1985 kayan noktalı sayı standardı kullanılarak çok yüksek hızlı entegre devreler için donanım tanımlama dili olan VHDL (**V**ery High Speed Integrated Circuit (VHSIC) **H**ardware **D**escription **L**anguage) dilinde kodlanmıştır. Çalışmada 2 girişli, 3 girişli ve 4 girişli olmak üzere 3 farklı girişe sahip softmax transfer fonksiyonu modellenmiştir. Tasarımı yapılan transfer fonksiyonu Xilinx'in ISE 13.1 aracı kullanılarak Virtex-6 FPGA çipi için sentezlenmiş ve test edilmiştir.

Bu makalenin ikinci bölümünde FPGA çipleri, YSA'larda kullanılan transfer fonksiyonları ve softmax transfer fonksiyonu hakkında genel bilgiler verilmiştir. Üçüncü bölümde tasarımı yapılan FPGA tabanlı softmax transfer fonksiyon modülü detaylarıyla anlatılmıştır. Dördüncü bölümde yapılan test çalışmaları ve bu çalışmalardan elde edilen veriler sunulmuştur. Son bölümde ise elde edilen sonuçların bir değerlendirilmesi yapılmıştır.

II. GENEL BİLGİ

A. FPGA Çipleri

Alanda programlanabilir kapı dizileri olarak tanımlanan FPGA çipleri programlanabilir tümdevrelerdir. Mantıksal fonksiyonları gerçekleştirebilmesi amacıyla, kullanıldığı yerde programlanabilir olarak üretilirler. FPGA çipleri genel olarak mantıksal bloklar, giriş-çıkış blokları ve ara bağlantılar olmak üzere programlanabilir üç bileşenden oluşmaktadır. Kullanıcının tasarladığı mantıksal devreye göre, mantıksal bloklar, aralarındaki bağlantılar ve giriş/çıkış blokları programlanabilmektedir.

a. Yapılandırılabilir Mantıksal Bloklar (Configurable Logic Blocks (CLB)) mantıksal fonksiyonların oluşturulabildiği başvuru tablosu (Look-up table (LUT)) ve Flip-Flop'lardan oluşmaktadır. CLB'ler, kullanıcının oluşturmak istediği mantıksal devre için fonksiyonel elemanlar sağlarlar. CLB mimarisinin esnekliği ve simetrisi, uygulamaların kolaylıkla yerleştirilmesine ve gerçeklenmesine olanak tanır.

b. Giriş Çıkış Blokları (Input/Output Blocks (IOB)) çipin iç sinyal hatları ile çipin pinleri arasında programlanabilir arabirim görevini yerine getirirler. IOB'ler sayesinde FPGA

pinleri giriş, çıkış ya da çift yönlü olarak programlanabilir. FPGA çipinin türüne göre bir çipteki IOB sayısı (dolayısıyla pin sayısı) 1000'li sayılara ulaşabilmektedir.

c. Ara Bağlantılar (Interconnections) hem CLB'ler arasında hem de CLB'ler ile IOB'ler arasında bağlantıları yapılandırmada kullanılırlar. Programlanabilir olduklarından çok esnek bir yapıya sahiptirler [11,12].

B. Softmax Transfer Fonksiyonu

Y.S.A.'larda kullanılan transfer fonksiyonları doğrusal ve doğrusal olmayan transfer fonksiyonları olmak üzere iki grupta toplanabilir. Doğrusal transfer fonksiyonlarına Pure Linear (PureLin), Positive Linear (PosLin), Hard Limiting (HardLim) ve Symmetric Hard Limiting (HardLims) fonksiyonları örnek olarak verilebilir. Doğrusal olmayan transfer fonksiyonlarından bazıları ise Radial Basis (RadBas), Log-Sigmoid (LogSig), Hyperbolic Tangent Sigmoid (Tansig) ve Softmax fonksiyonlarıdır. Aşağıda bu transfer fonksiyonlarının matematiksel denklemleri (1), (2) (3) verilmiştir.

$$RasBas\ (n) = e^{-n^2} \tag{1}$$

$$LogSig\ (n) = \frac{1}{1+e^{-n}} \tag{2}$$

$$TanSig\ (n) = \frac{2}{(1+e^{(-2n)})} - 1 \tag{3}$$

Yukarıdaki eşitliklerden de görüleceği üzere bu transfer fonksiyonların tek bir giriş ve çıkışı vardır. Bu transfer fonksiyonlarından farklı olarak *n* tane giriş/çıkışa sahip olan doğrusal olmayan transfer fonksiyonlarından birisi softmax transfer fonksiyonudur. Softmax transfer fonksiyonun matematiksel denklemi (4) verilmiştir.

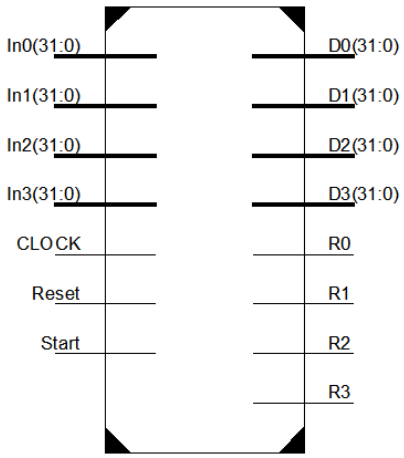
$$p_i = \frac{e^{\varsigma_i}}{\sum_{j=1}^{n} e^{\varsigma_j}} \tag{4}$$

Burada p_i ağdaki nöronun çıkışı ve j=1'den n'e kadar nöron giriş sayısıdır. ς_i ise i'nci nöronun giriş değeridir. Yukarıdaki matematiksel eşitliklerden de görüleceği üzere doğrusal olmayan transfer fonksiyonları genellikle üstel işlemler içermektedirler. Bu nedenle bu fonksiyonların geniş değer aralıklarında hassas bir şekilde donanımsal olarak hesaplanması oldukça zordur.

III. SOFTMAX TRANSFER FONKSİYONU MODÜLÜ TASARIMI

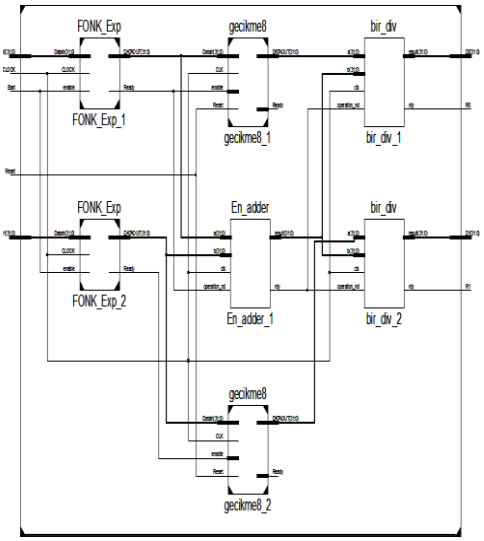
Bu çalışmada 32-bit IEEE 754-1985 kayan noktalı sayı (floating-point) standardına uygun, YSA uygulamaları için FPGA sistemleri ile birlikte kullanılabilecek donanımsal softmax transfer fonksiyonu modülü tasarlanmıştır. Modül 32-bitlik 2, 3 ve 4 adet farklı giriş sayısına sahip olacak şekilde modellenmiştir. Modül bir donanım tanımlama dili olan VHDL'de kodlanmış ve Xilinx ISE 13.1 aracı kullanılarak Virtx-6 FPGA çipi için sentezlenmiştir. Tasarımda kullanılan çarpıcı, toplayıcı, bölücü ve bazı diğer modüller Xilinx'in IP CORE Generator'ü kullanılarak oluşturulmuştur.

Şekil 1'de tasarımı yapılan 32-bitlik 4 girişli softmax transfer fonksiyonu modülünün en üst seviye blok diyagramı görülmektedir. 32-bitlik 2, 3 ve 4 girişli-çıkışlı olarak tasarımı yapılan modüllerin, karmaşıklığı engellemek amacıyla en üst seviye blok diyagramlarından sadece 4 girişli modülün blok diyagramı verilmiştir. *Reset, Basla*, *R0*, *R1*, *R2* ve *R3* sinyalleri modül zamanlaması ve modülün bağlı bulunduğu sistem ile arasındaki senkronizasyonu (hand-shaking) sağlamak için kullanılmaktadır. 32-bit kayan noktalı sayı formatında *In0*, *In1*, *In2* ve *In3* giriş sinyalleri ve *D0….D3* ise yine aynı formatta çıkış sinyalleridir.



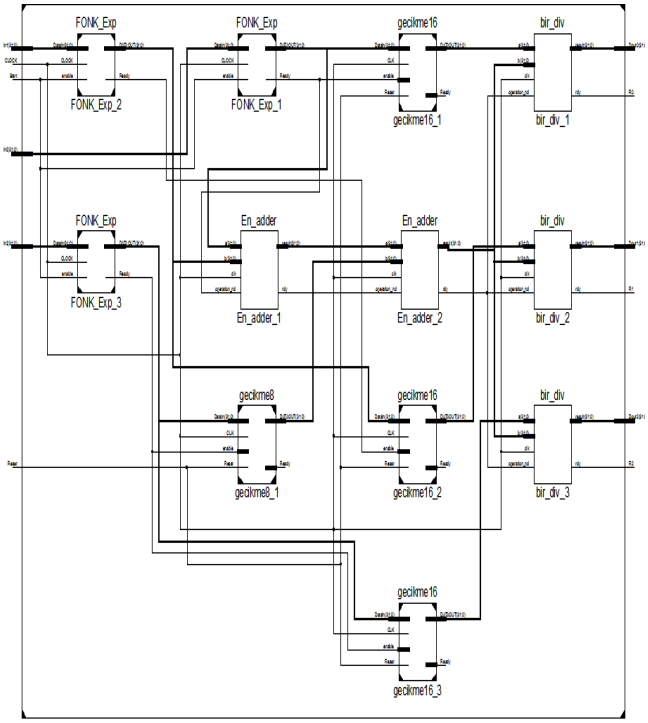
Şekil 1: 4 girişli Softmax transfer fonksiyonu ünitesi en üst seviye blok diyagramı.

Şekil 2'de 2 girişli softmax transfer fonksiyonu ünitesi ikinci seviye blok diyagramı görülmektedir. Modülde kontrol sinyalleri, 2 adet 32-bit giriş sinyali ve 2 adet 32-bit çıkış sinyali bulunmaktadır. Tasarım pipeline olarak çalışabilmektedir. Bu maçla herbir girişin *Fonk_Exp* ünitesi ile üsteli alınmakta ve bu değerler *En_adder* ünitesi ile toplanmaktadır. Sistemin ilk çıkışının elde edilebilmesi için tüm girişlerin üsteli alınarak toplanmış ve ilk girişin üsteli alınarak toplamına bölünmüş olması gerekmektedir. Bu iki işlem arasında zamanlama farkı bulunduğundan sistemin senkron bir şekilde çalışabilmesini sağlamak amacıyla 8 saat darbelik *gecikme8* üniteleri kullanılmıştır. Sistem çıkışları, *gecikme8* ünitesinden çıkan sinyallerin *En_adder* ünitesinden çıkan sinyallere bölünmesi ile elde edilmektedir.



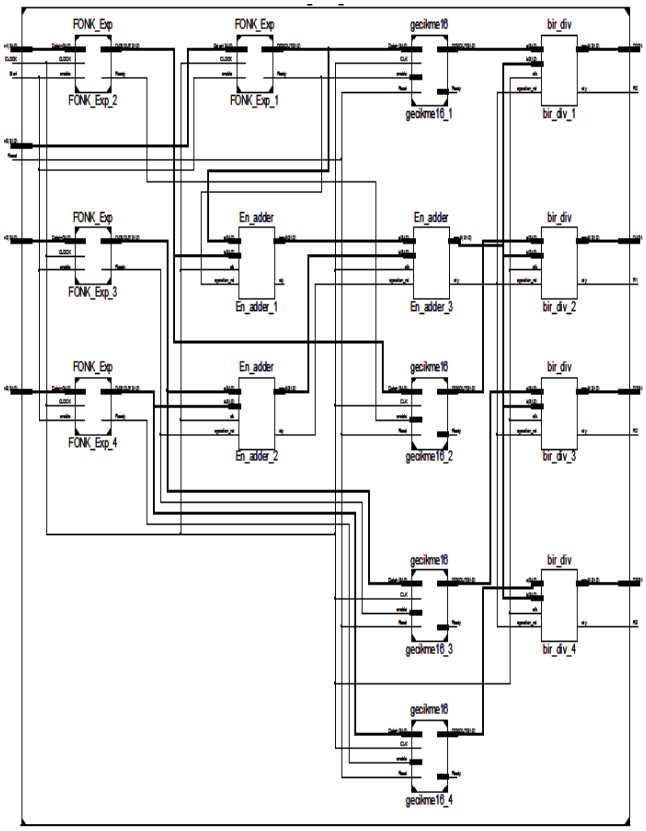
Şekil 2: 2 girişli softmax transfer fonksiyonu ünitesi ikinci seviye blok diyagramı.

Şekil 3'te 3 girişli softmax transfer fonksiyonu ünitesi ikinci seviye blok diyagramı görülmektedir. Modülde 3 adet 32-bit giriş/çıkış sinyalleri ve çeşitli kontrol sinyalleri bulunmaktadır. Tasarımda 3 tane *Fonk_Exp,* 2 tane *toplama,* 4 tane gecikme ünitesi ve çıkışta 3 tane bölme ünitesi kullanılmıştır. Toplama ünitesi 8 saat darbesinde iki adet girişi toplayabilmektedir. Sistemde 3 giriş olduğundan tüm girişlerin toplanabilmesi için 2 tane toplama ünitesine ihtiyaç duyulmuştur. Bu işlemlerin yapılabilmesi için toplam 16 saat darbesi sürenin geçmesi gerekmektedir. Bu nedenle sistemin senkronizasyonu için bölme işleminden önce *Fonk_Exp* ünitesi çıkışları 16 saat darbesi gecikme sağlayan *gecikme16* ünitesinden geçirilmiştir.



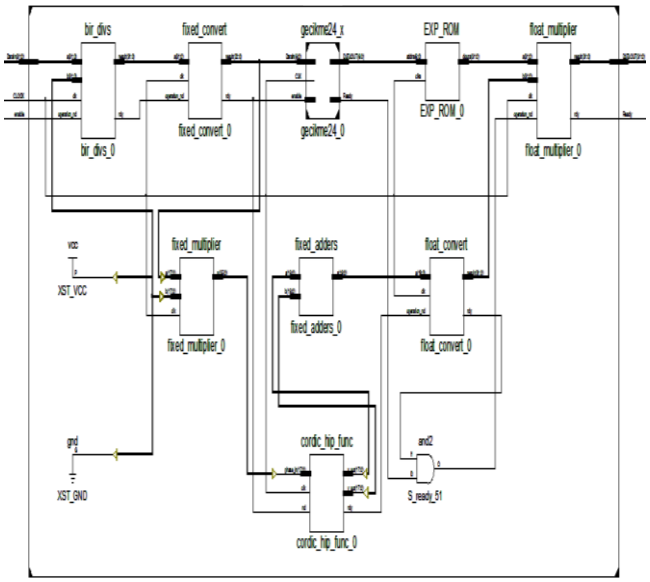
Şekil 3: 3 girişli softmax transfer fonksiyonu ünitesi ikinci seviye blok diyagramı.

Şekil 4'te 4 girişli softmax transfer fonksiyonu ünitesi ikinci seviye blok diyagramı görülmektedir. Modülde, 4 adet 32-bit giriş/çıkış ve çeşitli kontrol sinyalleri bulunmaktadır. Tasarımda 4 tane *Fonk_Exp, gecikme ve bölme* ünitesi ve 3 tane *toplama* ünitesi kullanılmıştır. Yine sistemin senkronizasyonun sağlanabilmesi amacıyla 16 saat darbesi gecikme sağlayan *gecikme16* ünitesi kullanılmıştır.



Şekil 4: 4 girişli softmax transfer fonksiyonu ünitesi ikinci seviye blok diyagramı.

Şekil 5'te 4 girişli softmax transfer fonksiyonu ünitesi içerisinde üstel işlemini gerçekleştiren *Fonk_Exp* ünitesi blok diyagramı görülmektedir. 2 ve 3 girişli softmax transfer fonksiyonu modüllerinde de aynı ünite kullanılmıştır. Ünite COordinate Rotation DIgital Computer (CORDIC) ve başvuru tablosu temelli yaklaşımlar birleştirilerek hesaplama yapmaktadır. Bu yolla ünite -48.0 ile +47.25 arasındaki herhangi bir *x* reel sayısı için 4-5 hane hassasiyetinde e^x değerini hesaplayabilmektedir. Ayrıntılı bilgi için bakınız [13].



Şekil 5: 4 girişli softmax transfer fonksiyonu ünitesi üçüncü seviye blok diyagramı.

IV. TEST SONUÇLARI

Tasarlanan 2, 3 ve 4 girişli softmax transfer fonksiyonu ünitesi, Virtex-6 FPGA çipi için sentezlenerek FPGA çip istatistiklerine bakılmış ve ünitenin maksimum saat frekansları incelenmiştir. Ünitenin belirlenen veriyi işleme süresi, ISE simülasyon programı kullanılarak elde edilmiştir.

Aşağıda şekil 6'da 4 girişli softmax transfer fonksiyonu modülünün FPGA'de gerçeklenmesinden elde edilen Xilinx ISE Simülator ekran sonuçları verilmiştir. Tasarımda 32-bit kayan noktalı sayı standardı kullanılmıştır. Ancak Xilinx ISE Simülator sonuçlarının daha kolay incelenebilmesi amacıyla şekil 6'da hexadecimal notasyonunda gösterilmiştir. 2 girişli modül ilk sonuçları 81 saat darbesi sonunda, 3 ve 4 girişli modüller ise 90 saat darbesi sonunda ilk sonuçları üretmişlerdir. 2 girişli için 81, 3 ve 4 girişli için 90 saat darbesinden önce modülün çıkışlarından herhangi bir çıkış alınamamaktadır. Bundan sonra her saat darbesinde modül çıkışlarından istenilen değerler alınabilmekte ve modül pipeline olarak çalışabilmektedir.

Şekil 6: 4 girişli softmax transfer fonksiyonu ünitesi Xilinx ISE Simülator sonuçları.

Tablo 1'de sentezleme işleminin ardından yapılan yerleştirme (place & route) sonrasında elde edilen FPGA çip istatistikleri verilmiştir. 2, 3 ve 4 girişli tasarımın minimum darbe periyodu 3.753ns olduğundan bütün tasarımların maksimum frekansı 266.429MHz'dir. Bu sentez sonuçları Xilinx Virtex-6 çip ailesinin en küçük çiplerinden biri olan XC6VCX75T aracında yapılmıştır. Virtex-6 çip ailesinin daha büyük çipleri veya Virtex-7 ailesinin çiplerinden birisi kullanıldığında oldukça düşük çip istatistikleri elde edilecektir.

Tablo 1: FPGA çip istatistikleri.

Softmax Giriş-Çıkış Sayısı	Slice Reg. Sayısı / %	LUTs Sayısı / %	Slice FFs Sayısı / %	Bounded IOBs Sayısı / %	Max Saat Hızı (MHz)
2	8643 / 9	8508 / 18	2132 / 19	133 / 36	266.429
3	12351 / 13	12806 / 27	3278 / 19	198 / 55	266.429
4	16507 / 17	17851 / 38	4423 / 20	263 / 73	266.429

Tablo 2'de tasarımı yapılan modüllerin 100, 10000 ve 1000000 veriyi işleme süreleri verilmiştir. İki girişli modülün tasarımında daha az yapı kullanıldığından pipeline mesafesi kısalmakta ve sonuçları daha kısa süre içerisinde üretmektedir. 3 ve 4 girişli modüllerin verileri işleme sürelerinin aynı olmasının nedeni ise pipeline gecikmelerinin ve çalışma frekansının aynı olmasından kaynaklanmaktadır.

Tablo 2: Tasarlanan modüllerin verileri işleme süreleri.

Veri Sayısı	2 Girişli (µs)	3 Girişli (µs)	4 Girişli (µs)
100	0,4563	0,4653	0,4653
10000	4,644	4,743	4,743
1000000	4644,081	4743,09	4743,09

V. SONUÇLAR

FPGA çiplerinin kapasitelerinin ve hızlarının artması bu çiplerin pek çok alanda kullanılabilirliğini arttırmıştır. Bu uygulama alanlarından birisi de YSA'lardır. YSA'larda doğrusal ve doğrusal olmayan transfer fonksiyonları bulunmaktadır. Doğrusal olmayan transfer fonksiyonları genellikle üstel işlemleri içerdiğinden bunların geniş değer aralıkları içerisinde donanımsal olarak gerçeklenmeleri oldukça zor olmaktadır. Bu çalışmada doğrusal olmayan transfer fonksiyonlarından biri olan softmax transfer fonksiyonu FPGA tabanlı olarak tasarlanmıştır. Tasarım bir donanım tanımlala dili olan VHDL'de kodlanmıştır. Tasarımda, Virtex-6 FPGA çipi ve 32-bit IEEE 754-1985 kayan noktalı sayı standardı kullanılmıştır. Çalışmada 2, 3 ve 4 giriş-çıkışlı olmak üzere 3 tane softmax transfer fonksiyonu ünitesi modellenmiştir. Yazılan kod Xilinx'in ISE 13.1 aracı kullanılarak Virtex-6 FPGA çipi için sentezlenmiş ve test edilmiştir. Sonuçlara göre tasarlanan ünitelerin hepsinin çalışma frekansı 266.429MHz'dir. Ayrıca tasarımı yapılan 2 girişli softmax transfer fonksiyon ünitesi bir milyon veri takımını 4.644 ms'de, 3 ve 4 girişli softmax transfer fonksiyon üniteleri ise aynı sayıdaki veri takımını 4.743ms gibi çok kısa bir sürede

hesaplayabilmektedir. İleriki çalışmalarda sistemin boyutunu azaltmak amacıyla çeşitli optimizasyon çalışmaları yapılabilir ve ünitenin 64-bit versiyonu tasarlanabilir. Ayrıca diğer doğrusal olmayan transfer fonksiyonları FPGA'de modellenerek sentezlenebilir.

VI. KAYNAKÇA

[1] Ö. Polat, T. Yıldırım, "FPGA implementation of a General Regression Neural Network: An embedded pattern classification system", Digital Signal Processing, vol. 20, pp. 881–886, 2010.

[2] J. Huang, J. Lee, Y. Ge; , "An array-based scalable architecture for DCT computations in video coding," Neural Networks and Signal Processing, 2008 International Conference on , vol., no., pp.451-455, 7-11 June 2008

[3] Q. N. Le, J. W. Jeon, "Neural-Network-Based Low-Speed-Damping Controller for Stepper Motor With an FPGA", IEEE Transactions on Industrial Electronics, vol. 57, no. 9, 2010.

[4] F. J. Lin, Y. C. Hung, "FPGA-Based Elman Neural Network Control System for Linear Ultrasonic Motor", IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 56, no. 1, 2009.

[5] C. J. Lin, H. M. Tsai, "FPGA implementation of a wavelet neural network with particle swarm optimization learning", Mathematical and Computer Modelling, vol. 47, no. 9-10, pp. 982-996, 2008.

[6] O. Banimelhem, R.B. Hani, "Neural network based optimization of CMOS transistor sizing for leakage power minimization," Innovations in Information Technology (IIT), 2012 International Conference on , vol., no., pp.167-172, 18-20 March 2012.

[7] R. Rieger, S. Deng, S, "Double-Differential Recording and AGC Using Microcontrolled Variable Gain ASIC," Neural Systems and Rehabilitation Engineering, IEEE Transactions on , vol.PP, no.99, pp.1, 2012.

[8] N. Kim, N. Kehtarnavaz, M. B.Yeary, S. Thornton, "DSP-Based Hierarchical Neural Network Modulation Signal Classification", IEEE Transactions on Neural Networks, vol. 14, no. 5, 2003.

[9] B. Yu, R. Chan, T. Mak, Y. Sun, C. Poon, C, "On-Chip Systolic Networks for Real-Time Tracking of Pairwise Correlations between Neurons in a Large-Scale Network," Biomedical Engineering, IEEE Transactions on , vol.PP, no.99, pp.1, 2012.

[10] I. Sahin, I. Koyuncu, "Design and Implementation of Neural Networks Neurons with RadBas, LogSig, and TanSig Activation Functions on FPGA", Electronics and Electrical Engineering", no. 5(121), 2012.

[11] I. Sahin, "A 32-bit floating-point module design for 3D graphic transformations", Scientific Research and Essays, vol. 5 (20), pp. 3070-3081, 2010.

[12] M. Papadonikolakis, C. Bouganis, "Novel Cascade FPGA Accelerator for Support Vector Machines Classification," Neural Networks and Learning Systems, IEEE Transactions on , vol.23, no.7, pp.1040-1052, July 2012.

[13] I. Koyuncu, I. Sahin, "A CORDIC Based ex Calculator Unit Design for FPGA Chips", e-Journal of New World Sciences Academy, vol. 6, no 4, 2011.

Register Transfer Level (RTL) Tasarım

İlker Eryılmaz

Ericsson Microelectronics Design Center (EMDC)
İTÜ Teknokent ARI2 B Blok
Maslak, İstanbul
ilker.eryilmaz@ericsson.com

Özetçe

- HDL Tasarım nedir?
- RTL tasarım kılavuzu
- Saat işaretleri arası geçiş
- Proje kurulumu ve hiyerarşisi
- Üretkenlik ve tekrar kullanılabilirlik
- Hata izleme ve ayıklama

İlker Eryılmaz Hakkında
İlker Eryılmaz, 1997'de İTÜ Elektronik ve Haberleşme bölümü tümleşik devre tasarım branşından mezun oldu. 1998'deUniversity of Southampton'da (İngiltere) VLSI üzerine yüksek lisans eğitimini tamamladı ve Alcatel Microeletronics İstanbul Tasarım Merkezi'nde sayısal tasarım mühendisi olarak işe başladı. Sayısal, karma işaret ve analog devre tasarımı üzerine çalıştı. 2003 yılında ST Microelectronics'de (Belçika) kablosuz iletişim teknolojileri üzerine çalışmaya başladı. Daha sonra sayısal tasarım konusunda danışmanlık hizmeti verdi. 2011'den beri Ericsson Microelectronics Design Center (EMDC) bünyesinde IP geliştirme, sistem tasarımı ve proje yöneticiligi konularında çalışmaya devam etmektedir.

Gömülü Video Sistemleri: Mimariler ve Tasarım Yaklaşımları

Hüseyin Atik, Yüksel Serdar

Aselsan A.Ş.
Mikroelektronik Güdüm ve Elektro-Optik Grub
Aselsan Akyurt Tesisleri, Ankara
hatik@aselsan.com.tr serdar@mgeo.aselsan.com.tr

Özetçe

ASELSAN, askeri gömülü sistem teknolojilerinde uzun yıllara dayanan tecrübesiyle uzmanlaşmış önder bir kuruluştur. ASELSAN'da geliştirilen ve gömülü video teknolojileri içeren ürünlere ait çeşitli örnek mimariler ve fonksiyonlar sunulacaktır. Bu ürünler, Görüntüleme Sistemleri ve Görev Sistemleri olmak üzere iki ana grupta toplanabilir. Görüntüleme Sistemlerini oluşturan video yakalama, sinyal işleme ve ekran çıkış devreleri örnek mimariler ve devre elemanları incelenerek anlatılacak, kullanılan geliştirme araçları ve tasarım yöntemlerinden bahsedilecektir. Görev Sistemleri, merkezi bir Görev Kontrol Bilgisayarı'na bağlı video kaynak cihazları, video görüntüleme/kayıt cihazları ve diğer aviyonik cihazlardan oluşmaktadır. Bu cihazlarda kullanılan video işleme, iyileştirme ve grafik oluşturma yöntemleri, kullanılan geliştirme araçları ve tasarım yaklaşımlarından bahsedilecektir.

Hüseyin Atik Hakkında
Hüseyin Atik, Hacettepe Üniversitesi Elektrik-Elektronik Mühendisliği bölümünden 1999 yılında mezun olmuştur. Aynı yıl ASELSAN Mikro-Elektronik, Güdüm ve Elektro-Optik Grubu Elektronik Tasarım Müdürlüğü'nde çalışmaya başlamıştır ve halen Sayısal Tasarım Lideri olarak görev yapmaktadır. Çalıştığı konular video işleme, grafik ve işlemci kartları ve FPGA tasarımları, termal görüntüleme ve aviyonik sistemler tasarımları ve bilgisayar ağı sistemleridir.

Yüksel Serdar Hakkında
Yüksel Serdar, Orta Doğu Teknik Üniversitesi Elektrik-Elektronik Mühendisliği bölümünden 1994 yılında mezun olmuş ve 1998 yılında aynı üniversiteden yüksek lisans derecesi almıştır. 1994 yılında ASELSAN Mikro-Elektronik, Güdüm ve Elektro-Optik Grubu Termal Sistemler Tasarım Müdürlüğü'nde sayısal tasarım mühendisi olarak çalışmaya başlamıştır. 2004 yılında Elektronik Tasarım Müdürlüğü'nde Kıdemli Uzman Mühendis olarak çalışmaya başlamış olup, Nisan-2011 tarihinden itibaren Elektronik Tasarım Müdürü olarak görev yapmaktadır. Çalıştığı konular sayısal gömülü kart ve FPGA tasarımları, termal görüntüleme ve aviyonik sistemler tasarımları, sistem mühendisliği ve proje yönetimidir.

IP ve Kırmıküstü Sistem Doğrulaması

Gürbey Fıçı

Ericsson Microelectronics Design Center (EMDC)
İTÜ Teknokent ARI2 B Blok
Maslak, İstanbul
gurbey.fici@ericsson.com

Özetçe

- İşlevsel doğrulama ve doğrulama adımları
- Tipik bir doğrulama projesinin akışı
- Dinamik ve statik doğrulama
- Kısıtlanmış rasgele doğrulama ve UVM
- UVM test ortamı
- Donanımla hızlandırılmış doğrulama

Gürbey Fıçı Hakkında
Gürbey Fıçı, 2009 yılında Bahçeşehir Üniversitesi Elektrik-Elektronik Mühendisliği bölümünden mezun oldu. Lisans tezini ST-Ericsson İstanbul Tasarım Merkezi'nde FPGA tabanli SOC-DFT doğrulamaları üzerine yaptı. Aynı yıl ST-Ericsson İstanbul Sayısal Tasarım takımına Doğrulama Mühendisi olarak katıldı. WLAN tümdevrelerinin fiziksel katmanlarının OVM ve System Verilog ile işlevsel doğrulaması üzerine çalıştı. 2011 yılında tez danışmanı Fatih Uğurdağ ile ortaklaşa yayınladıkları "Row and Column Compression for High-Performance Multiplication on FPGAs" adlı makalesi IEEE East-West Design & Test Sempozyumunda 'outstanding paper' ödülü kazandı. Halen Ericsson Microelectronics Design Center (EMDC) bünyesinde IP ve kırmıküstü sistemlerin işlevsel doğrulaması üzerine çalışmaktadır.

MATLAB & Simulink Algoritmalarının FPGA Üzerinde Gerçeklenmesi

Erman Üret

Figes A.Ş.
ODTÜ Teknokent Silikon Blok
Ankara
erman.uret@figes.com.tr

Özetçe

Mathworks, MATLAB ve Simulink algoritmalarınızı DSP, Mikroişlemci, FPGA gibi yapılar içeren birçok gömülü sisteme aktarabilmeniz için gelişmiş araçlar sağlamaktadır. Bu sunumda, günümüzde kullanımı hızla artmakta olan FPGA yapılarına yönelik olarak geliştirilen, kod dönüştürme ve doğrulama araçlarının kullanımı adım adım detaylı bir şekilde anlatılacaktır. MathWorks'un HDL kod oluşturma aracı sayesinde Model Tabanlı Tasarım metodolojisi ile MATLAB & Simulink ortamında geliştirilen algoritmalar, otomatik olarak, kolay bir şekilde okunabilir HDL kodlarına dönüştürülmektedir. Bu dönüşüm sırasında birçok optimizasyon seçeneği kullanıcılara sunulmaktadır. Doğrulama araçlarının yardımıyla sağlanan eş zamanlı HDL-Model simülasyonu ve FIL (FPGA in the loop) simülasyonu ile tek bir ortamdan algoritma geliştirme, doğrulama ve gerçekleme işlemlerinizi yapabilmektesiniz.

Erman Üret Hakkında

Erman Üret, Anadolu Üniversitesi Elektrik-Elektronik Mühendisliği Bölümü'nden mezun oldu. Çizgi-Elektronik bünyesinde 2.5 yıl kadar FPGA teknik destek ve uygulama mühendisi olarak çalıştı. Bu sürede uzaktan erişimli FPGA laboratuvarı, CPU Turkey yarışması ve birçok eğitim materyalinin hazırlanmasına katkıda bulundu. Birçok kez genel katılımlı ve özel kuruluşlara FPGA eğitimi verdi. 1 yılı aşkın bir süredir FİGES A.Ş. bünyesinde gömülü sistemlerden sorumlu uygulama mühendisi olarak görev almaktadır. Uzmanlık alanları dijital sistemler, bilgisayar mimarisi, microprocessor tasarımı ve gömülü sistemlerdir. Halen aynı üniversitede yüksek lisansına "FPGA Üzerinde Görüntü Sıkıştırma" tezi ile devam etmektedir.

Kızılötesi Kameralar için Süper Çözünürlüğün FPGA Uygulaması

Mehmet AKTUKMAK
Elektronik Tasarım Müdürlüğü
MGEO Grubu ASELSAN A.Ş.
Akyurt, 06750, Ankara
e-posta: maktukmak@aselsan.com.tr

Uğur HALICI
Orta Doğu Teknik Üniversitesi
Elektrik ve Elektronik Müh. Bölümü
Çankaya, 06800, Ankara
e-posta: halici@metu.edu.tr

Özetçe—**Kızılötesi kameraların görüntü çözünürlükleri düşük olmasından dolayı, bu tip kamera görüntülerine uygulanan görüntü iyileştirme yöntemleri büyük önem kazanmaktadır. Görüntü işleme tekniklerinden süper çözünürlük yöntemi ile görüntü çözünürlüğü artırılabilmektedir. Mevcut olan süper çözünürlük algoritmalarının çoğunluğunun gerçek zamanlı uygulanması mümkün değildir. Bu makalede, FPGA üzerinde uyarlamalı filtre teorisini kullanan bir süper çözünürlük algoritmasının, kızılötesi kameralı gerçek zamanlı sistemde uygulanması üzerinde durulmuştur.**

I. GİRİŞ

Süper çözünürlük, düşük çözünürlüklü görüntüler kullanılmasıyla yüksek çözünürlüklü görüntüler oluşturmak için kullanılan bir sinyal işleme tekniğidir. Düşük çözünürlüklü kameralardan bu teknik ile yüksek çözünürlüklü videolar elde etmek mümkündür. Yüksek çözünürlüklü kameraların kullanılması yerine düşük çözünürlüklü kameralar ve bu teknik kullanılarak maliyeti düşük bir çözüm elde edilmiş olur. Süper çözünürlük algoritmaları için piksel altı seviyede göreceli global hareket farklılıklarına sahip düşük çözünürlüklü görüntüler gerekmektedir. Görüntüler arasındaki global hareket, global hareket kestirme yöntemleri ile piksel altı seviyede kestirilmektedir. Daha sonra bu hareket bilgileri ve düşük çözünürlüklü görüntüler kullanılarak yüksek çözünürlüklü görüntü oluşturulmaktadır. Gerçek zamanlı bir sistemde düşük çözünürlüklü bir videonun iyileştirilmesi amaçlanmaktadır. Süper çözünürlük tekniği için farklı global hareket kestirme [1] ve yüksek çözünürlüklü görüntü oluşturma yöntemleri [2] kullanılmaktadır, fakat bu algoritmaların birçoğu yüksek işlem gücü ve süresi gerektirdiği için gerçek zamanlı sistemlere uygulanamamaktadır. Gerçek zamanlı sistemlerde bir video çerçeve süresi içerisinde, ardışık iki çerçeve arasındaki global hareket kestirme işleminin tamamlanması gerekmektedir. Bu işlemin paralelinde global hareket bilgileri ve düşük çözünürlüklü çerçeveler kullanılarak yüksek çözünürlüklü çerçeve oluşturma işleminin de bitirilmesi gerekmektedir. Bu tür paralel işlem yükü fazla ve süre kısıtlaması olan işlemler için en uygun donanım mimarisi FPGA tarafından sağlanmaktadır.

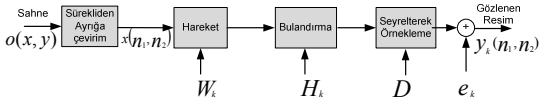
Bu makalede, görüntü stabilizasyonu için kullanılan bir global hareket kestirme algoritmasının [3] FPGA üzerinde uygulaması yapılmıştır. Bu algoritma düşük işlem gereksinimine sahip olmaktadır. Hareket modeli sadece x ve y eksenindeki kayma miktarı olarak sınırlandırılmıştır.

Literatürde gerçek zamanlı sistemlerde yüksek çözünürlüklü görüntü oluşturma işleri daha önce yapılmıştır. VLIW ve ARM işlemciler kullanılarak bir biçimli olmayan örnekleme (non-uniform interpolation) metodunu temel alan bir algoritma gerçek zamanlı olarak uygulanmıştır [4]. Daha sonra FPGA kullanılarak MAP (Maximum a Posteriori) tabanlı bir algoritmanın uygulanması önerilmiştir [5]. Bir başka FPGA uygulaması da, IBP (Iterative Back Projection) yöntemi kullanılarak yapılmıştır [6]. Gerçek zamanlı sistemlerde uygulanmış bu algoritmalar birden fazla düşük çözünürlüklü görüntünün RAM tipi belleklerde saklanmasına ihtiyaç duymaktadır. Bu makalede uygulanan yöntem [7], sadece bir adet düşük çözünürlüklü görüntünün saklanması ihtiyacı gerektirdiği için daha az bellek kullanımını sağlamaktadır.

II. TEORİK ALTYAPI

A. Gözlem Modeli

Düşük çözünürlüklü resimlerin oluşum modeli Şekil 1'de gösterilmektedir. Gerçek sahnenin, örtüşme ve NYF (Nokta Yayılım Fonksiyonu) bulandırması olmadan örneklenmiş hali, süper çözünürlük algoritmalarının ulaşmaya çalıştığı yüksek çözünürlüklü resimdir. Modele göre bu resim, kameranın veya sahnenin global hareketlerinden dolayı, her yakalanan resimde göreceli global hareket farklılıklarına sebep olmaktadır. Oluşan bu hareket farklılıklarına sahip yüksek çözünürlüklü resimler, kameranın sensör veya optik özelliklerinden kaynaklanan bir NYF ile bulandırılmaktadır. Daha sonra bu resimler seyrelterek örneklenmekte ve üzerine gürültü eklenip gözlemlenen düşük çözünürlüklü resimler elde edilmektedir. Denklem 1, p tane düşük çözünürlüklü resmin oluşum formülüdür.



Şekil 1 Gözlem Modeli

$$y_k = DH_kW_kx + e_k \qquad k = 1, \dots, p \tag{1}$$

Denklem 1'de y_k düşük çözünürlüklü resmi, x elde edilmek istenen yüksek çözünürlüklü resmi, D seyrelterek örnekleme matrisini, H NYF matrisini, W global hareket matrisini, e_k düşük çözünürlüklü resimlerdeki sıfır ortalamalı Gauss gürültüsünü göstermektedir.

B. Global Hareket Kestirme

Global hareket kestirme için uygulanan algoritma [3], izdüşüm tabanlı bir yöntemdir. Giriş resimlerinin piksel değerlerinin yatay ve dikey eksende izdüşümleri aşağıdaki formüllerle hesaplanmaktadır.

$$P'_Y(x) = \sum_{y=1}^{M} F(x,y), \quad for\ x = 1 \cdots N \tag{2}$$

$$P'_D(y) = \sum_{x=1}^{N} F(x,y), \quad for\ y = 1 \cdots M \tag{3}$$

Hesaplanan izdüşüm vektörlerinin ortalama değerleri çıkarılması ile yöntem, resimler arasındaki kontrast değişimlerine karşı güçlendirilir. Aşağıda ortalama değer çıkarma işlemleri görülmektedir.

$$\overline{P'_Y} = \sum_{x=1}^{N} P'_Y(x) \Big/ N \tag{4}$$

$$\overline{P'_D} = \sum_{y=1}^{M} P'_D(y) \Big/ M$$

$$P_Y(x) = P'_Y(x) - \overline{P'_Y}$$

$$P_D(y) = P'_D(y) - \overline{P'_D}$$

İki resim arasındaki global hareket kestirim işlemini yapmak amacıyla her resim için bu vektörler hesaplanmaktadır. Ardından bir vektör sabit tutulup diğer vektör üzerinde kaydırılarak, vektörler arasındaki farkların toplamı (SAD: Sum of absolute differences) hesaplanmaktadır. Referans görüntü "ref" ve aktif görüntü "cur" arasındaki yatay izdüşüm vektörlerinin SAD değerleri aşağıdaki denkleme göre hesaplanmaktadır.

$$SAD_Y(m) = \sum_{k=S_{max}}^{N-S_{max}} \left| P_{Y,ref}(k+m) - P_{Y,cur}(k) \right|, \qquad m = -S_{max} \cdots S_{max} \tag{5}$$

Denklemdeki S_{max} bir sistem parametresidir ve beklenen maksimum kayma miktarı ile bağlantılıdır. Minimum SAD değerine sahip m indeksi, yatay kayma miktarını göstermektedir. Aynı işlemler dikey izdüşüm vektörleri için yapılarak, dikey kayma miktarı da hesaplanmaktadır.

Bu yöntem içinde izdüşüm vektörleri bir ara değerleme yöntemi kullanılarak üst örneklenmekte ve bu işlem sayesinde piksel altı seviyede kayma değeri hesaplanmasına olanak sağlanmaktadır.

C. Yüksek Çözünürlüklü Görüntü Oluşturma

Bu makalede kullanılan yüksek çözünürlüklü görüntü oluşturma algoritması uyarlamalı filtre teorisini kullanmaktadır [7]. Bu yöntem, resimlerin zaman içerisindeki sürekli akışını göz önünde bulundurmaktadır. Algoritmanın yüksek çözünürlüklü görüntü oluşturmada kullandığı döngü denklemi aşağıda verilmiştir.

$$\hat{x}^{j+1}(n) = \hat{x}^{j}(n) + \lambda H^T D^T \left[y(n) - DH\hat{x}^{j}(n) \right] - \beta \mathrm{T}^{\mathrm{T}} \mathrm{T} \hat{x}^{j}(n) \tag{6}$$

Denklemde j döngü indeksini, n mevcut çerçeve süresini, T düzenlileştirme için kullanılan matrisi, λ ve β ise döngüdeki güncelleme katsayılarını göstermektedir [7].

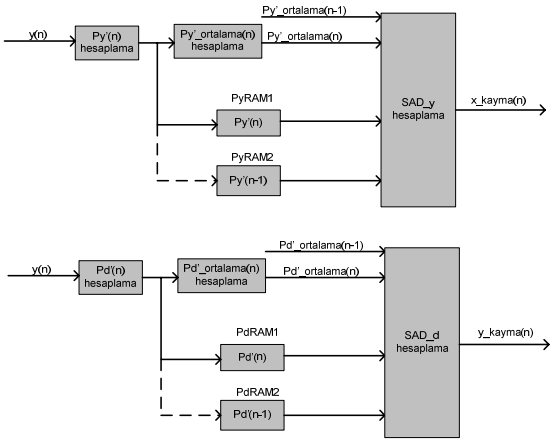
Yüksek çözünürlüklü çerçeve oluşturulurken sadece son gelen düşük çözünürlüklü çerçeve kullanılmaktadır. Bir sonraki yüksek çözünürlüklü çerçeve için başlangıç resmi, bir önceki oluşturulan yüksek çözünürlüklü çerçevenin hareket farklılığı kullanılarak elde edilmektedir. Bu işlemde kullanılacak hareket bilgileri, global hareket kestirme algoritmasının sonuçları olmaktadır. Aşağıda yüksek çözünürlüklü çerçeve için başlangıç resminin elde edilmesi görülmektedir.

$$\hat{x}^0(n) = W(n,1)\hat{x}^k(n-1) \tag{7}$$

III. FPGA UYGULAMASI

A. Global Hareket Kestirme FPGA Yapısı

FPGA'nın paralel işlem kabiliyeti sayesinde, global hareket kestirme işlemi, x ve y eksenindeki kayma miktarını hesaplama işlemleri şeklinde ayrılmaktadır. Bu işlemler paralel olarak yapılmaktadır. Şekil 2'de tasarlanan blok diyagram görülmektedir. Üst akış x yönündeki kayma miktarını, alt akış y yönündeki kayma miktarını hesaplamaktadır.

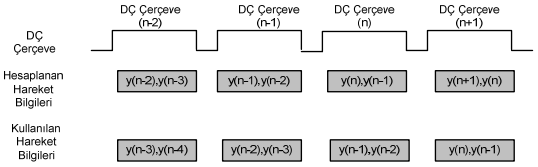


Şekil 2 Global Hareket Kestirme FPGA Yapısı

Her bir akış için iki farklı RAM tipi bellek alanı kullanılmaktadır. Bu alanlar izdüşüm vektörlerini saklamak amacı ile kullanılmaktadır. Üst akışta kullanılan PyRAM1 ile adlandırılan FPGA içi bellek alanında aktif çerçeve süresi, PyRAM2 ile adlandırılan alanda ise bir önceki çerçeve süresi içerisinde hesaplanan yatay izdüşüm vektörü saklanmaktadır. Vektörler hesaplanıp ilgili alanlara yazılırken aynı zamanda vektörlerin ortalama değerleri de hesaplanmaktadır. Bir önceki çerçevede hesaplanmış vektör ortalama değerleri de kütüklerde saklanmaktadır. Daha sonra, ortalama değerler ve bellek alanlarında tutulan izdüşüm vektörleri kullanılarak, kayma değerleri hesaplanmaktadır. Yeni bir çerçeve süresinin başlaması ile bellek alanları kendi aralarında yer değiştirmektedir.

Bu yöntemde mevcut çerçeve ile bir önceki çerçeve arasındaki global hareket bilgileri, mevcut çerçeve süresi içerisinde hesaplanmaktadır. Yüksek çözünürlüklü resim

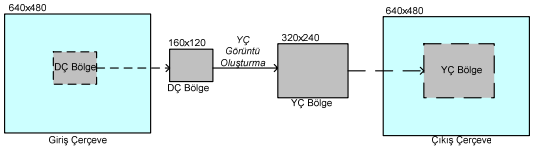
oluşturma algoritması Şekil 3'te gösterildiği gibi mevcut ile bir önceki çerçeve arasındaki global hareket bilgilerine bir sonraki çerçeve süresi içerisinde ihtiyaç duyar. Bu yüzden hesaplanan değerler kütüklerde yeni çerçeve süresi boyunca saklanmaktadır.



Şekil 3 Hesaplanan ve Kullanılan Hareket Bilgileri

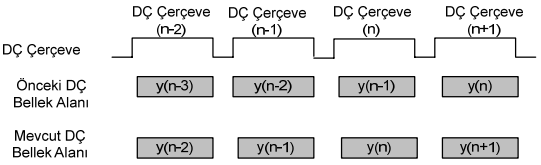
B. Yüksek Çözünürlüklü Görüntü Oluşturma FPGA Yapısı

Mevcut yüksek çözünürlüklü görüntü oluşturma algoritmalarının birçoğu döngüsel yöntem ile çalışmaktadır. Bir çerçeve süresi içerisinde yüksek çözünürlüklü çerçeve oluşturmak için birden çok döngüye gerek duyulmaktadır. Çözünürlük arttıkça bir çerçeve süresi içerisinde uygulanabilecek döngü sayısı düşmektedir. Bu çalışmada kullanılan sistemde video çözünürlüğü 640x480'dir. Görüntü formatı binişmeli olmakla birlikte çerçeve sıklığı 30 Hz'dir. Sistem çözünürlüğünün değiştirilmemesi gerekliliği ve bellek alanı kısıtlamaları dolayısıyla uygulamada giriş videosunun belli bir bölgesi seçilip, o bölgenin görüntü çözünürlüğü artırılmaktadır. Bu seçilen bölgeye düşük çözünürlüklü (DÇ) bölge ismi verilmiştir. DÇ bölgenin çözünürlüğü artırıldığında oluşan resme ise yüksek çözünürlüklü (YÇ) bölge ismi verilmiştir. Uygulamada çözünürlük yükseltme faktörü 2 olarak seçilmiştir. DÇ bölgenin çözünürlüğü 160x120 olduğundaki çözünürlük durumları Şekil 4'te gösterilmiştir.

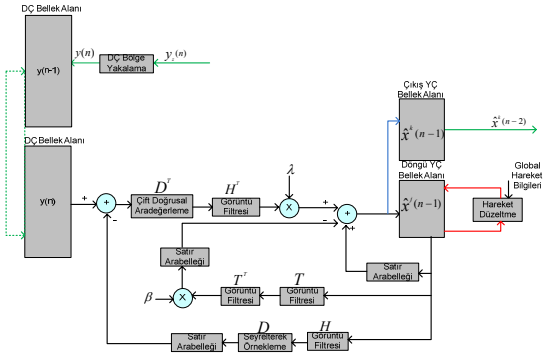


Şekil 4 İlgilenilen Bölgeler

Bu durumda YÇ bölge çözünürlüğü 320x240 olmakta, fakat sistem çıkış çözünürlüğünün 640x480 olması gerektiğinden dolayı YÇ bölge, 640x480'lik bir bölgenin orta kısmına yerleştirilip geri kalan kısmı siyah yapılmaktadır. Şekil 6'teki DÇ bölge yakalama bloğu, giriş videosundaki DÇ bölgeyi belleğe aktarma işlemini yapmaktadır.



Şekil 5 DÇ bellek durumları

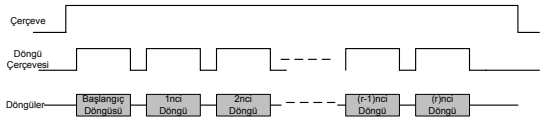


Şekil 6 Yüksek Çözünürlüklü Resim Oluşturma FPGA Yapısı

Yapıda iki adet DÇ bellek alanı kullanılmaktadır. Video akışı esnasında DÇ bölge, bir DÇ bellek alanına yazılırken, bir önceki çerçeve süresinde diğer DÇ bellek alanına yazılmış DÇ bölge işlenmektedir. Bu sayede döngüler için gerekli süre elde edilmektedir. DÇ bellek yapısı Şekil 5'da gösterilmektedir.

Mevcut DÇ bellek alanı, giriş DÇ bölgeyi saklayan bellek alanına verilen isimdir. Önceki DÇ bellek ise bir önceki çerçeve süresindeki DÇ bölgenin saklandığı bellek alanıdır. Bu DÇ bellek alanından her döngüde DÇ bölge okunmakta ve işleme konmaktadır.

Döngüler için döngü video zamanlamaları oluşturulmaktadır. Bir çerçeve süresi içerisine birden fazla döngü çerçevesi sığdırılmalıdır. Şekil 7, bu yapıyı göstermektedir.



Şekil 7 Döngü Yapısı

Şekilde gösterilen başlangıç döngüsü ile Denklem 7'deki işlem yapılmaktadır. Bu döngünün sonucunda YÇ bölge için başlangıç tahmini yapılmaktadır. Diğer döngülerde bu tahmin, Denklem 6'ya göre güncellenmektedir.

Bir çerçeve süresinde döngüler için toplam döngü sayısı kadar yüksek çözünürlüklü video zamanlamaları oluşturulmaktadır. Bir döngünün süresi oluşturulmak istenen YÇ bölgenin çözünürlüğüne bağlıdır. Tablo 1, YÇ bölgenin çözünürlüğüne göre bir çerçeve süresi içerisinde yapılabilecek maksimum döngü sayısını göstermektedir.

Tablo 1 YÇ bölge çözünürlüğüne göre maksimum döngü sayısı

YÇ bölge çözünürlüğü	**160x120**	**320x240**	**640x480**
Toplam yatay aktif veri (Saat çevrimi)	160	320	640
Toplam dikey aktif veri (Satır)	120	240	480
Toplam döngü süresi (ms)	0,387	1,485	5,814
Maksimum döngü sayısı	86	22	5

Şekil 6'te gösterilen görüntü filtreleme blokları, resim filtreleme işlemlerini gerçekleştirir. Katsayıları

değiştirilebilen 5x5'lik bir filtre tasarlanmıştır. Bu sayede tek bir filtre tasarlanarak Denklem 6'daki H, H^T, T ve T^T matrislerinin yaptığı filtreleme işlemleri yapılmaktadır.

Denklem 6'da D^T ile gösterilen ara değerleme işlemi için çift doğrusal ara değerleme metodu [8] kullanılmıştır. Şekil 5'teki çift doğrusal ara değerleme bloğu bu işlemi gerçekleştirmektedir. Bu işlem için DÇ bölgenin bir satırını tutabilecek büyüklükte iki adet bellek alanı kullanılmıştır. Bu sayede video akışı bozulmadan sadece iki satır süresi gecikme ile bloğa giren çerçevenin çözünürlüğü x ve y ekseninde çift doğrusal ara değerleme yöntemi ile iki katına çıkarılmaktadır.

Satır arabelleği blokları FPGA içerisindeki FIFO blokları kullanılarak oluşturulmaktadır. Bu blokların konulma amacı video zamanlamalarını senkronlaştırmaktır.

Seyrelterek örnekleme bloğu x ve y yönündeki çözünürlüğü düşürmek için kullanılır. Maskeleme işlemi ile satırlar ve sütunlar seyreltilerek, çözünürlük x ve y ekseninde yarıya indirilmiştir.

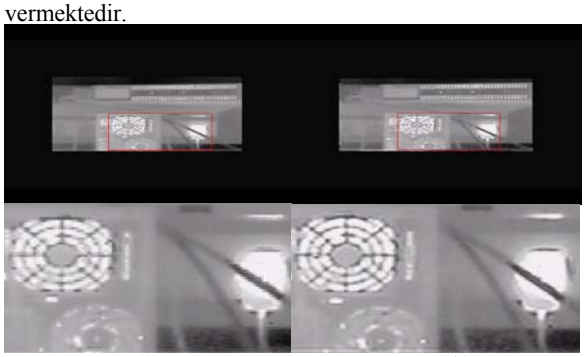
Yapıda iki adet YÇ bellek alanı bulunmaktadır. Bellek alanları, döngü YÇ bellek alanı ve çıkış YÇ bellek alanı olarak adlandırılmıştır. Döngü YÇ bellek alanı, Denklem 6'ya göre güncellenen YÇ bölgeyi saklamaktadır. Çıkış YÇ bellek alanı ise son döngüde elde edilen YÇ bölgeyi saklamaktadır. Bir sonraki çerçeve süresinde çıkış YÇ bellek alanındaki YÇ bölge, çıkış görüntüsü oluşturmak için kullanılmaktadır. Eğer YÇ bölgenin çözünürlüğü sistem çözünürlüğünden küçük ise kalan kısımlar siyah basılarak sistem çözünürlüğü korunmaktadır.

Denklem 7'ye göre, ilk döngüde bir önceki çerçeve süresinde hesaplanmış YÇ bölge ve global hareket bilgileri kullanılarak, yeni çerçeve süresindeki YÇ bölge için başlangıç tahmini oluşturulmaktadır. Bunun için yeni çerçeve süresi başladığında, döngü YÇ bellek alanındaki YÇ bölge okunmaktadır. Global hareket bilgileri kullanılarak resimler arasındaki hareket farklılıkları giderilmektedir. Daha sonra tekrardan döngü YÇ bellek alanına yazılmaktadır. Bu işlem sadece ilk döngüde gerçekleştirilmekte ve kırmızı çizgi ile Şekil 6'te gösterilmiştir. Diğer döngülerde, döngü YÇ bellek alanından, son hesaplanmış YÇ bölge okunmaktadır. Daha sonra Denklem 6'ya göre Şekil 6'te siyah çizgilerle gösterilmiş işlemler yapılarak resim güncellenmektedir. Bu işlemler ilk döngü hariç diğer tüm döngülerde gerçekleştirilmektedir. Son döngüde hesaplanan YÇ bölge sistem çıkışı için kullanılmak üzere döngü YÇ bellek alanına ek olarak çıkış YÇ bellek alanına da yazılmaktadır. Bu işlem Şekil 6'te mavi çizgi ile gösterilmiştir. Yeşil çizgilerle gösterilen işlemler ise her çerçeve süresinde gerçekleştirilen döngülerden bağımsız işlemlerdir.

IV. SONUÇLAR

Global hareket kestirme ve YÇ görüntü oluşturma işlemi Xilinx firmasının Virtex V ailesine ait XC5VFX130T FPGA kullanılarak uygulanmıştır. FPGA uygulaması, sentez sonrasında 7189 Look-up Table, 10636 Flip-Flop ve 2772 KB FPGA içi bellek alanı kullanmıştır. Blok 230MHz hızında çalışabilmektedir.

YÇ bölge çözünürlüğü, 320x240 olarak seçilmiştir. Bir çerçeve süresi içerisindeki toplam döngü sayısı 4 olarak ayarlanmıştır. Şekil 8 (a) kızılötesi kameradan alınan görüntünün çift doğrusal ara değerleme ile x ve y ekseninde 2 kat büyütülmüş halidir. Şekil 8 (b) ise süper çözünürlük bloğu aktif edildiğinde elde edilen görüntüdür. Resimlerden görüldüğü üzere uygulanan metot çift doğrusal ara değerleme yöntemine göre daha iyi görsel sonuçlar vermektedir.



(a) (b)

Şekil 8 Kızılötesi Kamera ile Gerçek zamanlı (FPGA ile) karşılaştırma sonuçları: a) Sadece çift doğrusal ara değerleme yöntemi b) YÇ görüntü oluşturma yöntemi

V. GELECEKTEKİ ÇALIŞMALAR

Global hareket kestirme algoritması, sadece x ve y eksenindeki doğrusal hareketi kestirme yetisine sahiptir. Yani görüntüde rotasyon olduğunda, algoritma yanlış sonuçlar verecektir. Dolayısıyla yüksek çözünürlüklü görüntü oluşturma algoritması da bundan etkilenip hatalı çıktı verecektir. Bunun için görüntüler arası rotasyonu tahmin edebilecek bir yöntem eklenmelidir. Ayrıca yüksek çözünürlüklü görüntü oluşturmada kullanılan hareket düzeltme bloğu da, rotasyon bilgilerini kullanacak şekilde değiştirilmelidir. Gelecekteki çalışmalara çift kübik ara değerleme yönteminin [8] uygulanması da eklenebilir. Yüksek çözünürlüklü görüntü oluşturma bloğunda kullanılan çift doğrusal ara değerleme yöntemine göre işlem yükü fazla olmasına rağmen daha iyi sonuçlar verecektir.

VI. KAYNAKÇA

[1] B. Zitova, J. Flusser, "Image registration methods: a survey", Image and Vision Computing 21 (2003) 977–1000.

[2] S. C. Park, M. K. Park, and M. G. Kang, "Super-Resolution Image Reconstruction A Technical Overview", IEEE Signal Processing Mag., vol. 20, pp. 21-36, May 2003.

[3] K. Sauer, "Efficient block motion estimation using integral projections", Circuits and Systems for Video Technology, vol. 6, pp. 513-518, Oct. 1996.

[4] G. M. Callico, S. Lopez, J. F. Lopez, R. Sarmiento, and A. Nunez. Low-cost implementation of a super-resolution algorithm for real- time video applications. In IEEE International Symposium on Circuits and Systems (ISCAS), 2005.

[5] O. Bowen and C. S. Bouganis. Real-time image super resolution using an FPGA. In International Conference on Field Programmable Logic and Applications (FPL), pages 89–94, September 2008.

[6] M. E. Angelopoulou, C.-S. Bouganis, P. Y. K. Cheung, and G. A. Constantinides. "Robust Real-Time Super-Resolution on FPGA and an Application to Video Enhancement". ACM Transactions on Reconfigurable Technology and Systems, to appear, 2009.

[7] M. Elad and A. Feuer, "Super resolution restoration of an image sequence: adaptive filtering approach," IEEE Trans. Image Processing, vol. 8, pp. 387-395, Mar. 1999.

[8] P. Miklos, "Image Interpolation Techniques", 2nd Serbian-Hungarian Joint Symposium, 2004.

H.264 Çok Bakışlı Video Kodlama İçin Düşük Enerji Kullanımlı Hareket Tahmini Donanımı

Yusuf Akşehir, Kamil Erdayandı, Tevfik Zafer Ozcan ve İlker Hamzaoğlu
Sabancı Üniversitesi
Mühendislik ve Doğa Bilimleri Fakültesi
Orhanlı, Tuzla 34956, İstanbul
{yusufaksehir, kerdayandi, tzaferozcan, hamzaoglu}@sabanciuniv.edu

Özetçe—**Çok Bakışlı Video Kodlama (ÇBVK) stereo (2 bakış) veya çok bakışlı video sinyallerini verimli bir şekilde kodlama işlemidir. H.264 ÇBVK standardı daha iyi sıkıştırma verimliliği sağlamak için hareket tahmininin hesaplama karmaşıklığını çok artırmıştır. Bir bakış içindeki çerçeveler arasındaki zamansal öngörü ve komşu bakışlardaki çerçeveler arasındaki bakışlar arası öngörü H.264 ÇBVK standardının en çok işlem yapılan kısımlarıdır. Bu nedenle bu bildiride H.264 ÇBVK'da zamansal öngörü ve bakışlararası öngörü için yapılan işlem miktarını çok az bir PSNR kaybı ve bit hızı artışıyla çok fazla azaltan özgün teknikler önerdik. Bu bildiride H.264 ÇBVK için önerilen teknikleri de içeren düşük enerji kullanımlı bir hareket tahmini donanımı da önerdik. Önerilen donanımı Verilog HDL ile gerçekledik ve Xilinx Virtex-6 FPGA'sına yerleştirdik. Önerilen özgün teknikler bu FPGA gerçeklemesinin enerji kullanımını %72 azalttı.**

I. GIRIŞ

H.264 video sıkıştırma standardının video sıkıştırma verimliliği daha önceki video sıkıştırma standartlarından çok daha iyi olduğu için, H.264 standardı birçok tüketici elektroniği ürününde kullanılmaya başlandı [1]. Hareket tahmini bir videoyu bu videonun çerçeveleri arasındaki zamansal artıklığı gidererek sıkıştırmak için kullanılır. Hareket tahmini video sıkıştırma işlem yükünün yaklaşık %70'ine sahip olduğundan video kodlayıcı donanımlarının hesaplama karmaşıklığı en yüksek kısmıdır. H.264 standardı daha önceki video sıkıştırma standartlarına göre hareket tahmininin sıkıştırma verimliliğini hareket tahmininin hesaplama karmaşıklığını artırarak artırmıştır.

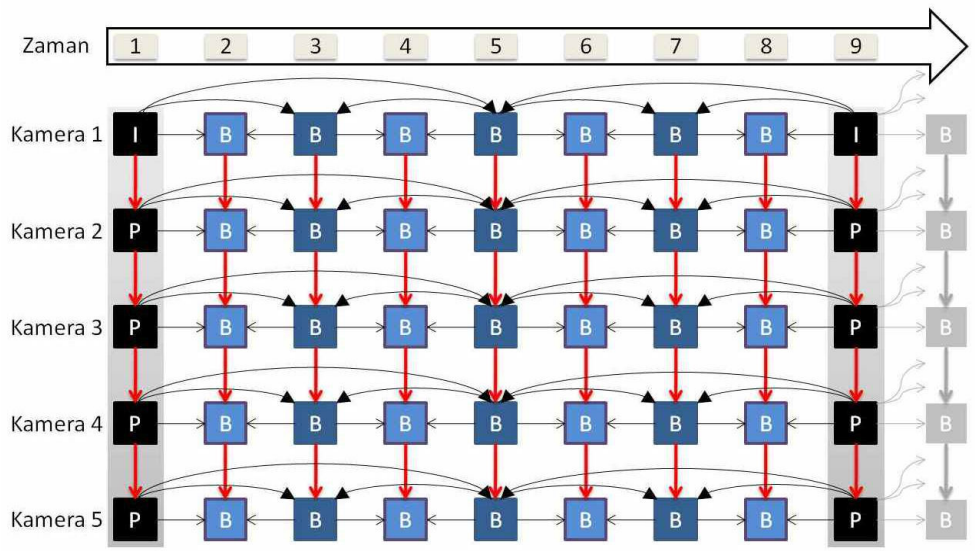
H.264 standardında hareket tahmini için blok eşleştirme algoritmaları kullanılır. Blok eşleştirme işlenmekte olan çerçeveyi NxN boyutundaki makrobloklara (MB) böler. Her işlenmekte olan MB için referans çerçevesindeki arama penceresinde Mutlak Fark Toplamı (MFT) kriterine göre bu MB'la en iyi eşleşen referans MB'u bulur ve bu MB'u gösteren hareket vektörünü belirler.

Çok Bakışlı Video Kodlama (ÇBVK) stereo (2 bakış) veya çok bakışlı video sinyallerini verimli bir şekilde kodlama işlemidir. ÇBVK'nın 3 boyutlu televizyon ve serbest bakış açılı televizyon gibi birçok uygulaması vardır. Şekil 1'de gösterildiği gibi çok bakışlı bir videonun her bakışı birbirinden bağımsız olarak bir H.264 video kodlayıcı ile kodlanabilir [3]. Fakat, çok bakışlı bir videoyu daha verimli bir şekilde sıkıştırmak için, sadece bir bakıştaki çerçeveler arasındaki zamansal artıklıkları değil, aynı zamanda komşu bakışlardaki çerçeveler arasındaki artıklıkları da gidermek gerekmektedir. Bu nedenle H.264 video sıkıştırma standardına ÇBVK özelliği eklendi [2,3,4]. H.264 ÇBVK eşzamanlı bakışları hem bir bakış içindeki çerçeveler arasında zamansal öngörü hem de komşu bakışlardaki çerçeveler arasında bakışlar arası öngörü yaparak kodlar. Böylece bakışların birbirinden bağımsız kodlanmasına göre geri çatılmış videonun kalitesini düşürmeden bit hızında azalma elde eder. Şekil 2'de iki bakışlı video için H.264 ÇBVK işlemi gösterilmiştir [3].

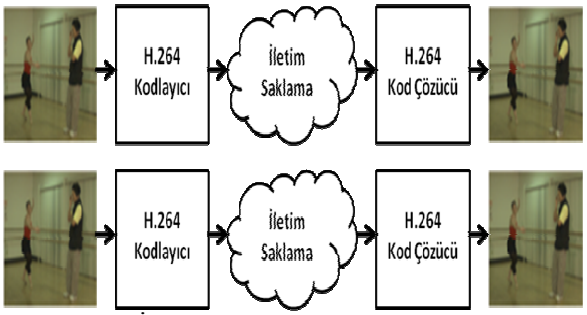
Doğrusal olarak yerleştirilmiş 5 kamera ve 3 seviyeli hiyerarşik B çerçevelerinin kullanıldığı 8 elemanlı resim gruplarından oluşan bir H.264 ÇBVK öngörü yapısı Şekil 3'te gösterilmiştir. Siyahla gösterilen anahtar çerçeveler arasında 8 adet çerçeve olduğu için, bu gruba 8 elemanlı resim grubu adı verilmiştir. H.264 ÇBVK birinci bakıştaki çerçeveleri tek bakışlı video kodlamada olduğu gibi kodlar. I çerçevesindeki bloklar bu çerçevedeki komşu bloklardan çerçeve içi öngörü yapılarak kodlanır. P çerçevesindeki bloklar bu çerçeve ile önceki bir çerçeve arasında hareket tahmini yapılarak çerçeveler arası öngörü yapılarak, kodlanır. B çerçevesindeki bloklar ise bu çerçeve ile hem önceki bir çerçeve hem de sonraki bir çerçeve arasında hareket tahmini yapılarak çerçeveler arası öngörü yapılarak kodlanır. H.264 ÇBVK diğer bakışlardaki çerçeveleri hem bir bakış içindeki çerçeveler arasında hareket tahmini yaparak zamansal öngörü yapıp hem de komşu bakışlardaki çerçeveler arasında hareket tahmini yaparak bakışlar arası öngörü yapıp kodlar.

H.264 ÇBVK standardı daha iyi sıkıştırma verimliliği sağlamak için hareket tahmininin hesaplama karmaşıklığını çok artırmıştır. Bir bakış içindeki çerçeveler arasındaki zamansal öngörü ve komşu bakışlardaki çerçeveler arasındaki bakışlar arası öngörü H.264 ÇBVK standardının en çok işlem yapılan kısımlarıdır. Bu nedenle bu bildiride H.264 ÇBVK'da zamansal öngörü ve bakışlararası öngörü için yapılan işlem miktarını azaltan özgün teknikler önerdik. JMVC 3.01 H.264 ÇBVK yazılımı [6] kullanılarak elde edilen sonuçlar önerilen tekniklerin VGA (640x480) boyutundaki 8 bakışlı ve her bakışında 81 çerçeve olan Ballroom and Vassar çok bakışlı videoları [7] için H.264 ÇBVK'da zamansal öngörü ve bakışlararası öngörü için yapılan işlem miktarını çok az bir PSNR kaybı ve bit hızı artışıyla çok fazla azalttığını gösterdi.

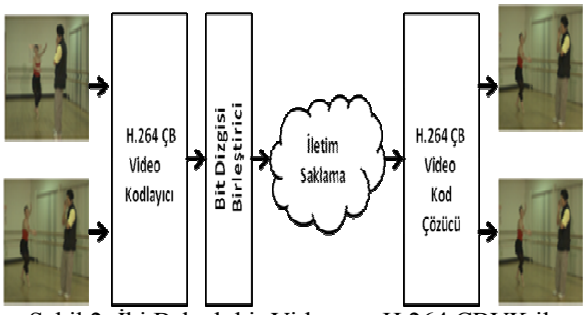
Bu bildiride Şekil 3'de gösterilen H.264 ÇBVK öngörü yapısını gerçekleyen ve önerilen teknikleri de içeren düşük enerji kullanımlı bir hareket tahmini donanımı da önerdik. Önerilen donanımı Verilog HDL ile gerçekledik ve Xilinx Virtex-6 XC6VLX760 FPGA'sına 125 MHz'de yerleştirdik. Bu FPGA gerçeklemesi 13152 slice kullanmaktadır. FPGA



Şekil 3: H.264 ÇBVK Öngörü Yapısı [5]



Şekil 1: İki Bakışlı bir Videonun H.264 Video Kodlayıcı ile Kodlanması [3]



Şekil 2: İki Bakışlı bir Videonun H.264 ÇBVK ile Kodlanması [3]

gerçeklemesini yerleştirme sonrasında zamanlama simulasyonu yaparak doğruladık. FPGA gerçeklemesi CIF (352x288) boyutundaki 8 bakışlı bir videonun saniyede 30*8=240 çerçevesini veya VGA (640x480) boyutundaki 2 bakışlı bir videonun saniyede 30*2=60 çerçevesini işleyebilmektedir. Önerilen özgün teknikler bu FPGA gerçeklemesinin enerji kullanımını %72 azalttı.

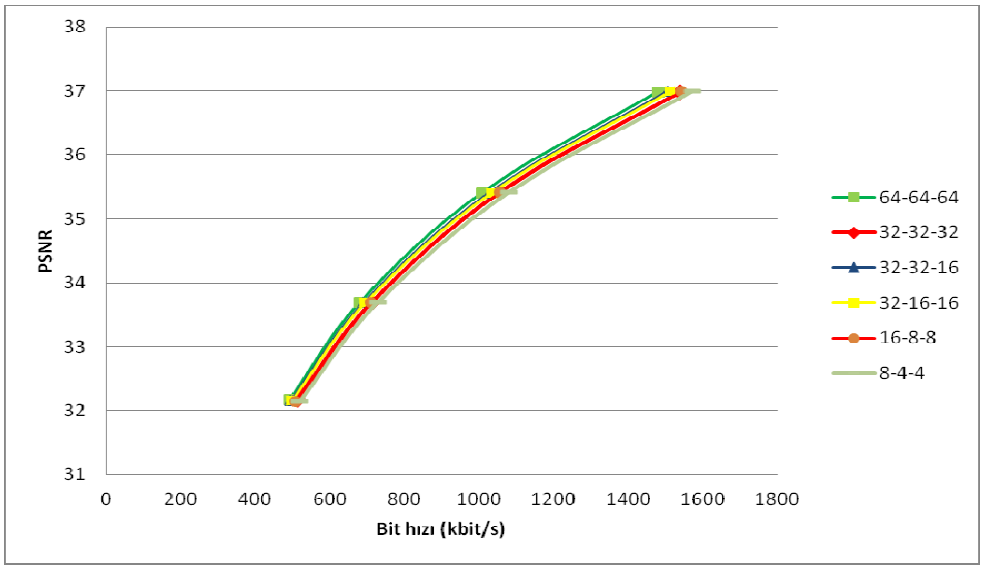
H.264 ÇBVK'da zamansal öngörü ve bakışlar arası öngörü için yapılan işlem miktarı [8]'de olduğu gibi hızlı hareket tahmini algoritmaları kullanılarak daha fazla PSNR kaybı ve bit hızı artışıyla önerilen tekniklerden daha fazla düşürülebilir. H.264 ÇBVK için bu bildiride önerilen hareket tahmini donanımı [8]'de önerilen hareket tahmini donanımından daha fazla işlem yaparak daha kaliteli sonuçlar elde etmektedir.

II. Önerilen İşlem Miktarını Azaltan Teknikler

3 seviyeli hiyerarşik B çerçevelerinin kullanıldığı 8 elemanlı resim gruplarından oluşan bir H.264 ÇBVK yapısını gerçekleyen tam arama hareket tahmini donanımının güç kullanımını azaltmak için hem zamansal öngörü için yapılan bir bakış içindeki çerçeveler arasındaki tam arama hareket tahmini algoritmasının hem de bakışlar arası öngörü için yapılan komşu bakışlardaki çerçeveler arasındaki tam arama hareket tahmini algoritmasının işlem miktarını azaltan özgün teknikler önerdik.

Şekil 3'de gösterildiği gibi 3 seviyeli hiyerarşik B çerçevelerinin kullanıldığı 8 elemanlı resim gruplarından oluşan bir H.264 ÇBVK yapısında 5. çerçeve için 4 önceki ve 4 sonraki (1. ve 9.) çerçeveler referans olarak kullanılarak hareket tahmini yapılır. 3. ve 7. çerçeveler için 2 önceki ve 2 sonraki (1. ve 5., 5. ve 9.) çerçeveler referans olarak kullanılarak hareket tahmini yapılır. Diğer çerçeveler için 1 önceki ve 1 sonraki çerçeveler referans olarak kullanılarak hareket tahmini yapılır. Bu nedenle, zamansal öngörü için yapılan bir bakış içindeki çerçeveler arasındaki tam arama hareket tahmini algoritmasının 5. çerçeve için büyük, 3. ve 7. çerçeveler için biraz küçük, diğer çerçeveler içinse biraz daha küçük hareket vektörü bulacağını dikkate alarak tam arama hareket tahmini algoritmasının arama penceresinin boyutunu 5. çerçeve için büyük, 3. ve 7. çerçeveler için biraz küçük, diğer çerçeveler içinse biraz daha küçük yaparak tam arama hareket tahmini algoritmasının işlem miktarını azaltmayı önerdik.

Şekil 3'de gösterilen H.264 ÇBVK öngörü yapısında olduğu gibi çok bakışlı bir video kayıt edilirken kullanılan kameraların pozisyonları arasında sabit doğrusal bir ilişki varsa, bakışlar arası öngörü için yapılan komşu bakışlardaki çerçeveler arasındaki tam arama hareket tahmini algoritmasının işlem miktarını bu ilişkiyi kullanarak azaltmayı önerdik. 3 seviyeli hiyerarşik B çerçevelerinin



Şekil 4: Hız Bozulum Eğrileri

kullanıldığı 8 elemanlı resim gruplarından oluşan Ballroom ve Vassar çok bakışlı videolarında kameraların pozisyonlarından dolayı aranan makroblok komşu bakıştaki referans çerçevedeki arama penceresinin sağ tarafında bulunmaktadır. Bu nedenle komşu bakıştaki referans çerçevedeki arama penceresinin sadece sağ tarafında arama yapmayı önerdik.

Çok bakışlı bir video kayıt edilirken kullanılan kameraların pozisyonları arasında sabit doğrusal bir ilişki varsa, bakışlar arası öngörü için yapılan komşu bakışlardaki şu anki çerçeveler arasındaki tam arama hareket tahmini algoritmasının bulacağı hareket vektörleri ile aynı komşu bakışlardaki daha önceki çerçeveler arasındaki tam arama hareket tahmini algoritmasının bulduğu hareket vektörleri benzer olacağı için komşu bakışlardaki şu anki çerçeveler arasındaki tam arama hareket tahmini algoritmasının arama penceresinin boyutunu aynı komşu bakışlardaki daha önceki çerçeveler arasındaki tam arama hareket tahmini algoritmasının bulduğu hareket vektörünün büyüklüğüne göre belirlemeyi önerdik.

Şu anki arama penceresinin boyutunu bir önceki hareket vektörü 17'den küçükse 16 ([0,+16]), 33'den küçükse 32 ([0,+32]), diğer durumlarda ise 48 ([0,+48]) yaptık. Ayrıca bir önceki mutlak farklar toplamı (MFT) belli bir eşik değerinden büyükse şu anki arama penceresinin boyutunu 16 artırdık. Bu nedenle arama penceresinin boyutu en fazla 64 ([0,+64]) olabilir. MFT eşik değerini belirlemek için JMVC 3.01 H.264 ÇBVK yazılımının hareket tahmini yaparken hesapladığı MFT değerlerine baktık. Bu MFT değerlerinin çoğu 2000'den küçük olduğu için eşik değerini 1500 olarak belirledik.

Önerdiğimiz tekniklerin işlem miktarını ne kadar azalttıklarını, ne kadar PSNR kaybına ve bit hızı artışına neden olduklarını belirlemek için bu teknikleri JMVC 3.01 H.264 ÇBVK yazılımına ekledik, ve bu yazılımı kullanarak önerilen tekniklerin VGA (640x480) boyutundaki 8 bakışlı ve her bakışında 81 çerçeve olan Ballroom and Vassar çok bakışlı videolarına etkilerini belirledik.

3 seviyeli hiyerarşik B çerçevelerinin kullanıldığı 8 elemanlı resim gruplarından oluşan bir H.264 ÇBVK yapısında 5. çerçeve, 3. ve 7. çerçeveler, ve diğer çerçeveler için tam arama hareket tahmini algoritmasının arama penceresinin boyutunu 64-64-64 ([-64, +64] - [-64, +64] - [-64, +64]), 32-32-32 ([-32, +32] - [-32, +32] - [-32, +32]), 32-32-16 ([-32, +32] - [-32, +32] - [-16, +16]), 32-16-16 ([-32, +32] - [-16, +16] - [-16, +16]), 16-8-8 ([-16, +16] - [-8, +8] - [-8, +8]), and 8-4-4 ([-8, +8] - [-4, +4] - [-4, +4]) yapmanın Ballroom çok bakışlı videosuna etkisini belirledik. Hız bozulum eğrileri Şekil 4'te ve 32 nicemleme parametresi değeri kullanıldığında hesaplanan MFT miktarları Tablo 1'de gösterilmiştir.

Şekil 4'de gösterilen 64-64-64 ve 32-32-32 arama penceresi boyutlarının hız bozulum eğrileri komşu bakışlardaki çerçeveler arasındaki tam arama hareket tahmini algoritmasının işlem miktarını azaltmak için önerilen teknikler kullanılmadan elde edilmiştir. Diğer arama penceresi boyutlarının hız bozulum eğrileri önerilen bütün teknikler kullanılarak elde edilmiştir. Şekil 4'de görüldüğü gibi 64-64-64, 32-32-16 ve 32-16-16 arama penceresi boyutları için birbirine yakın sonuçlar elde edilmiştir. 32-32-32, 16-16-8 ve 8-4-4 arama penceresi boyutları aynı kalitede daha fazla bit kullanılmasına neden olmuştur. Bu nedenle 32-32-16 ve 32-16-16 arama penceresi boyutları 32-32-32 arama penceresi boyutuna göre daha az işlem yaparak daha iyi hız bozulum sonucu, 64-64-64 arama penceresi boyutuna göre ise daha az işlem yaparak benzer hız bozulum sonucu alınmasını sağlamıştır. 32-16-16 arama penceresi boyutu 32-32-16 arama penceresi boyutuna göre yaklaşık %14 daha az MFT hesapladığı için 32-16-16 arama penceresi boyutunu kullanmaya karar verdik.

Şekil 4 ve Tablo 1'deki sonuçlar önerilen tekniklerin H.264 ÇBVK'da zamansal öngörü ve bakışlararası öngörü için yapılan işlem miktarını çok az bir PSNR kaybı ve bit hızı artışıyla çok fazla azalttığını göstermiştir. Önerilen tekniklerin eklendiği JMVC 3.01 H.264 ÇBVK yazılımı ile 22 nicemleme parametresi değeri kullanılarak VGA (640x480) boyutundaki 8 bakışlı ve her bakışında 81 çerçeve olan Ballroom and Vassar çok bakışlı videolarının simulasyonları sonucunda elde edilen PSNR ve bit hızı değerleri sırasıyla Tablo 2 ve 3'de gösterilmiştir.

Tablo 1: Ballroom için Yapılan Hareket Tahmininde Hesaplanan Mutlak Farklar Toplamları (MFT)

Arama Penceresi Boyutları	Bakışlar arası MFT	Zamansal MFT	Toplam MFT	Azalma (%)
32-32-32	2786918400	5505024000	8291942400	0
32-32-16	801155584	3145728000	3946883584	52.40
32-16-16	801860352	1966080000	2767940352	66.61
16-8-8	802980608	491520000	1294500608	84.38
8-4-4	804428032	122880000	927308032	88.81

Tablo 2: Ballroom için PSNR ve Bit Hızı

Bakış	Y	U	V	Bit Hızı
0	40.594	43.080	43.143	4236.669
1	40.268	42.849	42.877	4219.356
2	40.396	43.201	43.224	3810.111
3	40.290	43.020	42.994	3895.188
4	40.241	42.941	42.830	4066.679
5	40.542	43.422	43.333	3625.361
6	40.027	42.681	42.788	4599.654
7	40.165	42.787	42.680	4232.484
Ortalama	40.315	42.998	42.983	4085.688

Tablo 3: Vassar için PSNR ve Bit Hızı

Bakış	Y	U	V	Bit Hızı
0	40.094	42.881	42.577	3663.854
1	40.039	42.866	42.425	3859.089
2	40.264	43.290	42.798	3409.817
3	40.125	43.148	42.950	3377.047
4	40.062	42.984	42.730	3363.538
5	40.600	43.993	43.516	2614.193
6	39.980	42.689	42.221	4484.854
7	40.135	42.908	42.570	3453.607
Ortalama	40.162	43.095	42.723	3528.250

III. ÖNERİLEN H.264 ÇBVK HAREKET TAHMİNİ DONANIMI

Şekil 3'de gösterilen H.264 ÇBVK öngörü yapısını tam arama hareket tahmini algoritması ile gerçekleyen ve önerilen teknikleri de içeren düşük enerji kullanımlı bir hareket tahmini donanımı tasarladık. Şekil 5'de gösterildiği gibi bu donanımda üç tane paralel çalışan tam arama hareket tahmini (HT) modülü var. Şu anki bloğu SoR HT modülü aynı bakış içindeki sol referans çerçevesinde, SaR HT modülü aynı bakış içindeki sağ referans çerçevesinde ve BaR HT modülü komşu bakıştaki referans çerçevesinde paralel olarak tam arama HT algoritması ile arayarak en düşük mutlak farklar toplamını (MFT) veren hareket vektörlerini bulurlar. SoR HT modülünün ve SaR HT modülünün buldukları en düşük MFT'nı veren referans blokların ortalaması alınarak oluşturulan bloğun MFT da hesaplanır. SoR, SaR, BaR HT modüllerinin buldukları 3 hareket vektörünün MFT'leri ve bu MFT karşılaştırılarak şu anki blok için en düşük MFT ve bu MFT'nı veren hareket vektörü bulunur.

HT modüllerini [9]'da önerilen 256 İşlem Birimi (İB) içeren ve sabit boyutta arama penceresi kullanan tam arama hareket tahmini donanımını temel alarak tasarladık. Şekil 6'da gösterildiği gibi bir HT modülünün arama penceresindeki pikseller 20 tane 32*80 bitlik blok RAM'de (BRAM) saklanır. Bir HT modülü once şu anki bloğu 16 saat çevriminde İB dizisine yükler. Daha sonra arama penceresi piksellerini BRAM'lere 5'erli gruplar halinde yükler. HT modülü arama penceresi piksellerini ilk 5 BRAM'in ilk 16 adresine yükledikten sonra MFT hesaplamalarını ve arama penceresi piksellerini BRAM'lere yüklemeyi paralel olarak yapar. Arama penceresi pikselleri ilk 5 BRAM'e 80 saat çevriminde, bütün BRAM'lere 4*80=320 saat çevriminde yüklenir.

MFT'ler 256 İB dizisi ve toplama ağacı tarafından hesaplanır. 256 İB dizisi pikselleri sağa, yukarı ve aşağı kaydırarak verilerin tekrar kullanılmasını sağlar. Verilerin tekrar kullanılması BRAM'lere erişimi önemli ölçüde azaltır. HT modülü MFT'leri hesapladıkca karşılaştırarak en düşük MFT'nı veren hareket vektörünü bulur. HT modülü 16x16 boyutundaki şu anki blok için [-32,+32] boyutundaki arama penceresinde en düşük MFT'nı veren hareket vektörünü 4128 saat çevriminde bulur.

Bu H.264 ÇBVK HT donanımına bu bildiride önerdiğimiz hem bir bakış içindeki çerçeveler arasındaki tam arama hareket tahmini algoritmasının işlem miktarını hem de komşu bakışlardaki çerçeveler arasındaki tam arama hareket tahmini algoritmasının işlem miktarını azaltan özgün teknikleri uyguladık. Bu donanımı Verilog HDL ile gerçekledik. Verilog RTL kodlarını Xilinx ISE 11.4 yazılımını kullanarak Xilinx Virtex-6 XC6VLX760 FPGA'sına 125 MHz'de sentezleyip yerleştirdik. Verilog RTL kodlarının FPGA'ya yerleştirilmesi sonrasında oluşan devreyi Mentor Graphics Modelsim 6.1c yazılımı ile zamanlama simulasyonu yaparak doğruladık. Önerilen donanım bu FPGA'da 13152 slice, 40874 LUT, 22024 DFF ve 60 BRAM kullanmaktadır.

Donanımın FPGA gerçeklemesi 3 seviyeli hiyerarşik B çerçevelerinin kullanıldığı 8 elemanlı resim gruplarından oluşan bir çok bakışlı videonun bir resim grubunun 5. zamanındaki çerçevenin bakışlar arası ve çerçeveler arası öngörüsünü 41.6ms de yapar. Birinci bakıştaki diğer B çerçevelerinin bakışlar arası ve çerçeveler arası öngörüsünü 10.4ms de yapar. Diğer bakışlardaki diğer çerçevelerin bakışlar arası ve çerçeveler arası öngörüsünü 14.44ms de yapar. Bu nedenle donanımın FPGA gerçeklemesi 3 seviyeli

hiyerarşik B çerçevelerinin kullanıldığı 8 elemanlı resim gruplarından oluşan VGA (640x480) boyutundaki iki bakışlı bir videonun bir resim grubunun bakışlar arası ve çerçeveler arası öngörüsünü 41.6*2 + 10.4*6 + 14.44*7 = 246.68 ms de yapar. Bu nedenle VGA (640x480) boyutundaki 2 bakışlı bir videonun saniyede 30*2=60 çerçevesini işleyebilir. Aynı şekilde, CIF (352x288) boyutundaki 8 bakışlı bir videonun saniyede 30*8=240 çerçevesini işleyebilir.

Hem önerilen işlem miktarını azaltan teknikleri içermeyen ve bakışlar arası ve çerçeveler arası öngörü için [-32,+32] boyutunda arama penceresi kullanan H.264 ÇBVK HT donanımının hemde önerilen işlem miktarını azaltan teknikleri içeren H.264 ÇBVK HT donanımının aynı FPGA'da VGA (640x480) boyutundaki Ballroom çok bakışlı videosunun 3. bakışındaki 2. çerçevesi için ne kadar güç kullandıklarını Xilinx Xpower yazılımını kullanarak belirledik. Önerilen teknikleri kullanmayan donanımın FPGA gerçeklemesi 1489.6 mW güç and 15.4 mJ enerji kullanmaktadır. Önerilen teknikleri kullanan donanımın FPGA gerçeklemesi 1529.8 mW güç and 4.3 mJ enerji kullanmaktadır. Bu nedenle önerilen teknikler H.264 ÇBVK HT donanımının FPGA gerçeklemesinin enerji kullanımını %72 azalttı.

IV. TEŞEKKÜR

TÜBİTAK'a 111E013 numaralı proje kapsamında bu araştırmayı desteklediği için teşekkür ederiz.

V. KAYNAKÇA

[1] ITU-T and ISO/IEC JTC 1, Advanced video coding for generic audiovisual services, ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), 2010.

[2] ISO/IEC JTC1/SC29/WG11, "Text of ISO/IEC 14496-10:200X/FDAM 1 Multiview Video Coding", Doc. N9978, Hannover, Almanya, Temmuz 2008.

[3] P. Merkle, K. Muller, T. Wiegand, "3D Video: Acquisition, Coding, and Display", *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, 2010.

[4] A. Vetro, T. Wiegand, G. J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard", *Proceedings of the IEEE*, 2011.

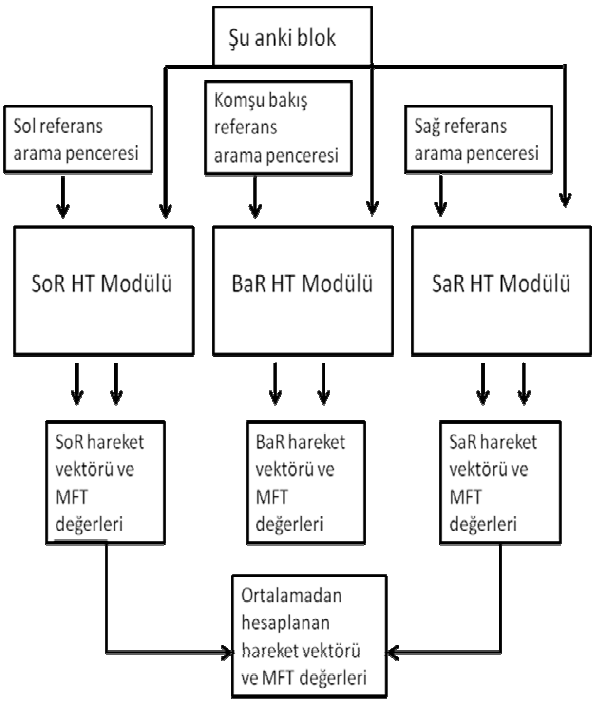
[5] P. Merkle, A. Smolic, K. Muller, T. Wiegand, "Efficient Prediction Structures for Multiview Video Coding", *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 17, no. 11, Kasım 2007.

[6] http://wftp3.itu.int/av-arch/jvt-site/2009_01_Geneva

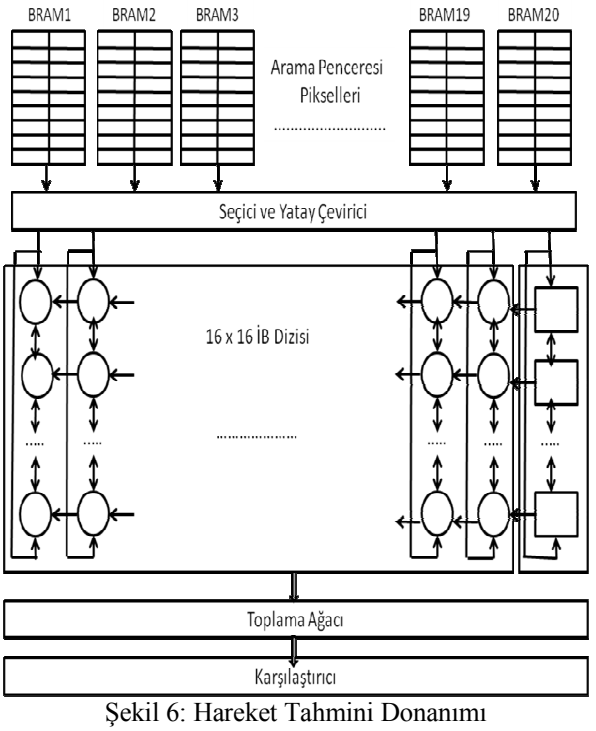
[7] http://www.merl.com/pub/avetro/mvc-testseq/orig-yuv

[8] B Zatt, S Bampi, M Shafique, J Henkel, "Multi-Level Pipelined Parallel Hardware Architecture for High Throughput Motion and Disparity Estimation in Multiview Video Coding", *DATE Conference*, Mart 2011.

[9] C. Kalaycioglu, O. C. Ulusel, I. Hamzaoglu, "Low Power Techniques for Motion Estimation Hardware", *International Conference on Field Programmable Logic and Applications*, Eylül 2009.



Şekil 5: H.264 ÇBVK Hareket Tahmini Donanımı



Şekil 6: Hareket Tahmini Donanımı

FPGA ÜZERİNDE SANAL MİKRODENETLEYİCİ KULLANILARAK SESLİ KOMUT UYGULAMASI

Süleyman Urmat, Evren Cesur, Nerhun Yıldız ve Vedat Tavşanoğlu
Yıldız Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Esenler, 34220, İstanbul
e-posta : l1406073@std.yildiz.edu.tr, ecesur@yildiz.edu.tr

Özetçe— **Bu çalışmada Sahada Programlanabilir Kapı Dizileri (Field Programmable Gate Array,FPGA) üzerinde yazılımsal bir mikrodenetleyici kullanılarak , tek bir platform üzerinde (system on a chip, SoC), sesli harflerin tanınması hedeflenmiştir. Sesli harflerin tanınması işlemi için Doğrusal Öngörü Algoritması (Linear Prediction Coding, LPC) algoritması kullanılmıştır. Tasarlanan sistemde sanal mikrodenetleyici openMSP430 üzerine DMA(Direct Memory Access) çevresel birimi eklenmiştir.**

I. GIRIŞ

Son yıllarda kullanılan gömülü sistemlerin büyük bir bölümü, tek bir platform üzerinde [1] sunulmaktadır. Buna uygun olarak bu çalışmada ses tanıma sistemi, Sahada Programlanabilir Kapı Dizileri (Field Programmable Gate Array,FPGA) [2] ile gerçeklenmiştir. FPGA üzerinde ses tanıma işlemi açık kaynak kodlu sanal bir mikrodenetleyici ile yapılmıştır. İlaveten sanal mikrodenetleyiciye yazılımsal olarak tasarlanan DMA(Direct Memory Access) çevresel birimi eklenmiştir.

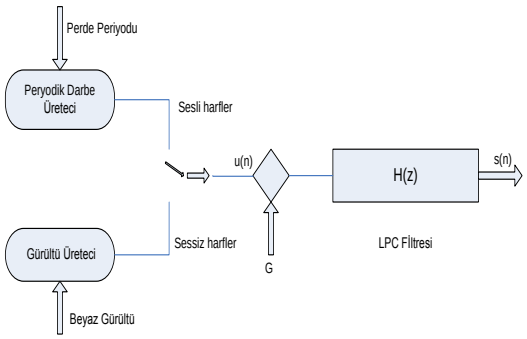
Ses tanıma işlemi, fiziksel olarak yapılması gereken bazı işlerin sesli komutlar yardımıyla kolaylıkla yapılabilmesine olanak sağlamaktadır.Ayrıca ses tanıma sistemi, FPGA platformu üzerinde yapılarak fiziksel olarak daha küçük boyutlarda bir cihazın gerçeklenmesine olanak sağlamaktadır.

Çalışma da esas itibariyle sanal bir mikrodenetleyici kullanılarak sesli harflerin tanınması amaçlanmıştır. Buna uygun olarak, ilk aşamada sesli harflerin tanınması için gerekli simulasyonlar Matlab üzerinde yapılmıştır. İkinci aşamada ise ses tanıma sisteminin FPGA üzerinde gerçeklenmiştir. FPGA üzerindeki modüller AC'97 Kontrolör ve openMsp430'dan [3] oluşmaktadır. Bu modüllerden AC'97 Kontrolör, ses işaretlerinin kaydedilmesi için AC'97 [4] standardı uyumlu analog sayısal dönüştürücüyü kontrol etmektedir. Ayrıca AC'97 Kontrolör modülü içersinde; sanal mikrodenetleyici üzerindeki işlem yükünü azaltmak için AC'97 Kontrolör ile openMSP430 arasındaki veri trafiğini kontrol eden DMA (Direct Memory Access) birimi yazılmıştır. AC'97 Kontrolör ve DMA modülleri VHDL ile yazılımıştır. openMSP430 sanal mikrodenetleyici 16 bitlik olup,Verilog ile yazılmış açık kaynak kodlu yazılımsal bir mikrodenetleyicidir. Sanal mikrodenetleyici üzerinde koşan kod C dili ile yazılmıştır.

Çalışma içerik olarak; ilk kısmında sesli harflerin tanınmasının matematiksel detayları yer almaktadır. Sonraki ksımda sesli harflerin tanınmasının Matlab üzerinde simulasyonu ve elde edilen sonuçlar sunulmuştur. En son olarak donanımsal tasarım ayrıntılı olarak anlatılmıştır.

II. SESLİ HARFLERİN İŞLENMESİNİN MATEMATİKSEL DETAYLARI

Ses işareti genel olarak sesli ve sessiz dalgalanmalar olarak tanımlanabilir. Sesli işaretler temel bir frekansa sahiptir. Ses tellerinden gelen uyarımlar gırtlakta darbeler olarak şekillenir ve temel frekansın yanında yan frekanslar (formantlar) oluşur. Sesli harflerin tanınmasında temel frekans ve formantlar belirleyici rol oynar. Sessiz harfler ise temel bir frekansa sahip değildir, daha çok beyaz gürültüye benzerler [5].



Şekil 1: Ses üretim modeli.

Ses işaretinin işlenebilmesi için, ilk olarak sayısal hale getirilmesi gerekir. Bu amaçla ses işareti örneklenir. Örneklenen işaret gürültülerden arındırmalı ve ses işaretinden alınan örneklerin başında ve sonunda oluşan süreksizlikleri gidermek için pencereleme işlemi yapılır. Ses işaretinin tanıma kısmında ise, elde edilen ses işaretinin öznitelikleri belirlenir. Bulunan öznitelik vektörlerinden yararlanılarak tanıma işlemi yapılır.

Pencereleme İşlemi

Örneklenmiş işaretin başında ve sonunda oluşan süreksizlikleri önlemek için yapılır. Pencereleme işlemi için aşağıdaki eşitlik (Hamming-Windowing) kullanılmıştır.

$$W_n = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), & 0 \le n < N \\ 0, & yada \end{cases} \quad (1)$$

Doğrusal Öngörü Algoritması (LPC)

Ses işaretlerinin tanınmasında kullanılan en yaygın algoritmalardan biridir. Doğrusal Öngörü algoritması düşük bit hızlarında sesin temsil edilmesini sağlarken, konuşma işaretinin saklanmasında ve işlenmesinde kullanılan perde, formant ve izge gibi parametrelerin kestiriminde üstün bir yöntemdir [6]. Konuşmayla ilgili birçok parametreyi daha az sayıda parametre ile ifade etmemizi sağladığından, işlem hızını ve bellek ihtiyacını en aza indirir. Bu algoritmadaki temel işleyiş, bir ses işaretinin daha önceki ses örneklerinin bir doğrusal birleşiminden elde edilmesidir. LPC katsayıları ise ses işaretiyle, onun doğrusal öngörüleri arasındaki karesel farkı en aza indirgeyen katsayılardır. Bu katsayılardan elde edilen LPC spektrumu spektral tepeler üzerinde yoğunlaşan bir tüm kutup fonksiyonu ile modellenir [7].

x̃ ses işaretinin , a_i LPC katsayıları ile M adet geçmiş örneğin doğrusal öngörüsü ile elde edilişi aşağıdaki gibidir .

$$\tilde{x}(n) = a_1 x(n-1) + \cdots + a_M x(n-M) = \sum_{i=1}^{M} a_i x(n-i) \quad (2)$$

Ses işaretiyle, onun doğrusal öngörüleri arasındaki farkın karesini en aza indirgeyecek katasayılar LPC katsayılarını oluşturur.

$$\varepsilon(n) = x(n) - \tilde{x}(n) = x(n) - \sum_{i=1}^{M} a_i x(n-i) \quad (3)$$

Doğrusal Öngörü hatasının karesi aşağıdaki gibi ifade edilir [8].

$$E = \sum_n \varepsilon^2 = \sum_n \left\{ x(n) - \sum_{i=1}^{M} a_i x(n-i) \right\}^2 \quad (4)$$

(4) numaralı eşitlikteki hatayı en aza indirgeyecek katsayıları bulmak için E'nin a'ya bağlı türevini (zincir kuralıyla alarak) sıfıra eşitleriz.

$$2 \sum_n x(n-k) \left[x(n) - \sum_{i=1}^{M} a_i x(n-i) \right] = 0$$
$$k = 1, 2, \cdots, M \quad (5)$$

eşitliğini seriye açacak olursak,

$$a_1 \sum_n x(n-k)\, x(n-1) + a_2 \sum_n x(n-k)\, x(n-2) + \cdots + a_m \sum_n x(n-k)\, x(n-M) = \sum_n x(n-k)\, x(n) \qquad k = 1,2,3,\cdots,M. \quad (6)$$

Konuşma işaretinin N uzunluklu örneklerden oluştuğunu varsayalım. Bu örnekleri 0'dan *N*-1'ekadar dizinlersek (*x(n)={x(0), x(1), x(2), ⋯ , x(N-2),x(N-1)}*) bu örneği *MxM* bir matris şeklinde ifade edebiliriz.

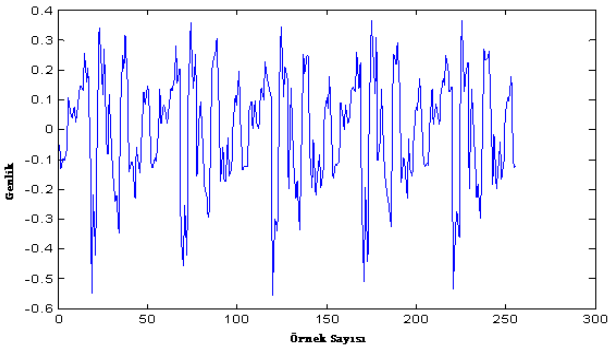
$$\begin{bmatrix} r(0) & \cdots & r(M-2) & r(M-1) \\ r(1) & \cdots & r(M-3) & r(M-2) \\ \vdots & \ddots & \vdots & \vdots \\ r(M-2) & \cdots & r(0) & r(1) \\ r(M-1) & \cdots & r(1) & r(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{M-1} \\ a_M \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(M-1) \\ r(M) \end{bmatrix} \quad (7)$$

$$r(k) = \sum_{n=0}^{N-1-k} x(n)x(n+k). \quad (8)$$

(7) numaralı eşitlikteki matrisin çözümü için Levin-Durbin modeli kullanılmıştır.

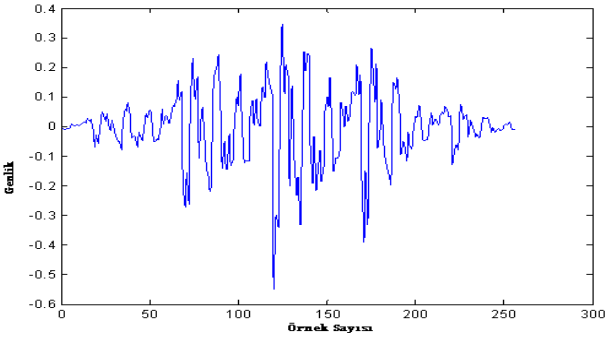
III. SESLI HARFLERIN TANINMASININ MATLAB ÜZERINDE SIMULASYONU

Sesli harflerin tanınmsı işlemini donanıma aktarmadan önce gerekli simülasyonlar Matlab üzerinde yapılmıştır. Ses işareti ilk önce ***wav*** formatında 8 kHz'de örneklenerek kaydedilmiştir. Kaydedilen ses işareti 256 örnek uzunluklu parçalara bölünmüştür.



Şekil 3: 256 örnek uzunluklu A harfi

Daha sonra bu parçalar içinden alınan 256 örnek uzunluklu ses örneği Hamming-Penceresinden geçilir.



Şekil 4: Pencerelenmiş A harfi

Bundan sonraki adımda pencerelenmiş sesli harfin LPC katsayıları bulunur. Bulunan LPC katsayıları yardımıyla sesli harfe ait formantlar hesaplanır [9].

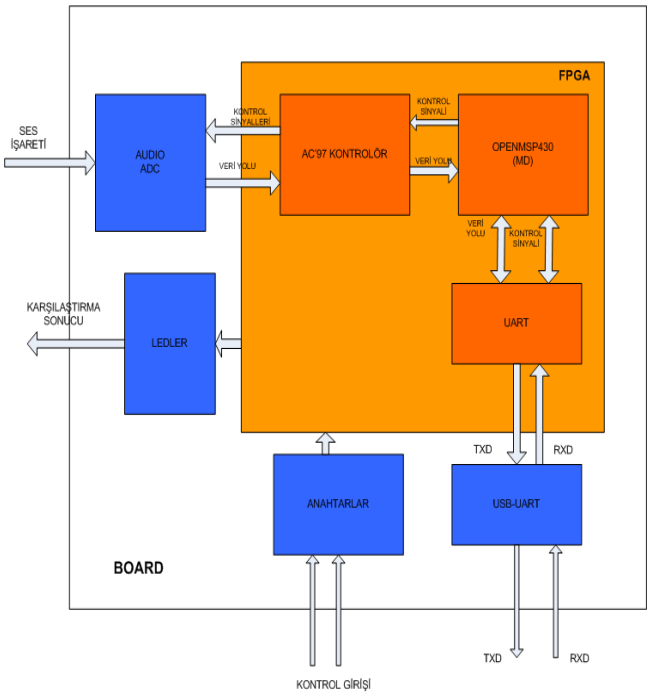
Sesli harfe ait formant değerleri bulunduktan sonra, sesli harflerin tanıma işlemi; eğitilen sesli harflerin formant değerleriyle, tanınması istenen sesli harfin formant değerleri belli bir yüzde sapmasıyla karşılaştırılarak yapılır. Bu sapma değeri deneysel olarak bulunarak uygulanmıştır. Sesli harflerin tanınması Matlab üzerinde %60 başarı ile sağlanmıştır.

Tablo 1: /a/e/ı/i/ ilişkin formant değerleri

Harfler		**a**	**e**	**ı**	**i**
Formantlar Hz	f_1	600	317	315	277
	f_2	1140	480	1510	2312
	f_3	2483	1920	2671	2800

IV. DONANIMSAL TASARIM

Donanım üstünde yapılacak sesli harflerin tanınması işleminde sırasıyla, ilk önce eğitilecek olan sesli harfler ve karşılaştırılacak sesli harf mikrofon yardımıyla kaydedilmiştir. Daha sonra bu ses işaretleri ses tümdevresi içerisinde 8 kHz'de 18-bit çözünürlüğünde örneklenerek saklanmıştır. Saklanan bu ses işaretleri 16 bitlik (18-bitlik verinin en anlamsız iki biti ihmal edilir) 240 örnek uzunluklu ses işaretleri openMSP430 içerisine DMA yardımıyla yazılmıştır. DMA işlemi, AC'97 Kontrolör modülü içerisindeki seslerin kayıtlı oluğu RAMler'den openMSP430'un veri belleğine doğrudan yazılmasına olanak sağlar. DMA yardımıyla işlemciye yazılan ses işaretleri sırasıyla pencereleme işleminden geçirilir. Daha sonra LPC katsayıları bulunur.



Şekil 4: Tasarlanan sistemin blok diyagramı

openMSP430

openMSP430 açık kaynak kodlu mikrodenetleyici çekirdeği, Texas Instruments MSP430 ailesiyle uyumlu olup, onun komut kümesini desteklemektedir. Bu temel çekirdeğin özellikleri [10], 16x16 donanımsal çarpıcı, genel amaçlı giriş-çıkış portları, zamanlayıcı/sayıcı olarak TimerA ve programlama için de seri port hata ayıklama ara yüzü şeklindedir.

Yazılım ortamı MSPGCC [11] adlı C derleyicisi kullanılarak sağlanmıştır.

openMSP430 özellikleri ve kısıtlamaları (MSP430™ ile karşılaştırarak) şunlardır.

openMSP430 Temel İşlemci Özellikleri

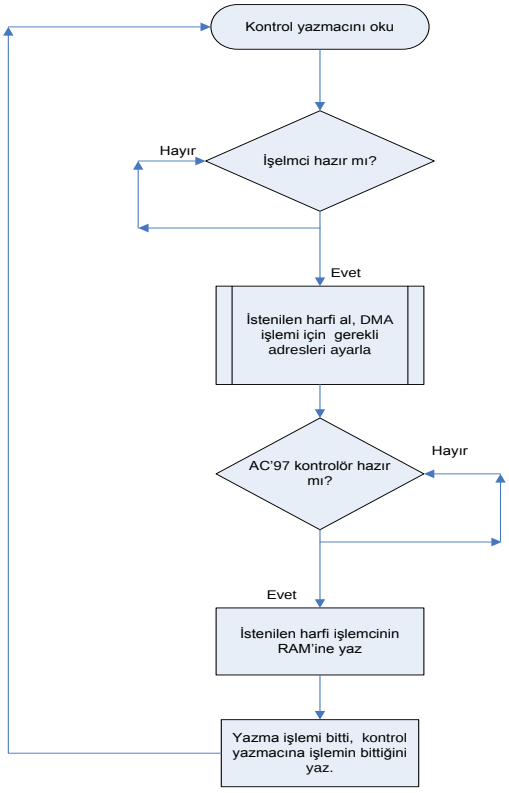
- Tüm komut kümesini ve adresleme türlerini destekler.
- IRQ (örtülebilir kesme) ve NMI (donanımsal kesme) kesmelerini destekler.
- Güç tasarrufu modlarını destekler.
- Artırılabilir program ve veri belleğine sahiptir.
- Seri hata ayıklama ara yüzüne sahiptir.

OpenMSP430'un MSP430'dan Eksikleri

- Temel sistem saati birimi, MSP430™'in saat birimini tam olarak desteklemez.
- Program belleği, veri belleğinden yürütülemez.
- MCLK (Master Clock) ölçeklenemez ve saat kaynağı olarak DCO (Digitally Controlled Oscillator) kullanılır.

DMA Bloğu

DMA işlemi için, openMPS430'un bellek haritası üzerinde 256 kelime uzunluğunda (0-255 arası) bir bellek alanı ayrılmıştır. Bu bellek alanının 240 kelime uzunluğundaki kısmına, AC'97 Kontrolör bloğunda yer alan örneklenmiş ses işaretleri yazılacaktır. Son kelime (255. kelime) ise kontrol yazmacı olarak kullanılacaktır. Kontrol yazmacı DMA işlemi için hem bayrak görevi görür hem de işlemcinin RAM'ye yazılacak harfleri belirler.



Şekil 5: DMA kontrol bloğu akış diyagramı

V. SONUÇLAR

Bu çalışmada hedeflenen sanal bir mikrodenetleyici kullanılarak FPGA platformu üzerinde sesli harflerin tanınmasıydı. İlk aşamada LPC algoritması ve seslerin tanıma işlemi Matlab üzerinde denenmiştir. Daha sonraki

kısımda donanım üzerinde sesli harflerin tanınma işlemine geçilmiştir. Donanım kısmında ise sesli harfleri kaydetmek için, AC'97 ses tümdevresini kontrol etmek için AC'97 Kontrolör modülü çalıştırılmıştır. Kaydedilen ses işaretlerini openMSP430 içine almak için DMA bloğu kullanılmıştır. DMA işlemini yapabilmek için işlemcinin RAM'leri dual port RAM'ye çevrilmiştir. DMA sayesinde işlemci ses verileri alırken, mikrodenetleyiciye fazla bir işlem yükü düşmeyecek ve bellek alanıda daha verimli kullanılmış olur. Sonra DMA yardımıyla içeriye alınan seslerin LPC katsayıları bulunmuştur.

Donanım üzerinde sesli harflerin tanınması kısmı tam olarak bitirilememiştir. Ses tanıma kısmı için donanımsal gerekli altyapı sağlanmıştır, ama sesli harflerin tanınması kısmı yazılımsal olarak geliştirilmesi gerekmektedir.

Sistem üzerinde kullanılan sanal mikrodenetleyici 40 MHz saat hızı, 41 KB program belleği ve 10 KB veri belleği ile çalıştırılmıştır FPGA kullanımına ilişkin bilgiler aşağıdaki tabloda verilmiştir.

Tablo 2: Tasarlanan sistemin lojik kullanımı

Lojik Üniteler	**Kullanılan**	**Mevcut**	**Yüzde**
Number of Slice Registers	1391	54576	2%
Number of Slice LUTs	8752	27288	32%
Number of bonded IOBs	28	218	12%
Number of Block RAM/FIFO	23	116	19%
Number of DSP48A1s	1	58	1%

VI. TEŞEKKÜR

Elektrik Mühendisleri Odasına bu çalışma kapsamında sağlamış oldukları FPGA kiti için teşekkür ederiz.

VII. KAYNAKÇA

[1] "System on a chip, " http://en.wikipedia.org/wiki/System_on_a_chip, son erişim Kasım 2012.

[2] " Field-programmable gate array, " http://en.wikipedia.org/wiki/Field-programmable gate array, son erişim Kasım 2012

[3] O. Girard, "openMSP430, " Rev.1.7, Ağostos 2012

[4] Intel, "Audio Codec '97," Revision 2.3, Nisan 2002

[5] Ergenç İ. , Güner L., "Sesin Doğası ve Oluşumu, " http://www.jandarma.tsk.tr/kriminal/turkish_internet/anasayfa/bilarinde_dıosyalar/bilarinde5.htm, pp. 11-21, son erişim Ekim 2012.

[6] Özer, H. , Arslan, L. , "LPC İzge Düzgünlüğü İle Konuşma Aktivitesi Tespiti," TÜBİTAK-UEKAE ve Elektrik ve Boğaziçi Üniversitesi Elektronik Mühendisliği Bölümü.

[7] Bolat, B. "Kamışlı Enstrüman Seslerinin Istatistiksel Sinir Ağlari İle Taninmasi, "Yıldız Teknik Üniversitesi, Elektrik-Elektronik Fak., Elektronik ve Haberleşme Müh. Böl., Yıldız-İSTANBUL, 2005

[8] Park, S. , " Chapter 7 Linear Predictive Speech Processing, " ders notları, pp. 1-11.

[9] "Formant Estimation Programs, " http://www.clear.rice.edu/elec431/projects96/digitalbb/fmnts.html, son erişim Ekim 2012

[10] Texas Instruments, "MSP430x1xxFamily User Guide, "2006

[11] "The GCC toolchain for Texas Instruments MSP430 MCU's, " http://mspgcc.sourceforge.net , son erişim Ekim 2012

SpO2 Ölçümü İçin Basit Bir Pulse Oksimetre Tasarımı

Tevfik Kadıoğlu
Yıldız Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Esenler, 34220, İstanbul
e-posta: tevfik.kadioglu@gmail.com

Serkan Erboral
İstanbul Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Maslak, 34469, İstanbul
e-posta: serkanerboral@gmail.com

Hakan Üner
Özel Küçükyalı Delta Hospital
Biyomedikal Birimi
Küçükyalı, 34840, İstanbul
eposta: hakanuner34@gmail.com

Özetçe—**Kandaki oksijen miktarı ve kalp atım hızının sürekli ölçümü yaşlılar, hamileler ve yoğun bakımda yatan hastalar için önemlidir. Yaygın olarak, ölçümler pulse oksimetre yöntemi kullanan cihazlar ile yapılmaktadır. Bu bildiride düşük maliyetli, küçük boyutlu ve düşük güç tüketimli, hastanın oksijen doygunluğunu ve kalp atım hızını ölçen bir cihaz tasarımı önerilmektedir. Öncelikle oksijen doygunluğu ve ölçüm kuramı üzerine değinilmiş ardından tasarlanan donanım ve üzerinde koşan yazılım tanıtılmıştır.**

I. GİRİŞ

Kalp atım hızı, kan basıncı, solunum hızı ve vücut sıcaklığı gibi ölçümlere ek olarak pulse oksimetre bilgisi de hastanın durumu konusunda bilgi veren bir ölçümdür. Birçok doku ve organ yeterli oksijeni alamadığı takdirde geri dönüşümsüz olarak zarar görmektedir. Oksijen yetersizliğinde bir kaç dakika içinde kalp, karaciğer ve böbrekler kalıcı olarak zarar görür. Bu süre beynin korteks tabakası için 1 dakikanın altındadır. [1] Bu ve benzeri örneklere dayanarak kandaki oksijen miktarını gerçek zamanlı olarak ölçmenin çok önemli olduğu söylenebilir. Pulse oksimetreler 1970'li yılların başlarından itibaren anestezi ve yoğun bakım ünitelerinde yoğun olarak kullanılmaya başlanmış ve ölçüm yöntemi zamanla güvenilirliğini ispatlamıştır.

II. PULSE OKSİMETRİ

Pulse Oksimetri yöntemi, arteriyal kandaki O_2 doygunluğunun dolaylı olarak (non-invasive), oksimetri ve pletismografi prensiplerinin birleşimi ile ölçümüdür. [2] 1930'lu yıllardan bu yana bilinen bu yöntem, bir ışık kaynağı ve ışık dedektöründen oluşan sensörün arasına parmak ucu, kulak memesi gibi iyi optik geçirgenliği olan dokuların yerleştirilmesi ile yapılmaktadır. Kapiller dolaşımda arteriyoler pulsasyon olması dolayısıyla bu yönteme "pulse oksimetri" adı verilmiştir.

Oksimetrede temel kural, oksijen yüklü ve yüklü olmayan hemoglobinin ayırt edilmesidir. [3] Bu ayrımı yapabilmek amacıyla ölçülebilecek bir değişime ihtiyaç vardır. Bu yöntemde, pompalanan kandaki oksijen doygunluğunun ışığın soğurma oranında oluşturduğu değişim ölçülmektedir. Pompalanan kana uygulanan kırmızı ve kırmızıya yakın kızılötesi ışınların ne kadar soğurulduğu tespit edilerek ölçüm alınmaktadır.

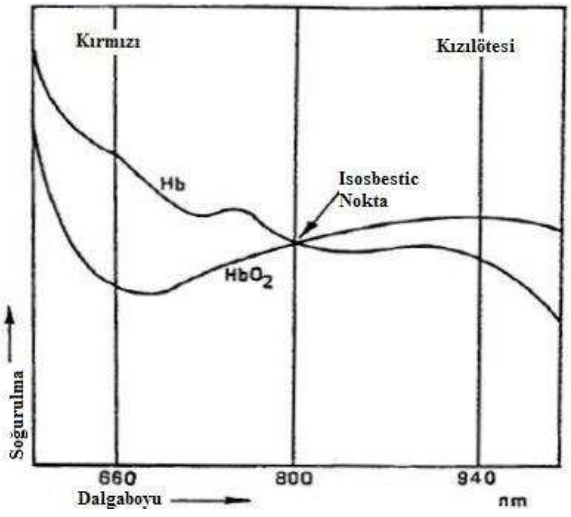
A. Klinik Kullanımı

Pulse oksimetri yöntemi günümüzde anestezi sırasında, yoğun bakımda, anestezi sonrası bakım ünitelerinde, endoskopik girişimlerde yoğun olarak kullanılmaktadır.

Kullanılan ölçüm modeline göre; SpO_2 eğer %90'dan fazla ise standart sapma (SD) ±%2 civarındadır. Eğer SpO_2 %90'dan az ise standart sapma ±%6 civarında olabilmektedir. Yapılan çalışmalarda gösterilmiştir ki; düşük satürasyon durumunda hatalı okuma oranı artmaktadır. [4, 5, 6, 7]

B. Çalışma Prensibi

Oksijen doygunluğu; bir adet foto diyot, bir adet kırmızı (660nm) ve bir adet kızıl ötesi (940nm) LED (Light emitting diode) kullanarak dokudan geçen ışığın şiddetine göre ölçülebilir. Oksijene doygun hemoglobinde kızıl ötesi ve kırmızı ışığın soğurulma miktarları farklıdır. Oksijene doygun bir hemoglobin için dalga boyuna göre emilimi gösteren grafik **Hata! Başvuru kaynağı bulunamadı.**'de verilmiştir. [8]



Şekil 1: Oksijenlenmiş ve oksijenlenmemiş hemoglobinin dalga boyuna göre ışık soğurumu

Farklı dalga boyunda LED ışık kaynakları (660nm ve 940nm) zaman dönüşümlü olarak ışık geçirgenliği olan bölgeye (deri vb.) uygulanarak ölçüm alınır. Daha önce yapılan çalışmalar, oksijen yüklü hemoglobin (HbO2) ve oksijen yüklü olmayan hemoglobin(Hb) moleküllerinin aynı dalga boyunda farklı optik soğurma özelliği olduğunu ortaya koymuştur. En iyi ölçüm için seçilen iki dalga boyu **Hata! Başvuru kaynağı bulunamadı.**'den de görüleceği gibi 660nm ve 940nm'dir. [9] Bu iki dalga boyunda HbO_2 soğurulması ile Hb soğurulması arasındaki fark en fazladır.

$SpO_2 = \frac{HbO_2}{HbO_2 + Hb} x100$ olarak yüzde cinsinden hesaplanır.

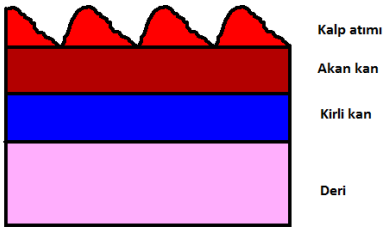
[10] Beer-Lambert modeli doğrultusunda; soğuran cisim iletilen ışık şiddetine bağlı bir çözümle belirlenebilir. İletilen ışık şiddeti (I_o) gelen ışığın şiddetine bağlıdır (I_N).

$$I_o = I_N e^{-\varepsilon cL}$$

Burada; ε dalga boyuna bağlı sönümleme katsayısı, c soğuran yoğunluğu ve L [cm] optik yol uzunluğu olarak verilmiştir. Işığın soğurulması aşağıdaki formül ile ifade edilir.

$$A = \ln\left(\frac{I_o}{I_N}\right) = \varepsilon cL$$

(A) Soğurum; boyutsuz bir ifadedir ve normalde optik yoğunluk olarak adlandırılır. Beer Lambert modeli oksijen doygunluğunu hesaplamada kullanılabilir. Kalp kasılmaları esnasında kanın debisi ve damar içindeki oksijen doygunluğu değişmektedir. **Hata! Başvuru kaynağı bulunamadı.**'de kanın akışı sırasında ışığın emilimini gösteren temsili şekil verilmiştir.



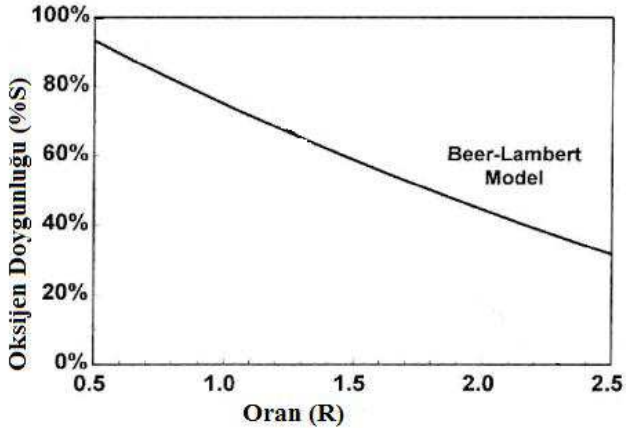
Şekil 2: Kanın akışı sırasında ışığın emilimi

Değişen işaretler AC, sabit kalan işaretler DC olarak kabul edilirse; kırmızı işaret kaynağı için AC_R ve DC_R, kızılötesi işaret kaynağı için AC_{IR} ve DC_{IR} olarak gösterilebilir. Normalize hale getirilmiş soğurulma oranı R aşağıdaki denklemle elde edilebilir.

$$R = \frac{AC_R / DC_R}{AC_{IR} / DC_{IR}}$$

Pompalanan kandaki oksijen doygunluğunun ölçülmesi hedeflendiğinden, PPG (photoplethysmogram) [11, 12, 13, 14] tepe noktasındaki oranı ölçmek gerekmektedir.

$\%SpO_2 = K \times R$ olarak hesaplanabilir. Buradaki K değeri kalibrasyon ile belirlenecek sabit değerdir.



Şekil 3: Beer-Lambert Modele göre SpO2 değeri [15]

C. Ölçüm Hataları

Erişkin bir insan kanında dört tür hemoglobin vardır:

1. Oksi hemoglobin (HbO2)

2. Redükte hemoglobin (Hb)

3. Methemoglobin (MetHb)

4. Karboksi hemoglobin (CoHb)

Methemoglobin ve karboksi hemoglobin çok az miktarda bulunmaktadır. Değişik dalga boylarında emilen bu hemoglobin türlerinin miktarları arttığı zaman hatalı doygunluk değerleri elde edilebilir. [16]

Örnek olarak; methemoglobinemi söz konusu ise kırmızı ve kızılötesi ışınlar aynı oranda emileceğinden dolayı yaklaşık %85 doygunluk değeri elde edilecektir. Oysa gerçek Hb doygunluğu daha yüksek bir değerdedir.

Bir diğer hatalı durum ise; arteriyel O_2 doygunluğu ile periferik O_2 doygunluğu arasındaki değişikliğin gecikmiş olarak pulse oksimetreye yansımasıdır. Kulak probları genellikle doygunluk değişikliklerini, parmak problarından daha erken gösterir. Bunun nedeni akciğer-kulak arasındaki dolaşım zamanının daha kısa olmasıdır. Ayrıca parmakta sayısal sinir blokajı, periferik vazokonstriksiyon, hipotansiyon, düşük ortam ısısı SpO_2'de hatalı sonuçlar verir.

Yine SpO_2'de hatalı düşük değerler elde edilme nedenleri arasında şunlar sayılabilir:

- Ortamın çok ışıklı, fazla aydınlık olması

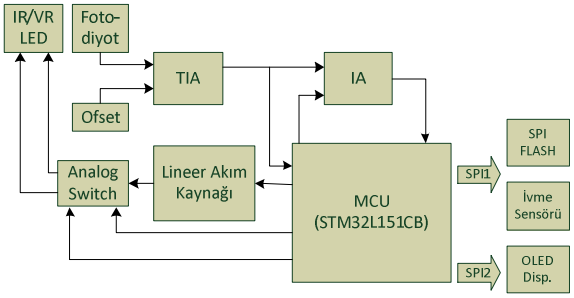
- Metilen mavisi, tırnak cilası, kına
- Hareket
- Ölçüm yapılan ekstremitede venöz pulsasyon olması
- Düşük perfüzyon (kalp debisinde düşme, Hb çok düşük olması, hipotermi, sistemik vasküler dirençte azalma)
- Sensörün yanlış uygulanması (yetersiz temas sonucu optik sızma oluşumu)
- Pigmentasyon (derin pigmentasyon sinyallerde azalma yapar.)
- Yapıştırıcı bantlar (cilt ve sensor arasında sinyallerin geçişini bozar)

III. SPO2 ÖLÇÜMÜ UYGULAMA

A. Mimari Tasarım

Hata! Başvuru kaynağı bulunamadı.'te sisteme ait blok şema verilmiştir. Sistem analog ve sayısal bloklardan oluşmaktadır. Ölçüm tekniği olarak kızıl ötesi ve kırmızı LED'lerin kullanıldığı dolaylı ölçüm tekniğidir.

Nellcor firmasına ait tek kullanımlık problar ile ölçümler alınmıştır.

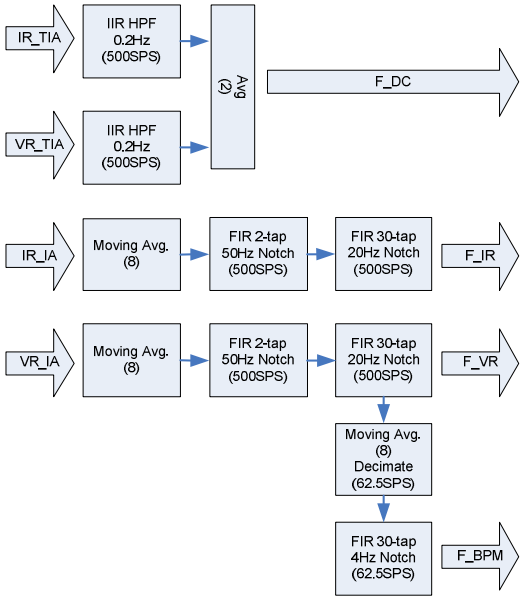


Şekil 4: Sistem blok şeması

Ölçüm kaynağı olarak kırmızı ve kızıl ötesi ışık kaynakları kullanılmıştır. Bir foto diyot ve ön yükselteç yardımı ile deri içinden geçen ve oksijen saturasyonuna göre zayıflamış işaret ölçülür. TIA (Trans-Impedance Amplifier) filtreleme ve işareti güçlendirme işlemini yapar. Eviren bir yükselteç olduğu için işareti IA'dan (Inverting Amplifier) da geçirmek gerekmektedir.

TIA çıkışındaki işaret üzerindeki DC bileşen IIR (Infinite impuls response) filtre kullanılarak filtrelenir ve IA girişine beslenir. Bir ADC (Analog-Sayısal Dönüştürücü) ile örneklenerek, sayısal işaret elde edilir.

IA çıkışındaki işaretler 50-60Hz'lik bir notch filtreden geçirilerek olası şebeke gürültüsü (ışık veya besleme hattından gelebilir) ortadan kaldırılır. Böylece sayısal olarak sentezlenmiş işaret elde edilir. Bu işaret üzerinde tepe bulma, bpm (beats per minute) cinsinden kalp atımı ve tepe noktalarına göre de SpO_2 değeri hesaplanır. **Hata! Başvuru kaynağı bulunamadı.**'te sayısal işaret akışını gösteren blok şema verilmiştir.

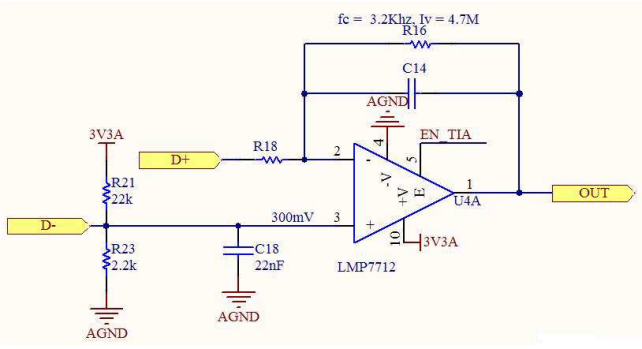


Şekil 5: Sayısal işaret akışı

Devre genelinde güç tüketimi göz önünde bulundurulmuştur. Gerektiğinde, işlemci güç tüketimini azaltmak amacıyla kullanılmayan alt devreleri kapatılabilmektedir. Güç tüketimi ve işlem gücü göz önünde bulundurularak, önerilen tasarımda, ST firmasına ait STM32L mikodenetleyicisi kullanılmıştır.

B. Analog İşaret İşleme Devreleri

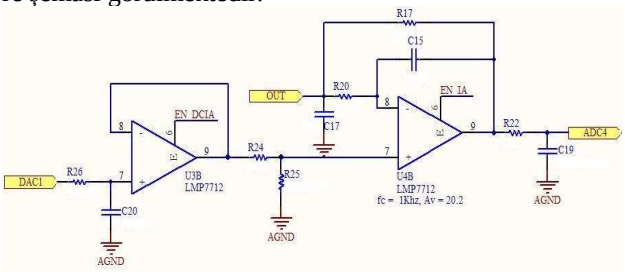
Fotodiyot yükselteci TIA, kızıl ötesi ve kırmızı LED'lerin 100us'lik darbelerini algılayabilmek için kesim frekansı uyumlu bir süzgeç (matched filter) olarak tasarlanmalıdır. Darbe genişliğinin 100us olduğu durumda uyumlu süzgeç köşe frekansı yaklaşık 3.5KHz olmalıdır. İşaretin darbe üstü eğilmeye ve eşik değerinin altına kadar zayıflamaya uğramadan geçebileceği 1. Derece filtre frekansı bu değerin üzerinde olacak şekilde ayarlanmalıdır. **Hata! Başvuru kaynağı bulunamadı.**'da önerilen TIA devresi verilmektedir.



Şekil 6: Transempedans yükselteci

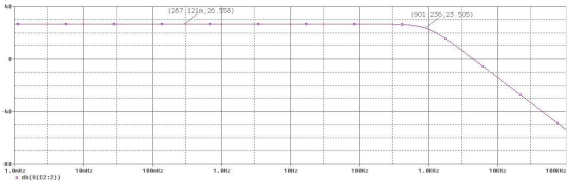
Uyumlaştırıcı devrelere ilave olarak, MFB (multiple feedback) devre topolojisinde 1KHz köşe frekansı olan ve 20 kat kazanca sahip bir fark alıcı kullanımının gerektiği ön görülmektedir. Kazanç bloğu, düşük gerilim seviyesinde (0-300mV) olan transempedans çıkışını yükselterek

mikrodenetleyicinin ADC girişinin daha etkin bir şekilde kullanılabilmesi için tasarlanmıştır. TIA çıkışındaki DC ofset hesaplandıktan sonra mikrodenetleyicinin DAC'ı yardımıyla IA girişinden çıkartılmaktadır. **Hata! Başvuru kaynağı bulunamadı.**'de önerilen fark alıcı yükseltece ait devre şeması görülmektedir.



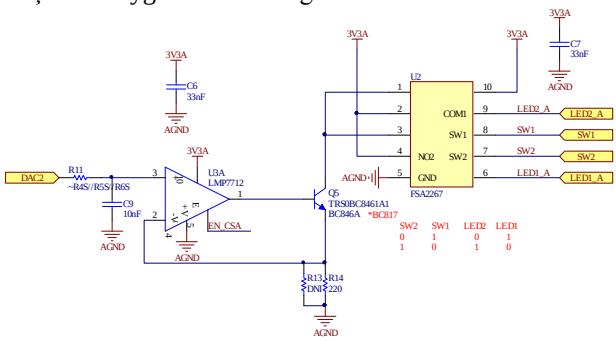
Şekil 7: Eviren yükselteç (Fark alıcı devre olarak)

Hata! Başvuru kaynağı bulunamadı.'de IA'ya ait frekans yanıtı SPICE simulasyonu çıktısı verilmiştir.



Şekil 8: MFB IA devresi frekans yanıtı

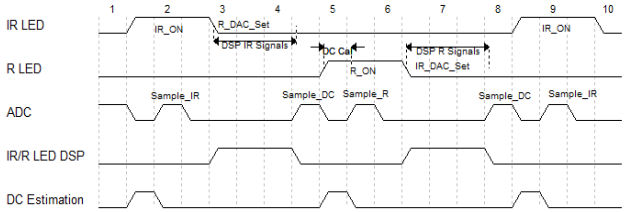
Hata! Başvuru kaynağı bulunamadı.'da kırmızı ve kızılötesi LED'leri sürebilmek için gereken analog anahtar ve akım kaynağı devre şeması verilmiştir. Tek bir foto diyot ile her iki işaret kaynağını aynı anda ölçmek mümkün olmayacağı için, foto diyot zaman paylaşımlı olarak soğurulmuş kırmızı ve kızılötesi ışınları ölçmek için kullanılabilir. Zaman paylaşımı bu analog anahtar yardımı ile yapılmaktadır. ADC girişine gelen işaretin genliği verilen devredeki akım kaynağının verdiği akıma bağlıdır. Devre çıkışını doğrusal bölgede tutmak için devre kullanılmadan önce kalibre edilmesi gerekmektedir. Bu işlem akım kaynağını süren DAC'ın kırmızı ve kızılötesi LED'ler için farklı işaretler uygulaması ile sağlanır.



Şekil 9: Akım kaynağı ve IR/VR anahtarı

C. *İşaret Akışı ve Yazılım Tasarımı*

Kızılötesi ve kırmızı LED'leri sürmek için yaklaşık 10us çözünürlükte tepki verecek bir sisteme ihtiyaç vardır. Bu ihtiyacı sağlamak için mikrodenetleyicinin zamanlayıcı kesmesi kullanılmış ve kesme alt programında bir durum makinesi koşturulmuştur. **Hata! Başvuru kaynağı bulunamadı.**'da durum makinasına ait zamanlama diyagramı verilmiştir.



Şekil 10: İşaret oluşturmak için kullanılan durum makinası zamanlama diyagramı

Mikrodenetleyici üzerinde aynı zamanda RTOS (Real Time Operating System – Gerçek zamanlı işletim sistemi) koşturularak yüksek çözünürlüğüne sahip işleri gerçekleştirmesi için görev atamaları yapılmıştır. RTOS olarak CooCox (CoOS) [17] kullanılmıştır. İşletim sistemleri güvenliği artırmak adına kesmeleri önce yazılımla karşılamakta sırası geldiğinde kesme olduğunu kullanıcısına bildirmektedir. Kullanılan RTOS mikrodenetleyiciye ait kesmeleri donanıma ve kullanıcının isteğine bıraktığı için bu uygulamada büyük esneklik sağlamıştır aynı zamanda hızlı tepki vermesi istenen bir sistem için uygun bir yöntemdir.

Mikrodenetleyicinin kalibrasyon ve konfigürasyonunu sağlamak için bir komut satırı eklenmiştir. Bu komut satırı üzerinden, sentezlenen SPO_2 işaretini PC üzerinde gözlemlemek, kızılötesi/kırmızı LED değerlerinin ölçüm ağırlıklarını güncellemek veya görüntülemek, FIR (Finite Impuls Response) filtrelerin katsayılarını değiştirmek mümkündür. Ayrıca SPI (Serial peripheral interface bus) arayüzü ile okunan değerler "Coffee File System" **Hata! Başvuru kaynağı bulunamadı.** kullanılarak flash belleğe aktarılmaktadır.

Komut satırı ve PC ile gerçek zamanlı işaret akışı RTOS üzerinde bir işleme atanmıştır.

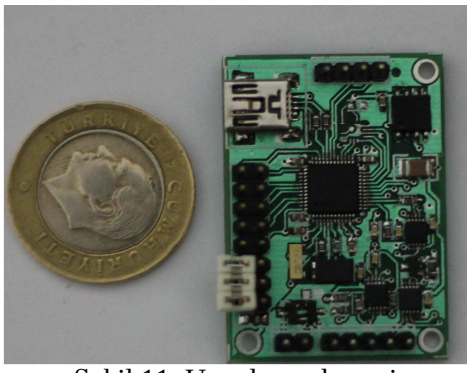
D. *Uygulama Devresi*

Uygulama devresi, iki katlı plaket üzerine Nellcor firmasının tek kullanımlık SPO_2 sensörlerini kullanacak şekilde gerçekleştirilmiştir. Kart üzerinde Bosch firmasına ait bir hareket algılayıcısı (BM180), ST firmasına ait düşük güç tüketimine sahip mikrodenetleyici (STM32L151CB), veri depolanması ve hasta bilgilerinin saklanması için 4MB'lık bir seri flash, Kırmızı/Kızılötesi LED'i seçmek için analog anahtar ve uyumlaştırma devrelerini gerçekleştirmek için düşük güç tüketimli işlemsel yükselteçler kullanılmıştır.

Devre USB (Universal Serial Bus) üzerinden beslenebilmekte ve USB üzerinden bilgisayar ile haberleşebilmektedir.

Hastabaşı monitörlere uyumluluk için devre üzerine bir UART eklenmiştir ve bilgisayar ile arayüzü bu protokol üzerinden sağlanabilmektedir.

Taşınabilir bir SPO_2 cihazı gerçekleştirmek için OLED (Organic light emitting diode) ekranı destekleyen bir SPI portu devre üzerinde bırakılmıştır. Şekil-11'de ilgili devre resimleri verilmiştir. Prototip boyutu daha küçük kılıfta malzemeler kullanılarak küçültülebilir ek devreler eklenebilir. Gerçekleştirilen prototip boyutu 42.8mmx30mm boyutlarındadır.

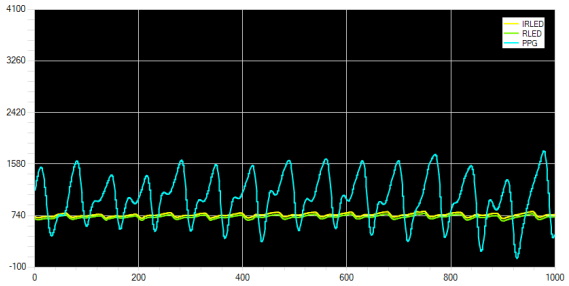


Şekil 11: Uygulama devresi

E. Uygulama Çıktıları

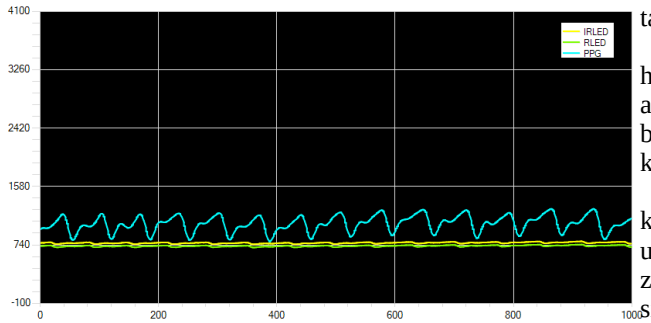
Hata! Başvuru kaynağı bulunamadı.2, **Hata! Başvuru kaynağı bulunamadı.**3, **Hata! Başvuru kaynağı bulunamadı.**4, **Hata! Başvuru kaynağı bulunamadı.**5'te uygulama devresi tarafından alınan işaretlerin aşama aşama işlenmesi gösterilmektedir.

Hata! Başvuru kaynağı bulunamadı.2'de arka plan ışıması bir miktar olmasına rağmen yer değiştirme olmadığı durumda işaretin düzgün elde edilebildiği görülmektedir.



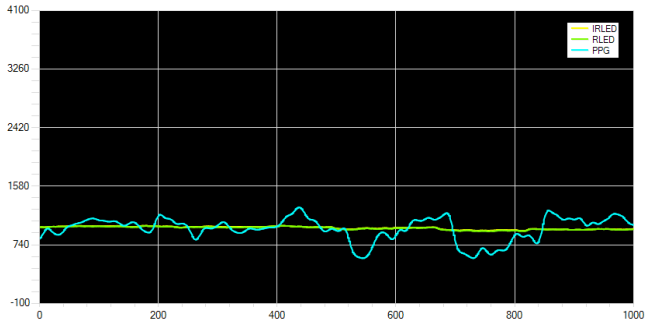
Şekil 12: Arkaplan ışıması ile birlikte

Hata! Başvuru kaynağı bulunamadı.3'te arka plan ışımasının mekanik önlemler alınarak filtrelenmesi sonucu elde edilen çıktı görülmektedir. İşarette kazanç farkından kaynaklanan kaymalar bulunmamaktadır. Bu nedenle işaretin düzgün olarak alındığı söylenebilir.



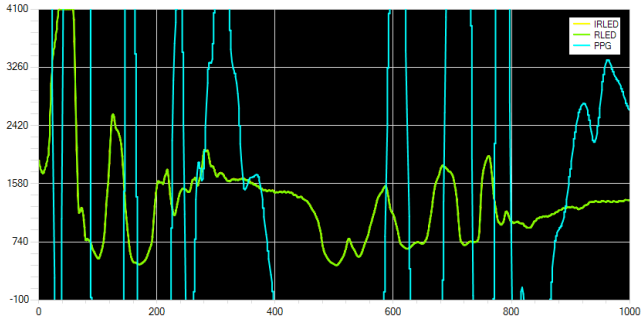
Şekil 13: Arkaplan ışıması azaltılmış durumda

Hata! Başvuru kaynağı bulunamadı.4 incelendiğinde SpO2 ölçümü alınmadığında ve fotodiyot LED'i görmeği durumda oluşan TIA'ın bir miktar kararsızlığı ve arkaplan ışımasının yavaş değişiminden kaynaklanan gürültü işareti görülmektedir.



Şekil 14: IR/VR işaretinin olmadığı durum

Hata! Başvuru kaynağı bulunamadı.5'te probun ucu boşta kalması sonucu değişen arka plan ışımasının değişimi görünmektedir. Bu durumda analog devre doyuma girmekte ve doğrusal bölgede çalışmadığı için devrede ölçüm yapmak mümkün olmamaktadır.



Şekil 15: IR/VR işaretinin olmadığı ve hareketli

IV. SONUÇLAR

Önerilen sistem, Nellcor firmasına ait tek kullanımlık SpO_2 probu ile başarılı bir şekilde gerçeklenmiştir. SpO_2 ölçüm sisteminin hareketten kaynaklanan arka plan ışıması değişimine ve ölçüm probu üzerinden sızan arka plan ışımasına çok duyarlı olduğu görülmektedir. Bu nedenle, mekanik tasarımda da ortamın ışığını filtrelemek için önlemler almak sistemin doğruluğunu arttıracaktır.

Hareket algılama ve filtreleme için bir hareket algılayıcısı (örneğin bir ivme algılayıcısı) kullanarak daha kapsamlı bir cihaz gerçekleştirilmesi mümkündür. Uyarlamalı bir süzgeç kullanarak SPO_2 işaretini zenginleştirme ve aynı zamanda hastanın hareketlerinin takip edilmesi mümkündür.

Diyetisyenler için hem oksidasyon miktarı hem hareketlilik yakılan kalori hesabı için önemlidir. Bir hareket algılayıcı ve kablosuz haberleşme modülü eklenmesi ile birlikte diyetisyenlerin de klinik kullanım dışında kullanabileceği bir ürün haline getirilmesi mümkündür.

Gerçekleştirilen prototip üzerinde kalibrasyon işlemini kolaylaştırmak amacıyla PC tarafında gerçekleştirilecek bir uygulama yazılımı gerekliliği öngörülmektedir. Aynı zamanda taşınabilir bir sistem olarak düşünüldüğünde bu sistemin güç tüketiminin, salt donanım tasarımı ve seçilen malzemelerden ziyade, yazılıma bağımlı olması beklenmektedir.

V. KAYNAKÇA

[1] Cymerman, A; Rock, PB. *Medical Problems in High Mountain Environments. A Handbook for Medical Officers*. USARIEM-TN94-

2. US Army Research Inst. of Environmental Medicine Thermal and Mountain Medicine Division Technical Report. Retrieved 2009-03-05.

[2] Squire JR. Instrument for measuring quantity of blood and its degree of oxygenation in the web of the hand. Clin Sci 1940;4.331-9

[3] Aoyagi T. Pulse oximetry: its invention, theory and future. J Anesth 2003;17.259-66

[4] Nickerson BG, Sarkisian C, Tremper KK. Bias and precision of pulse oximeters and arterial oximeters. Chest. 1988; 93:515–517.

[5] Webb RK, Ralston AC, Runciman WB. Potential errors in pulse oximetry. II. Effects of changes in saturation and signal quality. . Anaesthesia. 1991;96:207–212.

[6] Hannhart B, Haberer JP, Saunier C, Laxenaire MC. Accuracy and precision of fourteen pulse oximeters. Eur Respir J. 1991; 4:115–119.

[7] Severinghaus JW, Naifeh KH. Accuracy of response of six pulse oximeters to profound hypoxia. Anesthesiology. 1987; 67:551–558.

[8] http://www.oximetry.org/pulseox/principles.htm

[9] P. D. Mannheimer, J. R. Casciani, M. E. Fein, and S. L. Nierlich, "Wavelength Selection for Low-Saturation Pulse Oximetry," IEEE Transactions on Biomedical Engineering, vol. 44, pp. 148 - 158, 1997.

[10] Anan Wongjan, Amphawan Julsereewong, and Prasit Julsereewong, "Continuous Measurements of ECG and SpO2 for Cardiology Information System", Proceedings of the International MultiConference of Engineers and Computer Scientists 2009, Hong Kong, Vol II IMECS 2009, March 18 - 20, 2009.

[11] S. Bagha, L. Shaw, "A Real Time Analysis of PPG Signal for Measurement of SpO2 and Pulse Rate", International Journal of Computer Applications, vol. 36 – No:11, pp. 45-50, December 2011

[12] Gazi Maruf Azmal, Adel Al-Jumaily, Moha"med Al-Jaafreh, "Continuous Measurement of Oxygen Saturation Level using Photoplethysmograph Signal", Intl. Conf. on Biomedical and Pharmaceutical Engineering, ICBPE 2006.

[13] M. Shamir, L. A. Eidelman, Y. Floman, L. Kaplan, and R. Pi-zov, Pulse Oximetry Plethysmographic Waveform During Changes in Blood Volume, Br. J. Anaesth., vol. 82, pp. 178-181, 1999.

[14] K. Shelley and S. Shelley, Pulse Oximeter Waveform: Photoelectric Plethysmography,in Clinical Monitoring, Carol Lake, R. Hines, and C. Blitt, Eds.: W.B. Saunders Company, 2001, pp. 420-428.

[15] Dr. Neil Townsend, Medical Electronics. Mischaelmas Term 2001.

[16] Y. Keçil, Ankara Üniversitesi Tıp Fakültesi Anesteziyoloji AD, Ders Notu, http://med.cu.edu.tr/anestezi/ii_cag/new_page_1.htm

[17] www.coocox.org; CooCox CoOS 1.14

[18] N. Tsiftes, A. Dunkels, Z. He, T. Voight, "Enabling Large-Scale Storage in Sensor Networks with Coffee File System", Internatinal Conference on Information Processing in Sensor Networks, pp 349-360, 2009

Mikrodenetleyici Tabanlı Lazer Mesafe Ölçer

Utku Esen[1], Olcay Davut Cabbas[1], Anıl Çelebi[1,2], Oğuzhan Urhan[1,2], Sarp Ertürk[1,2]
[1]Kocaeli Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü, Umuttepe, Kocaeli
[2]PALSIS Elektronik Optik Müh. ve Dan. Hiz. Ltd. Şti.
Kocaeli Üniversitesi Teknoparkı, Kocaeli
e-posta: {utku.esen91, olcay.d.cabbas} @gmail.com , {anilcelebi, urhano,sertur}@kocaeli.edu.tr

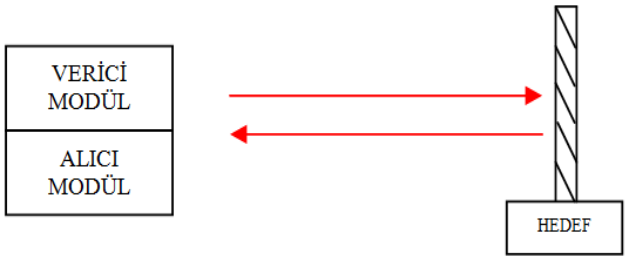
Özetçe— **Bu çalışmada lazer tabanlı mesafe ölçerlerin elektronik sistemlerinin 16-bitlik bir mikrodenetleyici ile yüksek performanslı şekilde gerçeklenmesi sağlanmıştır. Bu tip zamanlamanın hassas olduğu sistemlerde tek yonga üzeri (SoC) veya FPGA tabanlı kontrol birimlerinin kullanılması yaygın olsa da maliyet nedeniyle FPGA yerine mikrodenetleyicilerin kullanımı önemli bir avantaj sağlamaktadır. Bu çerçevede yapılan çalışmalar sonucunda, yüksek performanslı zamanlama modülüne sahip bir mikrodenetleyici kullanılarak metre altı çözünürlükte mesafe ölçümünün yapılabileceği görülmüştür.**

I. GİRİŞ

Lazerlerin günlük hayatta kullanımı her geçen gün artmaktadır. Marketlerdeki barkod okuyuculardan, CD/DVD'lere, tıbbi uygulamalarından, askeri/güvenlik uygulamalarına lazerler birçok farklı alan ve uygulamada kendisine yer bulmaktadır [1,2]. Mesafe ölçümü de lazerlerin etkin şekilde kullanıldığı alanlardan birisidir. Mesafe ölçümü 3-boyutlu modelleme, askeri uygulamalar ve harita/şehircilik alanlarında birçok uygulamaya sahiptir [3].

Literatürde mesafe ölçümü için kullanılan yaklaşımları pasif ve aktif yöntemler olarak iki ana grupta ele almak mümkündür [4]. Pasif yöntemler çeşitli optik ve mekanik düzeneklerle temel trigonometri yaklaşımları ile mesafe ölçümü yapabilmektedir. Ultrasonik, sonar, radar, lazer gibi aktif yöntemlerde ise ölçüm yapmak isteyen cihaz belirli bir işaret yayını yaparak bu yayın sonrasında geri dönen işareti inceleyerek istenen cisme olan mesafeyi tespit edebilmektedir.

Tipik bir aktif lazer mesafe ölçer sistemi Şekil 1'de gösterildiği gibi bir verici ve bir alıcı modülden oluşmaktadır. Verici modül ile belirli bir dalga boyunda gönderilen lazer ışını hedefe ulaştıktan sonra belirli miktarda saçılarak foton olarak geri dönmektedir. Alıcı modül ise dönen ışını tespit etmek amacıyla kullanılmaktadır. Mesafe ölçümünde izlenen temel yaklaşım, lazer ışının ortamda yayılırken yakaladığı hızı temel alarak verici modülden gönderildikten ne kadar süre sonra alıcı modüle ulaştığının (Time of Flight - TOF) hesaplanmasına dayanmaktadır [3,4]. Bu sürenin tespit edilmesinden sonra basit matematiksel hesaplar ile hedefe olan mesafe elde edilebilmektedir. Bu noktada mesafenin hesabı açısından en önemli problem vericinin yayına başladığı an ile alıcının bu lazer ışınını yakaladığı an arasındaki zamanı hassas bir şekilde ölçebilmektir. Yaklaşık olarak 3×10^8m/s hızında yayılmakta olan lazer ışığının hızı dikkate alındığında yüksek hassasiyetli mesafe ölçümü yapabilmek için kullanılacak kontrol sisteminin zamanlamasının oldukça iyi olması gerekmektedir.



Şekil 1 : Lazer mesafe ölçer sistemin çalışma metodu.

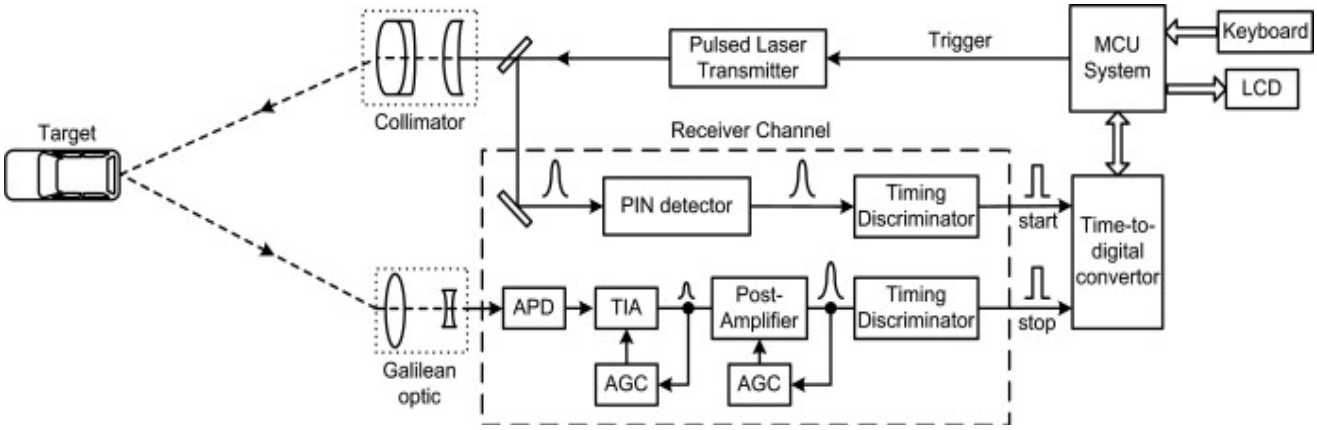
TOF yöntemi ile mesafe ölçme tekniğinin bir avantajı alıcı verici modülünün eş eksenli olmasıdır. Bu nedenle ölçüm doğruluğu mesafeden bağımsızdır [5]. TOF yönteminde farklı yaklaşımlar mevcuttur. Bunlar darbe modülasyonu, genlik modülasyonu ve frekans modülasyonu ile gerçekleştirilen ölçümdür. Darbe modülasyonunda darbenin gönderilmesi ile alınması arasında geçen uçuş süresi hesaplanır [6-8]. Genlik modülasyonunda bir sinüs işareti ile modüle edilmiş huzmenin giden ve yansıyan bileşenleri arasındaki faz farkı ölçülür [9,10]. Frekans modülasyonu yönteminde işaret frekansı sınırlı bir aralıkta (chirp) taranır. Heterodin yöntemi kullanılarak bir orta frekans bileşeni üretilir ki bu bileşenin frekansı hedefin uzaklığına bağlıdır [11]. Bu çalışmada darbe modülasyonu yöntemi tercih edilmiştir çünkü genlik modülasyonuna göre daha yüksek doğruluğa, frekans modülasyonuna göre ise daha uzak mesafeleri ölçebilme kabiliyetine sahiptir [3].

Mesafe ölçümünde uçuş zamanını kullanan yöntemler zamandan sayısala dönüştürücü denilen bir mantığı kullanmaktadırlar. Bu mantığa göre yüksek doğruluklu sayıcı devreleri başlama ve bitiş sinyallerine göre sayısal bir zaman bilgisi üretmektedirler. Günümüzde çok yüksek doğruluklu FPGA'lar kullanılarak tasarlanmış donanım mimarileri bulunmaktadır [11,12]. Bunun yanında bazı mikrodenetleyici sistemler de mevcuttur [13].

Ancak hassas zamanlama ölçümü için FPGA kullanımı maliyet açısından önemli bir sorun olarak öne çıkmaktadır. Bu çalışmada yüksek performanslı zamanlayıcı arayüzüne sahip bir mikrodenetleyici ile daha düşük maliyetli bir lazer mesafe ölçer elektronik kontrol donanımı geliştirilmesi ele alınmıştır.

II. GELİŞTİRİLEN SİSTEM

Lazer mesafe ölçerlerin genel tasarımı ile ilgili bilgiler literatürde bir çok çalışmada bulunmaktadır [14,15]. Verici olarak bir PLD (pulsed laser diode) alıcı olarak ise bir APD (Avalanche photodiode) kullanan tipik bir lazer mesafe ölçerin blok diyagramı Şekil-2'de verilmektedir.



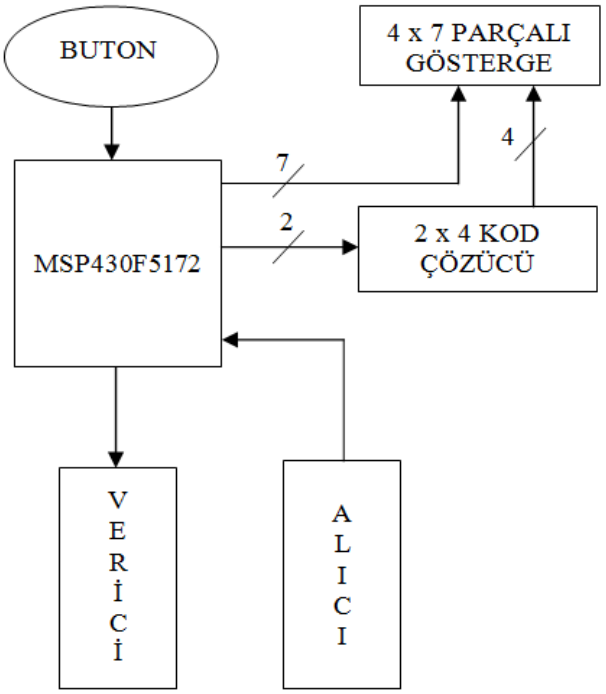
Şekil 2: PLD-APD kullanılan bir mesafe ölçerin blok diyagramı [15]

Şekil-2 incelendiğinde verici kanalda bir mikrodenetleyici ile kontrol edilen PLD ve lazer ışığını odaklayacak bir optik bulunmaktadır. Alıcı kanalda ise gönderilen lazer ışının verici optikten çıkış anı bir PIN diyot ile yakalanmakta, hedefe çarptıktan sonra geri dönen lazer ışını ise APD ve sonrasındaki analog devreler ile tespit edilmektedir. Bu aşamada alıcı kanalda yakalanan bu iki işaretin arasındaki zaman farkı bilgisinin sayısal veriye bir modüle girmektedir. Mikrodenetleyici ise kendisine gelen bu bilgiyi kullanıcı arayüzünde göstermektedir.

Bu çalışmada zaman farkı bilgisinin sayısal veriye dönüştürülmesi işinin mikrodenetleyici ile yapılması ele alınmıştır. Yani önerilen sistemde mikrodenetleyici sadece kullanıcıdan gelen isteğe göre PLD'yi tetikleyip başka bir modül tarafından hesaplanan ölçüm sonucunu kullanıcı arayüzünde göstermeyip, aynı zamanda ölçümün en kritik aşaması olan zaman farkı bilgisinin sayısal veriye çevrilmesi işlemini de yapmaktadır.

Bu çalışmada vericiden gönderilip alıcıdan algılanan lazer işaretinin havada uçuş süresini (ToF) ölçmek için kullanılan donanımın blok şeması Şekil-3'de gösterilmektedir. Sistemin genel kontrolünü ve lazer ışının uçuş süresini hesaplamak için Texas Instruments firmasının 16-bitlik mikrodenetleyici ailesi olan MSP430 serisinden MSP430F5172 kullanılmıştır. Bu mikrodenetleyici yüksek performanslı ve oldukça detaylı şekilde konfigüre edilebilir zamanlayıcısı ile öne çıkmaktadır. Sistemin ölçüm almasını başlatmak için bir buton kullanılmaktadır. Butona basılması sonrasında "bouncing" etkisi de dikkate alınarak vericiye uygun tetikleme işareti yollanır. Tam bu anda mikrodenetleyicinin zamanlayıcı modülü çalışmaya başlar ve hedefe çarpıp alıcıdan geri dönen işaret ile zamanlayıcı modülü durdurulur. Bu aşamadan sonra havada yaklaşık olarak 3×10^8m/s hızında yayılmakta olan lazer ışığının hızı dikkate alınarak basit matematiksek hesaplar ile mesafe tespit edilir ve dört adet 7 parçalı göstergede gösterilir.

Sistemde kullanılan mikrodenetleyici en fazla 25MHz saat işareti ile çalışabilmektedir. Bu durumda tek bir komut çevriminde çalışan komutların çalıştırılması 40ns sürmektedir. Öte yandan lazer ışığının havada yayılma hızı dikkate alındığında 1m'lik bir mesafede ışığın gidiş geliş süresi yaklaşık olarak 6.67ns olmaktadır. Yani bu mikrodenetleyici ile zamanlayıcı modülünün ilgili bazı komutlar sıralı şekilde işletilerek kontrolü durumunda



Şekil 3: Geliştirilen sistemin blok diyagramı

metre altı çözünürlükte mesafe ölçümü yapılabilmesi mümkün olamamaktadır. Bunun temel nedenlerinden biri sıralı şekilde komut işleten mikrodenetleyici ile zamanlayıcının tam olarak lazer ışının gönderildiği anda sıfırlanıp çalışmasını sağlamanın yazılımsal olarak mümkün olmamasıdır.

Verici lazer tetiklendikten sonra zamanlayıcı modülünün çalıştırılması durumunda -ölçülmek istenen mesafeye bağlı olmakla birlikte- muhtemelen daha zamanlayıcı sıfırlanamadan lazer ışını alıcıya ulaşmış olacaktır. Yani mikrodenetleyici yazılımsal olarak aynı anda hem verici ile tetikleme işareti üretip hem de zamanlayıcıyı çalıştıramaz. Benzer bir durum alıcıdan gelen işaretin kesmeler ile algılanması durumu için de geçerlidir.

Bir kesme kullanılarak alıcıdan gelen işaretin algılanması durumunda, kesmeye girmeden önce mikrodenetleyicinin yapması gereken işlemler nedeniyle

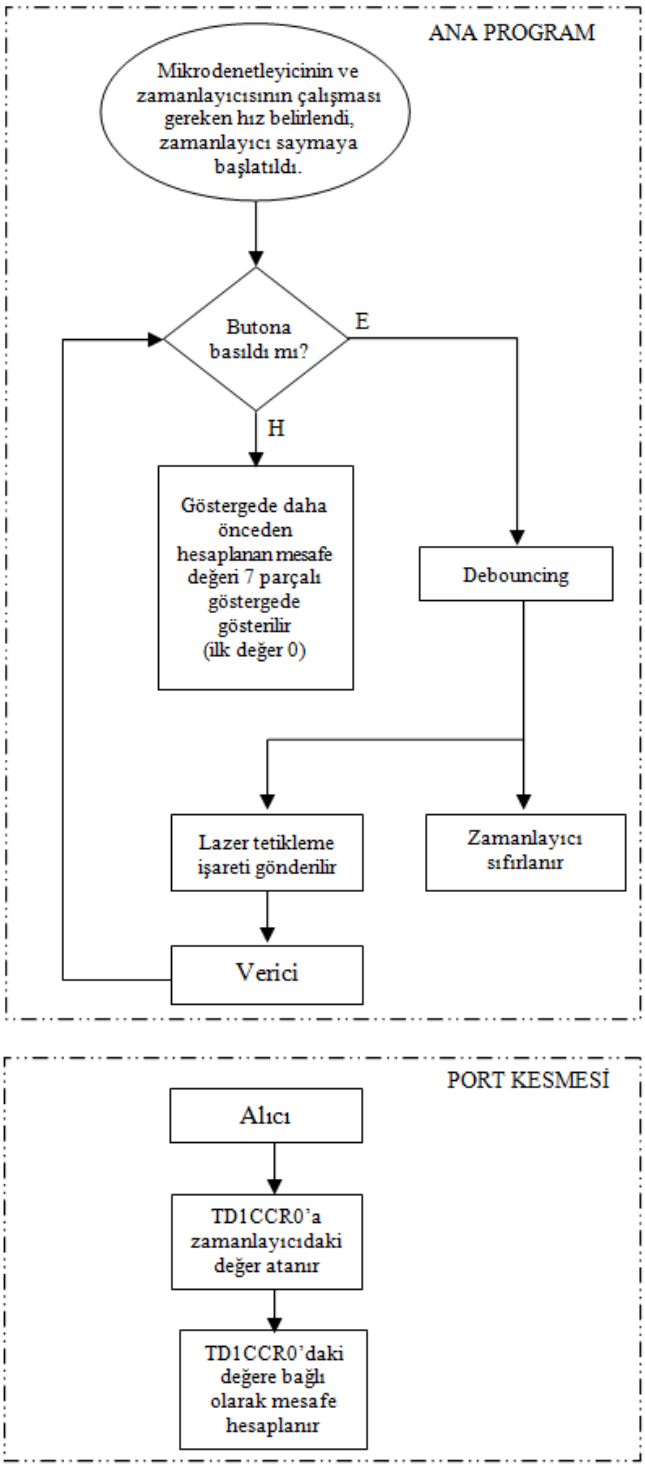
kaybettiği süre (interrupt latency) ölçüm hassasiyetini önemli ölçüde düşürmektedir. MSP430 ailesinde bu süre 12 saat işareti çevrimine kadar çıkabilmektedir. Dolayısıyla lazer ışınının havada uçuş süresinin doğrudan mikrodenetleyicinin yazılım yetenekleri kullanılarak yüksek hassasiyetle ölçülmesi mümkün görünmemektedir. Aslında bu tip lazer mesafe ölçerlerde mikrodenetleyici yerine FPGA'ların tercih edilmesinin en önemli nedenlerinden birisi de budur.

Bu çalışma kapsamında ise yukarıda açıklanan sorunların üstesinden düşük maliyetle gelebilmek için bir MSP430F51×2 ailesi mikrodenetleyicilerin yüksek performanslı ve oldukça yetenekli zamanlayıcı modüllerinden faydalanılmıştır. Bu mikrodenetleyicilerde bulunan D zamanlayıcısı (Timer D - TD) modülleri hem 256MHz'e kadar çalışmayı desteklemekte hem de yetenekli yakalama/karşılaştırma (capture/compare) ve sıfırlama özellikleri ile bu çalışmada ele alınan problem için oldukça uygun bir yapıdadır. Bu zamanlayıcının sıfırlanması ve herhangi bir andaki değerini yakalaması dış dünyadan gelen işaretlerle ek bir yazılımsal desteğe gerek kalmadan kontrol edilebilmektedir. Böylelikle yüksek hassasiyetle lazer ışının havada uçuş süresi hesaplanabilmektedir. Bu amaçla zamanlayıcı olay kontrolü (Timer Event Control - TEC) saklayıcıları kullanılarak sayıcı sıfırlama, saat işareti durdurma ve harici işaretle yakalama özellikleri aktif hale getirilmiştir.

Sistemde butona basıldığında mesafe ölçümü yapılması planlanmıştır. Ayrıca ölçüm sonucunun dört adet 7 parçalı göstergede gösterilmesi hedeflenmektedir. Bu noktada mekanik yapıdaki butondan kaynaklanan "bouncing" etkilerinin giderilmesi gerekmektedir. Aksi durumda lazer farklı zamanlarda birden fazla tetiklenebilir. Bu çalışmada "debcouncing" için donanımsal çözüm yerine yazılımsal bekleme kullanılarak butona her basışın bir kez algılanması sağlanmıştır.

Önerilen sistemde TD modülü mikrodenetleyicinin dahili saat işareti üreticisi (DCO) ile sürekli olarak çalıştırılmaktadır. "Decbouncing" sonrası butona basıldığı algılandığında verici lazer için bir tetikleme işareti uygulanmaktadır. Bu tetikleme işareti aynı anda TD modülünün sıfırlanması amacıyla da kullanılmaktadır. Şekil-2'de görülen yapı incelendiğinde aslında bu işlemin lazeri çalıştıran tetikleme işareti ile değil de lazer çıkışındaki işaretin tekrar elektriksel işarete dönüştürülmesi ile yapılması gerektiği görülmektedir. Ancak tetikleme işaretinin üretilmesinden lazerden ışığın çıkışına kadar geçen süre sabit olarak ele alınabileceğinden bu sürenin bir "off-set" değeri olarak bu çalışmada önerilen yaklaşımla hesaplanan süreye eklenebileceği değerlendirilmiştir. Alıcı kanalda ise alıcıdan gelen işaret hem bir kesme üretilmesine hem de TD'de o anda bulunan değerin TD1CCR0 isimli saklayıcıya aktarılmasını sağlamaktadır. Yani zamanlayıcı modülünün "interrupt latency" adı verilen gecikmeden etkilenmeden ölçüm alabilmesi sağlanmıştır.

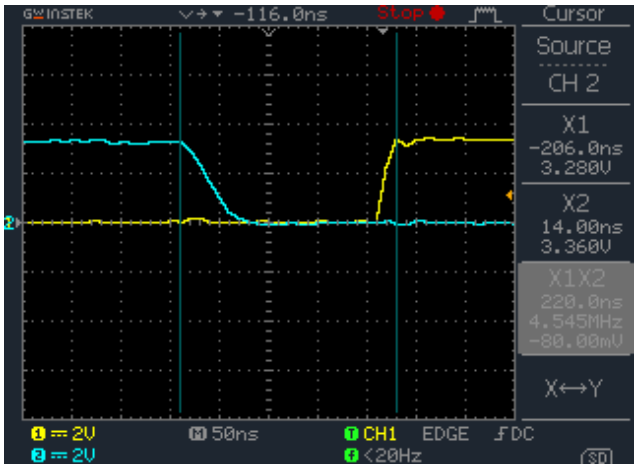
Sistemin çalışması için mikrodenetleyicide koşan yazılımın akış şeması Şekil-4'de verilmektedir. Bu şekilden de görüldüğü gibi zamanlayıcı sürekli olarak çalışmakta, lazer tetikleme ve zamanlayıcı sıfırlama işlemi aynı anda yapılmaktadır.



Şekil 4: Mikrodenetleyicide koşan yazılımın algoritması

III. DENEYSEL ÇALIŞMA

Geliştirilen mikrodenetleyicili lazer mesafe ölçer sisteminin zamanlama hesabının doğruluğunu test edebilmek için öncelikle verici tarafta LS9-40/220-30/110-S10-11 model numaralı 905nm dalga boyunda çalışan PLD modülü, alıcı tarafta ise yine aynı frekansa duyarlı LCSA1500-25 model numaralı APD modülü kullanılmıştır. Bu modüllerin mikrodenetleyici sisteme bağlanması ile yapılan testlerde kullanılan halihazırdaki test optiklerinin



Şekil 5: FPGA'li test düzeneğinden üretilen test işareti

uygun olmaması nedeniyle gönderilen lazer ışını hedefte odaklanamadığından ve benzer şekilde geri dönen ışınlar APD üzerine düşürülemediğinden bu kurulum ile uzak mesafelerde ölçüm alınması mümkün olamamıştır. Ancak yakın mesafelerde alıcı üzerine düşen işaretteki değişim incelenebilmiştir.

Bu durumda çalışmada ele alınan mikrodenetleyicili sistemin performansını ölçebilmek için alıcı ve verici işaretleri simüle edebilecek düşük periyotlu işaretler geliştirilen sistemin verici ve alıcı uçlarına uygulanmıştır. Şekil-5'de FPGA'li test sisteminden üretilen 220ns'lik örnek ölçüm başlangıç ve bitişini gösteren işaretler verilmiştir.
Bu işaretlerden düşen kenarlı olan sayıcının çalışmasını başlatırken (PLD'nin tetiklenmesi simüle edilmektedir), yükselen kenarlı işaret, sayıcıdaki değerin TD1CCR0 isimli saklayıcıya aktarılmasını sağlamaktadır. (APD ile lazer işaretinin alınması simüle edilmektedir).

Tablo-1'de farklı uzunluktaki test işaretleri için 10 farklı test için TD1CCR0 saklayıcısındaki değerlerin en küçük, en büyük değeri ile ortalaması verilmektedir. Bu tablodan görüldüğü gibi tetikleme işaretleri arasındaki süre ile ilgili saklayıcıda elde edilen değer arasında doğrusal sayılabilecek bir ilişki bulunmaktadır. Saklayıcıda elde edilen değerlerin farklılık göstermesinin temel nedeninin mikrodenetleyicinin düşük performanslı dahili osilatörünün (DCO - Digitally Controlled Oscillator) zamanlayıcı devresi için saat işareti kaynağı olarak kullanılması olduğu değerlendirilmektedir.

Tablo 1: Farklı test süreleri için TD1CCR0 değerleri

Süre	En küçük değer	En büyük değer	Ortalama
64ns	16	21	17,5
120ns	27	31	28,3
220ns	54	60	55,9
2µs	592	600	596,4

IV. SONUÇLAR

Bu çalışmada metre altı hassasiyete ölçüm yapabilen lazer mesafe ölçerlerin elektronik sistemlerinin kontrolü için FPGA yerine mikrodenetleyici tabanlı bir sistem önerilmiştir. Geliştirilen sistem temel olarak yüksek performanslı bir zamanlayıcı modülüne sahip mikrodenetleyici içermektedir. Bu sistem ile yüksek performanslı bir saat işareti üreticisi kullanılması durumunda düşük maliyetli ve metre altı hassasiyetle ölçüm yapabilen bir lazer mesafe ölçer kontrol sistemi oluşturmak mümkün olmaktadır.

TEŞEKKÜR

Bu çalışma TÜBİTAK TEYDEB tarafından 7120172 numara ve Elektro-optik Mesafe Ölçer isimli proje kapsamında desteklenmiştir.

KAYNAKÇA

[1] Online: Laser in Medicine, Surgery, Dentistry, and Veterinary, http://www.lasermedico.ch/

[2] Richard J. Dunn, "Operational Implications Of Laser Weapons", Northrop Grumman Co., 2005.

[3] Kilpela, A., "Pulsed Time of Flight Laser Range Finder Techniques for Fast, High Precision Measurement Applications", PhD Thesis, Electrical and Information Engineering, University of Oulu, 2004, Finland.

[4] S. Chen, "A Single Chip Real-Time Range Finder", PhD Thesis, Texas A&M University, 2003.

[5] T.C. Strand, "Optical three-dimensional sensing for machine vision", Optical Engineering, vol. 24, no. 1, pp. 33-40, 1985.

[6] Goldstein, B.S., Dalrymple, G.F., "Gallium arsenide injection laser radar," Proc. IEEE, vol.55, no.2, pp. 181- 188, Feb., 1967.

[7] Koechner, W., "Optical ranging system employing a high power injection laser diode", IEEE Trans. Aerosp. Electron. Syst., vol. AES-4, no. 1, pp.81 - 91 , 1968.

[8] Määttä, K., Kostamovaara, J., and Myllylä, R., "Profiling of hot surfaces by pulsed time-of-flight laser range finder techniques", Applied Optics, vol. 32, no 27, pp. 5334 - 5347, Sep. 1993.

[9] Payne, J.M., "An Optical Distance Measuring Instrument", Review of Scientific Instruments, vol. 44, no. 3, pp. 304-306, 1973.

[10] Zahid M., Smith J.S., Lucas J. "High-frequency phase measurement for optical rangefinding system", IEE Proc. Sci. Meas. Technol., vol. 144, no. 3, pp. 141-148, May 1997.

[11] Hulme K.F., Collins B.S., Constant G.D., Pinson J.T., "A CO2 laser rangefinder using heterodyne detection and chirp pulse compression", Optical and Quantum Electronics, vol. 13, pp. 35-45, 1981.

[12] Actel's Application Note: "Laser Range Finder Using Actel's Axcelerator FPGA", 2004.

[13] Johannesen, B., "Low Cost Scanning Laser Rangefinder", Submitted for the Degree of Bachelor of Engineering in the Division of Electrical Engineering, University of Queensland, 2001.

[14] H.N. Burns, C.G. Christodoulou, G. D. Boreman, "System design of a pulsed laser rangefinder", Optical Engineering, vol. 30, no. 3, pp. 323-329, 1991.

[15] F. Zhu, K. Gong, Y. Huo, "A wide dynamic range laser rangefinder with cm-level resolution based on AGC amplifier structure", Infrared Physics & Technology, vol. 55, no 2-3, pp. 210-215, 2012.

FPGA Üzerinde MicroBlaze Tabanlı Video İşlemci Tasarımı

Abdulkadir Koçdoğan, Ramazan Yeniçeri ve Müştak Erhan Yalçın
İstanbul Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Maslak, 34469, İstanbul
e-posta: {kocdogan, yenicerir, mustak.yalcin}@itu.edu.tr

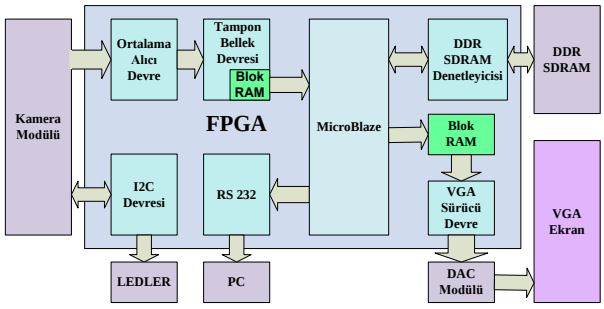
Özetçe—**Projede, Sahada Programlanabilir Kapı Dizisi (Field Programmable Gate Array – FPGA) üzerinde donanım ve yazılımın birlikte tasarımıyla MicroBlaze tabanlı bir video işlemci gerçeklenmesi amaçlanmıştır. FPGA üzerinde gerçeklenen bir donanım yardımıyla görüntü dizisi bir CMOS sensör modülünden alınmakta, 32 bitlik MicroBlaze mikroişlemcisinde işlenmekte ve FPGA üzerinde gerçeklenen ek bir donanım yardımıyla işlenen görüntü dizisi VGA ekranda görüntülenmektedir. Önerilen donanım, yazılımın esnekliğiyle karmaşık video işleme algoritmalarının gerçeklenmesine izin vermektedir.**

I. GİRİŞ

Donanımın paralel işlem yapma özelliğinden dolayı sonuca hızlı ulaşabilmesi ve yazılımın tasarım esnekliği sağlaması, gömülü sistemlerde donanım ve yazılımın birlikte tasarımını avantajlı hale getirmiştir. Görüntü işleme uygulamalarında da büyük boyutlardaki verilerle hızlı bir şekilde sonuca ulaşmak önemli hale gelmiştir. Literatürde FPGA üzerinde donanım ve yazılımın birlikte tasarlandığı birçok görüntü/video işlemci mevcuttur [1, 2]. Bu bildirideki tasarımda, donanımlar Verilog HDL ile tasarlanmıştır. Yazılımı çalıştıran mikroişlemci olarak da normalde FPGA üzerinde bulunmayıp istenildiğinde eklenebilen MicroBlaze mikroişlemcisi kullanılmıştır. Oluşturulan sistemde görüntü kameradan FPGA'ya bir donanım yardımıyla alınır. Çözünürlüğü düşürülerek MicroBlaze'e verilir. Alınan görüntüler MicroBlaze üzerinde çalışan program yardımıyla işlenir ve çıkış donanımına verilir. Görüntü buradan FPGA dışındaki bir DAC modülünden geçirilerek VGA formatında çıkış görüntüsü oluşturulur [3]. Dolayısıyla görüntü üzerinde yapılan işlemenin sonucu hemen ekranda görülebilmektedir. Sistemin doğru bir şekilde çalıştığını göstermek için iki tane görüntü işleme algoritması uygulanmış ve sonuç başarılı bir şekilde görülmüştür.

II. FPGA ÜZERİNDE SİSTEMİN TASARIMI

FPGA üzerinde tasarlanan sistemde kullanılan donanım blokları: Ortalama Alıcı Devre, Tampon Bellek Devresi, Blok RAM, MicroBlaze işlemci çekirdeği, VGA Sürücü Devre, I2C Devresi, UART, DDR SDRAM Denetleyicisidir. MicroBlaze işlemcisi yazılımın çalıştırılması için, geliştirme kartı üzerinde bulunan 64 MB büyüklükteki DDR SDRAM'i kullanır. FPGA üzerinde tasarlanan sistem ve bu sistemin dışarıyla bağlantılarını gösteren blok diyagram Şekil 1'de verilmiştir.



Şekil 1: Görüntü İşlemcinin Blok Diyagramı.

A. *Donanım Tasarımı*

Kameradan alınan görüntü FPGA içerisinde öncelikle Ortalama Alıcı Devre'ye gelir. Burada, kameradan 320x240 çözünürlükte alınan görüntünün çözünürlüğü 160x120 çözünürlüğe düşürülür. Böylece FPGA içerisinde bulunan Blok RAM bir çerçeve görüntünün alınması için yeterli hale gelmiş olur. Görüntü, Ortalama Alıcı Devre'nin çıkışından Tampon Bellek Devresi'ne geçerek Blok RAM'e yazılır. Görüntü Blok RAM'den MicroBlaze tarafından alınır. MicroBlaze yazılımla görüntüyü işler ve çıkış için ayrılan diğer bir Blok RAM'e yazar. Yazılımın kullandığı ara değişkenler DDR SDRAM'de tutulur. I2C Devresi kamerayla FPGA'nın haberleşmesini, UART da FPGA ile bilgisayarın haberleşmesini sağlar. Görüntü VGA Sürücü Devre tarafından Blok RAM'den okunur. VGA Sürücü, yatay ve düşey senkronizasyonu ayarlayarak görüntüyü FPGA dışında bulunan bir DAC modülüne verir. DAC modülü kullanılmasının nedeni, FPGA'dan çıkan her bir pikselin 8 bit sayısal veri olmasına karşılık VGA ekranın 0 – 0.7 V arası analog bir işaretle sürülmesidir. Buna uygun olarak hesaplamalar yapılmış ve DAC modülü tasarlanmıştır. Son olarak DAC modülünün çıkışından ekrana gönderilen görüntü, ekranda gri tonda görülebilmektedir.

B. *Yazılım Tasarımı*

Sistemde MicroBlaze yazılımının çalıştırıldığı bellek olarak kart üzerinde bulunan DDR SDRAM seçilmiştir. Bunun nedeni, görüntü işleme algoritmalarında büyük boyutlu görüntülerden çok miktarda saklama ihtiyacı olabilmesidir. FPGA içerisindeki Blok RAM'ler bu ihtiyaç için yetersiz kalır. Şekil 2'de kameradan alınan ve üzerinde işleme yapılmadan ekrana gönderilen bir görüntü

gösterilmektedir. Burada kameranın önünde bulunan nesneler (bir bilgisayar ekranı, bir bilgisayar kasası ve bir dolap) ekranda açıkça görülmektedir.

Şekil 2: Kameradan alınan görüntünün ekrandaki gri seviye görüntüsü.

Oluşturulan sistem üzerinde kameradan alınan görüntünün gri tonda ekrana verilebildiği görüldükten sonra, zamansal ortanca filtre ve histogram algoritması olmak üzere iki tane görüntü işleme algoritması uygulanmıştır. Bu uygulamalardan başarılı sonuçlar alınarak sistemin doğru bir şekilde çalıştığı gösterilmiştir. Bu bildiride ortanca filtre uygulaması hakkında bilgi verilmekte ve elde edilen sonuçlar sunulmaktadır.

1) *Zamansal Ortanca Filtre:* Bu filtre birçok görüntünün art arda gelmesiyle oluşan videolara uygulanır. Bir videoda art arda gelen görüntülerin her birinin aynı koordinatlarındaki piksel değerleri alınır ve sıralanır. Bu sıralamada ortadaki değer yeni bir görüntü çerçevesinin aynı koordinatlarındaki piksel değeri yerine konur [4]. Yapılan bu işlem Denklem 1'deki gibi ifade edilir.

$$G(i,j) = \text{Ortanca}\{F1(i,j), F2(i,j), \ldots\ldots, Fn(i,j)\}. \quad (1)$$

Sonuçta elde edilen görüntüde videodaki hareketsiz arka plan dururken hareketli kısım silinmiş olur. Projedeki ortanca filtre uygulamasında kameranın karşısında Şekil 2'deki görüntü durmaktadır. Kameradan 21 kare görüntü video modunda alınmıştır. Bu esnada kameranın önünden bir kişi geçmektedir. Yani bu 21 karenin her birinde kişi, farklı konumlarda bulunmaktadır. Bu karelerden iki tanesi Şekil 3'te gösterilmiştir. Bu görüntü dizisi alınıp DDR SDRAM'de saklanmış ve ortanca filtre algoritması uygulanmıştır. Sonuçta elde edilen görüntü Şekil 4'te verilmiştir. Uygulanan filtre ile arka plan görüntüde korunmuş, hareketli olan kısım silinmiştir.

Şekil 2'deki orjinal arka plan görüntüsüyle Şekil 4'teki filtre sonucu elde edilen görüntü karşılaştırıldığında aralarındaki farkın çok az olduğu görülmektedir.

III. SONUÇLAR

Sonuç olarak, donanım ve yazılımın birlikte tasarımıyla FPGA üzerinde video işlemci tasarımı yapılmıştır. Kameradan alınan görüntü FPGA içerisinde yazılımla işlenerek ekrana gri tonda gönderilebilmiştir. Böylece uygulanan görüntü işleme algoritmalarının sonucu ekranda görülebilmektedir. Burada iki basit uygulamayla sistemin doğru bir şekilde çalıştığı gösterilmiştir. İleriki çalışmalarda, bu sistem kullanılarak daha üst seviye görüntü işleme uygulamaları yapılabilecektir.

(a)

(b)

Şekil 3: Zamansal ortanca filtre uygulamasındaki videodan iki kare.

Şekil 4: Zamansal ortanca filtreden elde edilen görüntü.

IV. KAYNAKÇA

[1] Sang Jun Lee, Dae Ro Lee, Seung Hun Jin, Jae Wook Jeon, Key Ho Kwon, "MicroBlaze based image processing system using IEEE1394a," *Control, Automation and Systems, 2007. ICCAS '07. International Conference on*, pp.644-648, 17-20 Oct. 2007.

[2] Alt, N., Claus, C., Stechele, W., "Hardware/software architecture of an algorithm for vision-based real-time vehicle detection in dark environments," *Design, Automation and Test in Europe, 2008. DATE '08*, pp.176-181, 10-14 March 2008, doi: 10.1109/DATE. 2008.4484682.

[3] Koçdoğan, A., "FPGA Üzerinde MicroBlaze Tabanlı Görüntü İşlemci Tasarımı", *Lisans Tezi, İ.T.Ü. Elektrik Elektronik Fakültesi*, İstanbul, 2012.

[4] Hung, M.H., Pan, J.S. and Hsieh, C.H., "Speed Up Temporal Median Filter for Background Subtraction," *Pervasive Computing Signal Processing and Applications (PCSPA), First International Conference*, pp.297-300, 2010.

ARM İŞLEMCİLİ GELİŞTİRME KARTI TASARIMI

Ahmet ALBAYRAK
Sinop Üniversitesi
Ayancık Meslek Yüksekokulu
Ayancık, 57400 Sinop
e-posta: aalbayrak@sinop.edu.tr

İsmail MERSİNKAYA
Sinop Üniversitesi
Ayancık Meslek Yüksekokulu
Ayancık, 57400 Sinop
eposta: imersinakya@sinop.edu.tr

Kemal MAŞALI
Sinop Üniversitesi
Ayancık Meslek Yüksekokulu
Ayancık, 57400 Sinop
eposta: kemalmasali@gmail.com

Özet— **Geliştirme kartları genel olarak; uygulama ağırlıklı derslerde, teorik olarak verilen bilginin uygulanarak pekiştirilebilmesi için üzerinde farklı deneylerin yapılmasında kullanılmaktadır. Her deney için ayrı bir devre hazırlamak yerine geliştirme kartı kullanılarak çok hızlı şekilde çalışma yapılabilmektedir. Sinop Üniversitesi Ayancık Meslek Yüksekokulu'nda uygulamalı derslerde deney amacı ile kullanılmak üzere geliştirme kartı tasarlanmıştır. Bu kart üzerindeki işlemci; Keil uVision, TrueStudio ve Ewarm yazılım geliştirme platformlarından herhangi biriyle programlanabilir. Tasarlanan geliştirme kartı ARM Cortex M4 32 bit işlemci barındırmaktadır.**

I. GIRIŞ

Günümüzde ARM (Advanced Risc Machine) işlemciler cep telefonları, medya oynatıcılar, oyun üniteleri gibi enerji tasarrufunun önemli olduğu uygulamalarda özellikle tercih edilmektedir. ARM işlemciler genel olarak Risc (Reduced Instruction Set Computers) mimarisine sahip komut kümesini kullanmaktadır. Komutlar daha az olduğundan yapılmak istenen iş daha kısa sürede tamamlanır. Diğer işlemcilere nazaran ARM işlemciler genel olarak Harvard mimarisine sahiptir. Harvard mimarisi kodların ve verilerin ayrı belleklerde saklandığı yapı olarak ifade edilir.

ARM işlemciler bugün 32 bit olarak üretilmektedir. İşlenecek komutların 16 bitlik ya da 32 bitlik olmasına bağlı olarak Thumb adı verilen komut çalıştırma mimarisi bulunmaktadır. Thumb mimarisi 8 bitlik, 16 bitlik işlemlerde devreye girerek 32 bitlik kaydedicilerin kullanılması engellenir. Bu da tabi ki enerji tasarrufu sağlamaktadır[1].

Elektronik dizayn sektöründeki hızlı gelişmeler ile entegre devreler daha geniş çaplı, minyatür ve yüksek hızlı olarak gelişme göstermektedir. Yüksek hızda dijital sistem çağının gereği olarak artık sinyal bütünlüğü (Signal Integrity) ve elektromanyetik uyum (Electromagnetic Compatibility) konuları göz ardı edilmemelidir. Yapılan bu çalışmada SI ve EMC'nin temel teorilerine dayanarak Altium Designer 6'da simülasyon yapılmış ve istenen sonuçları verdiği görülmüştür[2].

Gömülü sistem, bir ya da birkaç atanmış görevi yerine getirmek üzere tasarlanmış özel bir bilgisayar sistemidir. Gömülü işlemci olarak STM32F103CB kullanılan bir çalışmada sıcaklığa bağlı olarak gerçekleşen eğim ölçülmüştür. İvme sensörünün çıkışının filtrelenmesi ile okunan eğim değerine göre pozisyon belirlenmiştir[3]. Günümüzde mikrodenetleyici sistemler hemen her alanda kullanılmaktadır. ARM işlemcili mikrodenetleyiciler çeşitli firma isimleri ile piyasaya sürülmektedir. NXP(Next eXPreience) fiması tarafından üretilen LPC1768 ARM işlemcili mikrodenetleyici ile bulanık mantık tabanlı pozisyon kontrolü yapılan otomatik test yatağı yapılmıştır[4]. ARM işlemcili mikrodenetleyiciler endüstriyel uygulamalarda denetleyici olarak kullanılmaktadır. MCB2300 Keil bordu ile DC motor kontrolü yapmak için PID (Proportional Integrated Derivated) denetimli hassas hız kontrolü yapılmıştır[5].

Altium Designer profesyonel bir baskı devre çizim ve elektronik devre şeması düzenleme programıdır. Bu program ile elektronik devre şemalarını çizebilir, PSpice tabanlı modelleri kullanarak simulasyon yapabilir, tek ve çok katlı baskı devreler çizilebilmektedir.

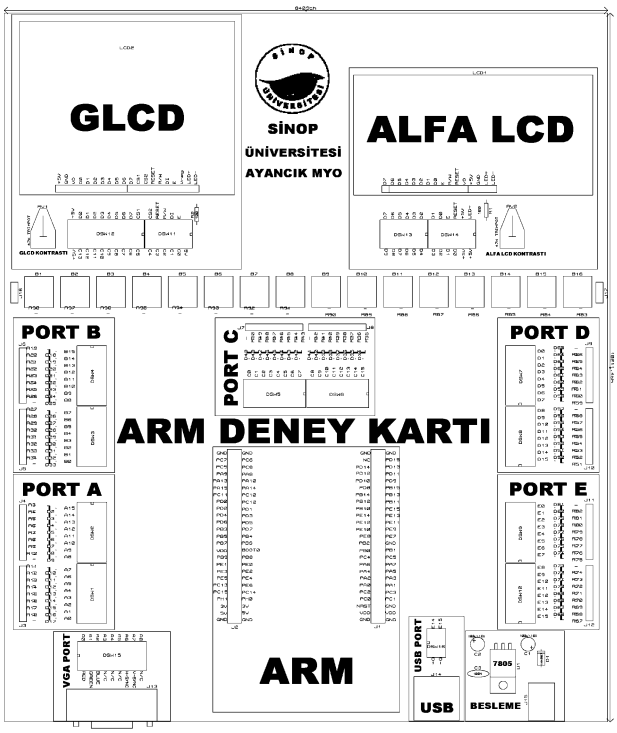
Gömülü sistemler üzerinde kullanılan programlama dillerinden en etkili olanı ve en çok kullanılanı C'dir. C dili hem assembly gibi donanıma yakın hem de üst seviye dillerin özelliklerine sahip bir dil olduğundan gömülü sistemlerin programlanmasında vazgeçilmez bir programlama aracıdır. Otomobillerde alanındaki standart işletim sistemi olan Autosar için yazılım geliştirilen uygulamada C dili kullanılmıştır[6].

II. MATERYAL VE METOT

Geliştirme kartı ile yapılan uygulamalar, elektronik alanındaki öğrencilerinin bilişsel süreçlerinde işlemci, komut yazımı ve yazılan komutların çalıştırılması konularının daha etkin ve verimli olmaktadır. Bu kart Sinop Üniversitesi Ayancık Meslek Yüksekokulu'nda C programlama dili temelli derslerde öğrencilerin teorik olarak öğrendiklerini örnek uygulamalarla pekiştirmek için geliştirilmiştir.

Tasarlanan geliştirme kartı Altium Designer 10'da çift katmanlı olarak tasarlanmıştır. Simulasyon üzerinde kodlar denenmiştir. Kartın üzerinde kendi besleme ünitesi, USB (Evrensel Seri Veriyolu) veri haberleşme portu, ARM işlemci üzerinde bulunan her bir 16 bitlik portların kullanımını sağlayan 5 adet I/O (Giriş/Çıkış) bağdaştırıcısı (A, B, C, D, E), VGA (Video Grafik Dizisi) portu, butonlar, Grafik LCD (Sıvı Kristal Gösterge) ve alfanumerik LCD bağlantıları bulunmaktadır. Ayrıca portları aktif veya pasif olarak konumlarını değiştirebilmek amacı ile DIP-switch kullanılmıştır. Kart üzerinde yüzey montaj led ve dirençler bulunmaktadır.

Geliştirme kartı üzerinde ARM Cortex-M4 32 bit işlemci bulunmaktadır. İşlemcinin kayan noktalı sayılarla işlemleri daha doğru ve hızlı yapması için FPU (Floating Point Unit) ünitesi bulunmaktadır. 1 MB Flash belleğe sahiptir. 168 Mhz çalışma frekansına sahiptir. Dijital sinyal işleme özelliğine sahip işlemci DSP (Digital Signal Processing) uygulamalarında rahatlıkla kullanılabilir. Her bir ünite için kodlar Keil uVision yazılım platformunda yazılmış ve geliştirme kartı denenmiştir. Geliştirilen deney kartı 26x 21,5 cm ebatlarındadır. Şekil 1'de ARM geliştirme kartının blok diyagramı verilmektedir.



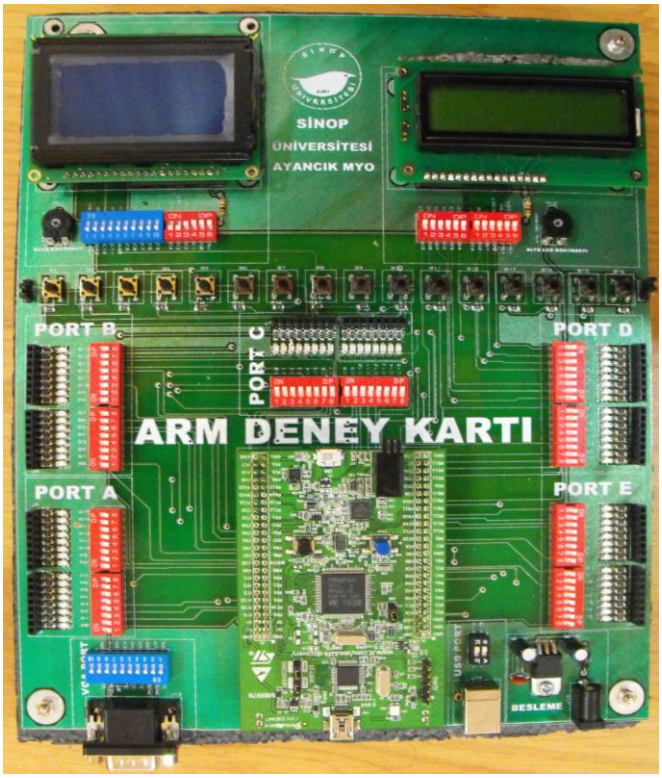
Şekil 1. ARM Geliştirme Kartı Blok Diyagramı.

Blok diyagram üzerinde her birim ayrı ayrı belirtilmektedir. Kartın kendisine ait besleme devresi 5V DC gerilim için oluşturulmuştur. Bu devre, programlamadan sonra ARM Cortex'in USB ile PC bağlantısının kesilmesi durumunda haricen 9V-12V DC gerilim ile çalıştırılabilmesini sağlamaktadır.

Kart üzerinde bulunan diğer bir birim B-Type USB bağlantı soketidir. Bu bağlantı soketi, ARM Cortex'in E14 ve E15 uçlarına bağlanmıştır. PC üzerinde oluşturulacak bir yazılım ile USB bağlantılı uygulamaların yapılmasına olanak sağlamaktadır. Geliştirme kartı ARM Cortex için soket barındırmaktadır. Bu soket yardımı ile ARM Cortex takılıp çıkarılabilmektedir.

Kartın tasarımında görsel grafik uygulamalarının yapılabilmesi için 1 adet Grafik LCD ve 1 adet Numerik LCD bağlantısı mevcuttur. Bu bağlantılar sayesinde farklı birçok LCD uygulamasının yapılmasına olanak sağlamaktadır. 16 adet buton ile LCD uygulamalarında ve diğer uygulamalarda farklı çalışmalar yapmak mümkündür.

Deney kartı üzerinde bağlantı noktalarının öncesinde SMD led ve dirençler yerleştirilmiştir. Bu sayede istenilen led diyot uygulamaları yapılabilmektedir. Ayrıca ledler iptal edilerek her bir bağlantı ucu daha farklı uygulamalarda kullanılabilmektedir.



Şekil 2. Arm Geliştirme Kartı.

Kartın üzerinde 1 adet VGA bağlantı noktası eklenmiştir. VGA bağlantı noktası, tasarımın kolay ve amaca uygun kullanılabilirliği için Cortex'in A0 ile A9 bağlantı uçları ile ilişkilendirilmiştir.

Uygulamaların çeşitliliği ve tasarımın görünümü açısından her bir bağlantı noktası, LCD, VGA ve USB bağlantı noktalarından önce dip-switch (dip anahtarı) kullanılmıştır. Bu dip anahtarları sayesinde istenilen bağlantı noktası ile istenilen uygulamanın yapılması daha kullanılabilir olmaktadır.

Deney kartının üzerine ayrıca pull-up ve pull-down bağlantı uçları da eklenmiştir. Bu uçların değiştirilmesi ile buton uygulamalarında yükselen kenar veya düşen kenar tetiklemeli işlemler, jumper (atlama) soketleri ile kolaylıkla yapılabilmektedir.

Örnek bir uygulama olan led-blinking (durum tersleme) uygulamasının pseudo kodları şöyledir;

```
main()
{
        while(1)
        {
        GPIOB->OBR= 0x00000000;
        bekle();
        GPIOB->OBR= 0x0000F000;
        bekle();
        } }
```

```
void bekle()
{
        for(int i=0;i<0x50000;i++);
}
```

III. SONUÇLAR

Uygulamalı derslerde öğrencilerin teorik bilgilerini pekiştirmek ve bilgilerin kalıcılığını sağlamak için bu deney kartı kullanılabilir. ARM işlemcilerle ilgili uygulamalar dünyada oldukça yaygın hale gelmiştir. Ülkemizde de enerji tasarrufu konuları önem kazanmakta ve daha az enerji ile daha yüksek performansa yönelik çalışmalar yapılmaktadır.

Tasarlanan geliştirme kartı ARM işlemcili sistem uygulamalarında ve C programlama dili temelli gömülü sistem uygulamalarında da kullanılabilir. Dokunmatik panel (Touch Screen), GSM (Mobil İletişim için Küresel Sistem) modülü, GPS (Küresel Konumlama Sistemi) modülü, Micro SD card socket (MiniSD hafıza kartı yuvası), Ethernet Port, Ses/Mikrofon giriş/çıkışları ve ivme sensörünün de eklenmesi ile geliştirme kartı farklı alanlardaki uygulamalarda da kullanılabilir.

ARM geliştirme kartı, yeni nesil bir işlemcinin kolay ve hızlı bir şekilde öğrenilerek yeni uygulamaların geliştirilmesini sağlamak üzere tasarlanmıştır. Tasarlanan kart temel düzeydeki gömülü sistem uygulamalarını desteklemektedir. İleri düzey uygulamalar için de ARM Cortex geliştirme kartı tasarlanabilir.

IV. KAYNAKÇA

[1] Tangaraj S., Gummadi S. ve Radhakrishnan S, "Enhancement in ARM Code Optimization for Memory Constrained Embedded Systems", Advanced Computing and Communications, 483-486, 2006

[2] Cheng Y.Q., Zhu M ve W. Ge., "Signal Integrity Simulation Design of Image Processor PCB Combined with Electromagnetic Compatibility Analyses Based on Altium Designer 6", 4th Industrial Electronics and Applications Conference, 2009.

[3] Lijie Z., Li C. ve Shaozhong L., "Development of a tilt mesaurement system based on Mems sensor and Cortex-M3", 10th Electronic Measurement ve Instrument Conference, 2011.

[4] F. Hanzic ve safaric R., "ARM-Cortex Microcontroller fuzzy position control on an automatic door test-bed", 19th International Robotics in Alpe-Adria-Danube Region, 2010.

[5] Pal T., Shekhar C. ve Dutt Sharma H., "Design and Implementation of Embedded Speed Controller on ARM for Micromanufacturing Applications", Advanced in Computing, Control ve Telecommunication Technologies, 2009

[6] Popa M., Popa S. A., Slavici T ve Silaghe L., "On the Implementation of the OSEK/VDX Operating System on advanced Microcontrollers", International Conference Computers as an Tool, 2007

Uzaktan İzleme ve Kontrol Sistemi - ReMoniCS

Dr Çağrı Tanrıöver
Whizcomm Limited
c/o Albert J. Pope, Westfield Court
3rd Avenue, BA3 4XD, Bath, UK
eposta: cagri@whizcomm.com

Özetçe— **ReMoniCS, insan erişiminin zor, seyrek ve tehlikeli olduğu yerlerde bulunan malzeme depoları, baz istasyon şebekeleri ve eski binaların uzaktan takibi, bu yerlerdeki çevresel değişkenlerin ölçülmesi ve raporlanması ve acil durumlarda gereken uyarı mesajlarının merkeze iletilebilmesi amacıyla tasarlanmış bir sistemdir. Sistem, insan gücünün daha verimli olarak kullanımı, denetim maliyetlerinin önemli şekilde düşürülmesi ve maddi yatırımların güvenliği ve sistemin güvenilir şekilde çalışması hedeflenerek geliştirilmiştir. Bu makalede ReMoniCS'in teknik özellikleri, kullanım alanları ve getirdiği avantajlar anlatılmıştır.**

I. GIRIŞ

Günümüzdeki uygulamaların bir kısmında insan gücünün kullanımı ekonomik, güvenlik ve lojistik açıdan uygun olmayabilmektedir. Şehir dışında bulunan ve yerleşim merkezlerine kilometrelerce uzakta kurulmuş GSM anten ve baz istasyonları buna uygun bir örnektir. Benzer şekilde insan yerleşiminden uzakta bulunan tarihi binaların korunabilmesi için bu yerlerde sürekli olarak bir güvenlik görevlisinin bulundurulması maliyet ve lojistik olarak uygun olmayabilir. Çoğu zaman insan müdahalesi olmadan güvenilir biçimde çalışan bir şebekeye, teknik bir sorun çıktığında veya güvenliği tehdit eden koşullar oluştuğunda bir teknisyenin müdahalesi gerekmektedir. Büyük bir malzeme deposuna insan erişimi sorun olmasa da bu depodaki raflardaki yük miktarının gerçek zamanlı olarak ölçümü çalışanların güvenliği ve depolanan malzemelerin zarar görmemesi bakımından önem taşımaktadır.

Yukarıda verilen örnekler gerçekte karşılaşılan problemlerin sayıca küçük bir bölümü olmasına rağmen, çoğu zaman karşılaşılan izleme, denetleme ve kontrol ihtiyaçlarının büyük bir kısmını temsil eden uygun senaryoları kapsamaktadır. ReMoniCS de bu uygun senaryolar göz önüne alınarak geliştirilmiş ölçeklenebilir ve uygulamaya göre özelleştirilebilen bir platformdur.

II. SISTEM ÖZELLIKLERI

ReMoniCS'in yapıtaşlarından bahsetmeden önce genel senaryolar göz önüne alınarak, bir kullanıcının gerçekçi ihtiyaçları ve bu ihtiyaçlara cevap verebilecek çözüm kümesine kısaca bakmak uygun olacaktır.

A. İhtiyaçlar ve Çözümler

Öngörülen genel kullanıcı ihtiyaçları aşağıda özetlenmiştir.

- Kapalı bir alanda personelin bulunup bulunmadığı otomatik olarak tespit edilerek gerektiğinde sorgulanabilmeli.
- Kapalı alandaki bir lambanın açık olup olmadığı sorgulanabilmeli. Enerji sarfiyatını azaltabilmek için gerekmeyen koşullarda lamba kapalı olmalı.
- Ortam sıcaklığı ve ortamdaki toz miktarı ölçülebilmeli.
- Kapalı alandaki yangın ve hırsızlık gibi risklere karşı otomatik uyarı üretilebilmeli.
- Raflardaki fazla yüklemeye karşı uyarı mekanizması bulunmalı.
- Pil ile çalışacak donanım en fazla ayda bir kez bakıma ihtiyaç duymalı.
- Toplanan algılayıcı verilerinin iletimi, donanımın kontrolü ve sorgulanması özel altyapı kurulumu olmadan ve mesafeden bağımsız olarak gerçekleştirilebilmeli.

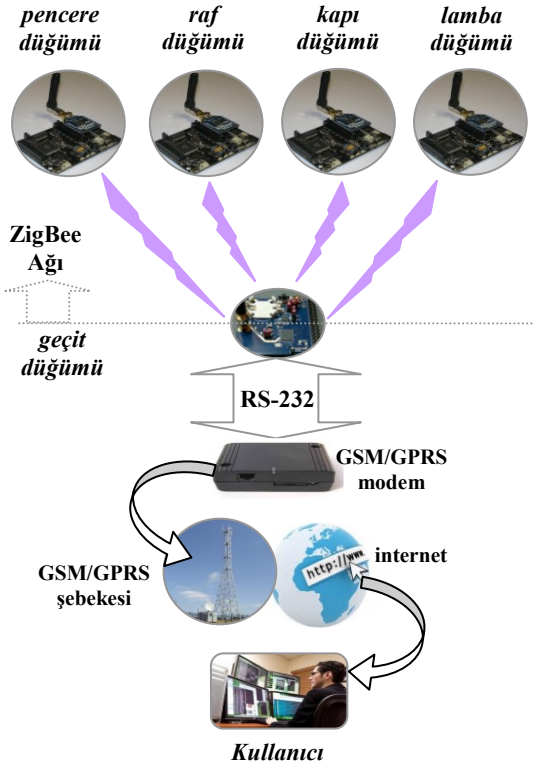
Yukarıda listelenen ihtiyaçlar doğrultusunda kullanıcıya sunulan çözüm genel çerçevesiyle aşağıdaki özellikleri içermektedir.

- Kapalı alandaki hareket algılayıcısı sayesinde personelin varlığı tespit edilebilir.
- Kapalı alandaki iki ışık algılayıcıdan birisi cam kenarında diğeri ise yapay ışık kaynağının yanında olacaktır. Enerji sarfiyatının azaltılması bu iki algılayıcıdan alınan bilgi sayesinde gerçekleştirilebilir. Ortamdaki yapay ışık kaynağı sistem tarafından otomatik olarak control edilebilir.
- Ortam sıcaklığı ^{0}C cinsinden ölçülecektir. Sıcaklık programlanmış seviyeyi aşınca sistem otomatik uyarı mesajı göndermektedir.
- Ortamdaki toz miktarı "parçacık algılayıcısı" kullanarak mg/m^3 cinsinden ölçülecektir. Sistem, toz ölçümü programlanan seviyenin üzerine çıktığında kullanıcıya otomatik uyarı mesajı gönderir.
- Karbondioksit (CO_2) algılayıcısından alınan veri sayesinde yangın riskine karşı kullanıcı uyarılabilir. Bu uyarının doğruluğu ortam sıcaklık bilgisi kullanılarak ve CO_2 algılayıcısı en az 10 dakika açık tutulduktan sonra ölçme yapılarak arttırılabilir.
- Kapı ve pencerelere takılan manyetik algılayıcılar ile odaya yapılacak giriş ve çıkışlar tespit edilebilir. Buna ek olarak cam kırılması ihtimaline karşı pencerelere yaklaşık 2 cm uzunluğundaki darbe algılayıcı da yerleştirilecektir.
- Raflardaki aşırı yükleme akselerometre yardımıyla üç eksendeki hareket değişimleri izlenerek ölçülebilir. Bu değişimler 0.05^0 hassasiyette yapılabilir. Önceden programlanan referans değişimlere göre gereken uyarı işareti otomatik olarak kullanıcıya iletilir.

- Pil ile çalışacak algılayıcı sistemlerinin uyku modu seçimi ve görev devri (duty cycle), minimum pil bakımı gerektirecek şekilde programlanabilir. Kullanıcının yüzde (%) cinsinden vereceği enerji sarfiyatı sistem tarafından uyku modu ve görev döngüsü parametrelerine çevrilmektedir.
- Sistem algılayıcılarla kablosuz iletişimi ZigBee [1] aracılığıyla sağlar. Geçit düğümü RS-232 seri kablosuyla GSM/GPRS/3G modemine [2] bağlanır.
- Sistemin yönetimi, programlanması ve izlenmesi kişisel bilgisayar ortamında çalışan özel bir uygulama ile gerçekleştirilir.

B. *Sistem Mimarisi*

1) *Genel Yapı:* ReMoniCS'in genel yapısı Şekil 1'de gösterilmiştir. Sistemin uç kısmında pencere, kapı, raf ve lamba olarak adlandırılmış dört tip algılayıcı düğümü bulunmaktadır. Bu düğümler ZigBee protokolünü kullanarak geçit düğümüyle haberleşmektedirler. Geçit düğümü RS-232 seri protokolünü desteklediği için birçok üretici firma tarafından sağlanan GSM/GPRS/3G modemleriyle kolaylıkla bağlaşabilir. Kullanıcı, kişisel bilgisayarına yüklenen uygulama sayesinde modem ve geçit düğümüyle iletişim kurarak, algılayıcı düğümlerden verileri dolaylı olarak okuyabilir veya bu düğümleri gerektiği şekilde programlayabilir.



Şekil 1: ReMoniCS Yapısı

Algılayıcı düğümler donanım olarak bir ana kart ve onun üzerine yerleştirilen üst kartdan oluşur. Üst kart birden fazla algılayıcıyı desteklediği için uygulamanın gerektirdiği şekilde yapılandırılabilir. Ana kart, mikrokontrolörü ve bir dizi giriş/çıkış portunu barındırır. Kullanılan işlemci ATMega1281 [3] şu temel özellikleri içerir.

- 8-bit tabanlı ve 16 MHz hız.
- 128 kB Flash bellek.
- 8 kB SRAM.
- 54 programlanabilir giriş/çıkış.

Ana kart üzerinde termometre ve akselerometre de bulunmaktadır. Geçit ve algılayıcı düğümlerde kullanılan ana kart aynıdır fakat geçit düğümünde algılayıcı üst kart bulunmamaktadır.

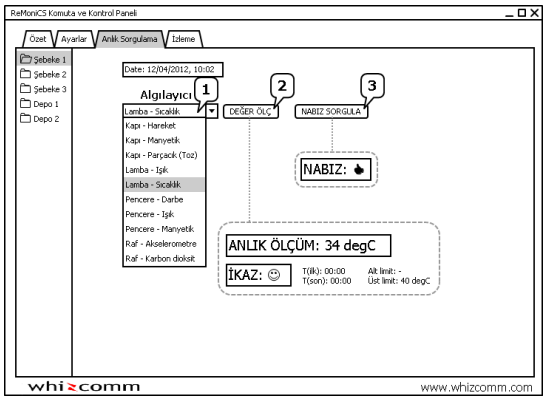
Sistemdeki algılayıcılar ve ilgili düğümler Tablo 1'de verilmiştir. Tüm düğümlerde akselerometre ve termometre bulunduğu halde, uygulamaya bağlı olarak bunların hepsi kullanılmayabilir. ReMoniCS'de raf düğümündeki akselerometre ve lamba düğümündeki termometre sorgulanabilmektedir.

Tablo 1: ReMoniCS Algılayıcıları.

Düğüm	Mevcut Algılayıcılar
Geçit	akselerometre, sıcaklık
Kapı	akselerometre, sıcaklık, hareket, manyetik, parçacık
Pencere	akselerometre, sıcaklık, ışık, manyetik, darbe
Lamba	akselerometre, sıcaklık, ışık
Raf	akselerometre, sıcaklık, karbon dioksit

2) *Arayüz ve Sistem Yönetimi:* ReMoniCS arayüzü depo, şebeke ve benzeri alanların yönetimini kolaylaştıracak şekilde tasarlanmıştır. Yönetim ekranlarından "Anlık Sorgulama" ve "İzleme" örnek bir senaryo için sırasıyla Şekil 2 ve Şekil 3'de gösterilmiştir.

Şekil 2'deki "Anlık Sorgulama" ekranının sol tarafında ReMoniCS donanımının bulunduğu yerler listelenmektedir. Listelenen yerlerin her birinin içeriği ekranın sağ tarafındaki alanda detaylı olarak izlenebilir. Bu, ReMoniCS'in tüm ekranları için geçerlidir ve kullanıcıya anlaşılabilir bir erişim sağlar. Ekranda yer alan günün tarihi ve saati alanının altında sistemde aktif olan alıcı düğümleri listelenir (Şekil 2'de '1' ile işaretlenmiştir). Kullanıcı bu listeden sorgulamak istediği her alıcıyı seçebilir. Şekil 2'de verilen örnekte kullanıcı, lamba düğümündeki sıcaklık alıcısını seçmiş görünmektedir. Seçilen bir algılayıcıya iki komuttan birisi gönderilebilir: "Değer Ölç" veya "Nabız Sorgula" (Şekil 2'de sırasıyla '2' ve '3' ile gösterilmiştir).



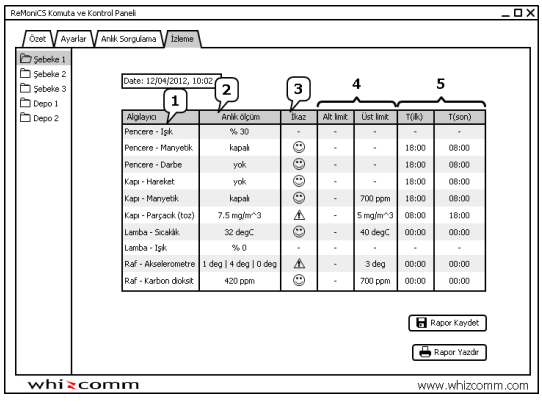
Şekil 2: Anlık Sorgulama Ekranı

Değer ölçme komutu geçit düğümüne ulaştıktan sonra ZigBee kablosuz ağına uygun protokole dönüştürülür ve ilgili algılayıcı düğümüne geçit tarafından yollanır. Bu noktadaki dönüştürme işlemi ne kadar kısa sürerse sistem kullanıcıya o kadar hızlı cevap verebilir. Bu nedenle RS-232 üzerinden iletilen paket yapısı ile ZigBee ağında kullanılan paket yapıları olabildiğince birbirine yakın tutulmuştur. Gereken ölçüm alındıktan sonra bilgiler geçit tarafından tekrar arayüze yönlendirilir. Kullanıcı eğer ölçülecek bilgi için bir ikaz bayrağı programlamışsa bu bayrağın anlık durumu da iletilir. İkaz bayraklarıyla ilgili detaylı bilgi ilerleyen bölümlerde sunulacaktır.

ReMoniCS arayüzündeki "Değer Ölç" butonu birden fazla bilginin ekrana gelmesini sağlar. Bu bilgilerden ilki algılayıcıdan alınan anlık ölçümdür. Kalan bilgiler kullanıcı tarafından tanımlanan ikaz durumu ile ilgilidir. İkaz bilgisi, algılanacak parametrenin alt ve üst sınır değerleri ile tanımlanmaktadır. Algılayıcıdan belli aralıklarla okunan değerler, tanımlanan aralığın dışına çıktığında ikaz bilgisi otomatik olarak kullanıcıya iletilir. Şekil 2'de sunulan ekranda termometre ikaz bilgisi için sadece üst limit olarak 40 degC tanımlanmıştır. İsteğe göre ikaz bilgisi gün içerisinde belli saat aralıklarında kullanıcıya gönderilebilir. Tanımlanan zaman aralığı dışındaki ikaz bilgisi algılayıcıda üretilse de kullanıcıya gönderilmez. Şekil 2'deki örnekte saat aralığı 00:00 – 00:00 olarak tanımlanmıştır ki bu da ikaz bilgisinin 24 saat boyunca kullanıcıya gönderilme isteğini belirtmektedir. İkaz durumu gerçekleşmemişse kullanıcı ekranında bu gülen yüz sembolüyle ifade edilir. Şekil 2'deki örnekte anlık sıcaklı ölçümü 34 degC olarak belirtilmiştir ve bu değer 40 degC üst limitinden düşük olduğu için ikaz alanında gülen yüz sembolu gösterilmiştir.

Arayüzdeki "Nabız Sorgula" butonu (Şekil 2, '3' numaralı alan) seçilmiş algılayıcının bağlı olduğu düğümün çalışır durumda olup olmadığını tespit etmek için kullanılır. Şekil 2'de verilen örnekte termometre'nin bağlı olduğu lamba düğümü'nün çalıştığı gösterilmiştir.

Şekil 3'deki "İzleme Ekranı" Şebeke 1'de mevcut olan tüm düğümler ve algılayıcıları hakkındaki bilgileri tek ekranda özetlemektedir.



Şekil 3: İzleme Ekranı

Şekil 3'de '1' ile işaretlenmiş sütunda Şebeke 1'de bulunan pencere, kapı, lamba ve raf düğümleri ve aktif olan algılayıcılar listelenmektedir.

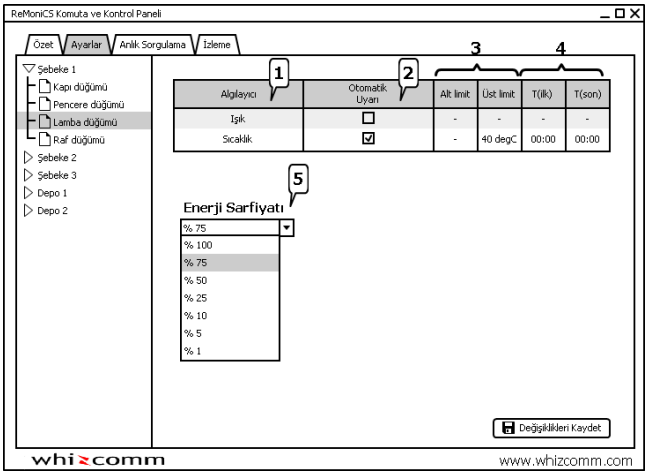
'2' ile gösterilen sütunda ise mevcut algılayıcılardan alınmış anlık ölçümler gösterilmektedir. Bu ölçümler kullanıcıdan bağımsız olarak periyodik olarak güncellenmekte ve ekrana yansıtılmaktadır. Algılayıcıların ikaz/uyarı durumu ise '3' numaralı sütunda bulunmaktadır. Bu sütunda olası üç durum görülebilir. İlk durum pencere-ışık ve lamba-ışık algılayıcılarındaki '-' sembolüyle gösterilmiştir. Bu sembol, ilgili algılayıcıda herhangi bir uyarının kullanıcı tarafından istenmediğini göstermektedir. İkinci sembol,'☺', ise algılayıcıda henüz bir uyarının olmadığını göstermektedir. Son durum sembolü ise '⚠' olup algılayıcıda ortaya çıkan ikazı gösterir.

"İzleme Ekranı"ndaki '4' ile işaretlenmiş sütunlar ikaz işaretine karşı düşen ölçüm değerlerinin üst ve alt sınırlarına ayrılmıştır. Bu sütunlarda üst ve/veya alt sınırlar programlanmamışsa, '-' sembolü kullanılır.

Son olarak '5' ile gösterilmiş sütunlar ise belirtilen ikaz işaretinin günün hangi saatleri arasında geçerli olduğunu belirtmektedir. İkaz işareti üretilmiyorsa bu aralıktaki sütunlarda '-' sembolü gösterilir (bakınız Pencere-Işık algılayıcı sırası). Eğer ikaz işareti sürekli olarak aktif olacaksa bu durum her iki sütunda da '00:00' ile belirtilir (bakınız Lamba-Sıcaklık algılayıcı sırası).

Şekil 3'deki "İzleme Ekranı"nda "Rapor Kaydet" butonu ekrandaki anlık görüntüyü kopyalayarak dosyaya yazılı olarak kaydetmeyi sağlar. Aynı ekrandaki görüntü doğrudan yazıcıdan çıktı olarak alınmak istenirse, "Rapor Yazdır" butonuna basmak yeterli olur.

ReMoniCS arayüzü mevcut lokasyonlardaki tüm düğümlerin konfigürasyonunun da yapılmasına imkan vermektedir. Bu özellik Şekil 4'de gösterilen "Ayarlar" ekranı aracılığıyla kullanılabilir. İlgili şekilde "Şebeke 1" adlı lokasyonda dört farklı düğümün bulunduğu görülmektedir. Bu düğümlerden "Lamba Düğümü" kullanıcı tarafından seçildiği için ekranın sağ tarafında bu düğümde yer alan tüm algılayıcılar listelenmiştir.



Şekil 4: Ayarlar Ekranı

Lamba düğümünde yer alan ışık ve sıcaklık algılayıcıları '1' ile işaretlenmiş sütunda görülmektedir. Mevcut algılayıcıların ikaz üretmesi de '2' ile gösterilen sütundaki "Otomatik Uyarı" alanından programlanabilir. Verilen örnekte ışık algılayıcı ikaz üretmezken sıcaklık algılayıcısı 40 derece üst sınırını aştığında ('3' ile işaretlenmiş sütun) uyarı bilgisi yollayacak şekilde programlanmıştır. Sıcaklık algılayıcısı bu ikazı 24 saat boyunca üretecek şekilde programlanmıştır ki bu da '4' ile gösterilen sütunlardaki T(ilk) ve T(son) alanlarının aynı olmasından anlaşılmaktadır.

ReMoniCS algılayıcı düğümleri pil ile çalıştıkları için bakım açısından işletime ek bir yük getirmektedirler. Dolayısıyla bu yükü en aza indirebilmek amacıyla her düğümde pil ömrünü uzatabilmek ve enerjiyi en verimli şekilde kullanabilmek için arayüzde "Enerji Sarfiyatı" ('5' numaralı alan) yüzde cinsinden programlanabilmektedir. Burada belirtilen yüzdelerin enerji sarfiyatı karşılıkları testlerle her algılayıcı için ayrı ayrı hesaplanarak düğümlerdeki yazılıma programlanmıştır. Bu nedenle düğümlere yüzde cinsinden programlanan bir sayı algılayıcıya ve onun çalışması için gereken pil sarfiyatına bağlı olarak, görev devrine çevrildikten sonra kullanılmaktadır.

Şekil 4'deki arayüzde düğümde programlanan algılayıcıların yeni ayarlara göre çalışabilmesi için kullanıcının "Değişiklikleri Kaydet" butonuna basması yeterlidir.

III. SONUÇLAR

Bu makalede Uzaktan İzleme ve Kontrol Sistemi – ReMoniCS'in çalışma prensibi ve kullanıcısıyla olan etkileşimi sunulmuştur. ReMoniCS insan erişiminin zor olduğu ve denetimlerin yeterli sıklıkta gerçekleştirilemediği yerlerde kontrol, gözetim ve kumanda imkanı veren bir sistemdir. Kullanıcıya getirdiği temel avantaj, yüksek maliyetli envanterin bulunduğu noktalarda çıkan sorunlardan anında haberdar olabilmek ve bu sorunları en kısa zamanda çözümlemenin yanısıra denetim maliyetlerini düşürerek insan gücünün daha verimli şekilde kullanılmasını sağlamaktır.

ZigBee geçit düğümünde ortaya çıkabilecek bir arıza, ReMoniCS sistemine erişimi engelleyebilir. Bunun temel nedeni koordinatör düğümü olarak programlanan geçitin merkezle iletişimi sağlayan tek nokta olmasıdır. Sistemdeki bu zayıf halkanın ek bir koordinatör ve GSM/GPRS modem ilavesiyle güçlendirilmesi kritik görev icra edilen kullanım senaryolarında güvenilirliği arttıracak bir önlemdir. Bu ilave elbette system maliyetini yükseltecektir fakat söz konusu görevin kritiklik derecesine bağlı olarak bu ek maliyet yükü önemsiz olabilir. Yukarıda bahsedilen zayıf noktada gerçekleşebilecek bir hata ReMoniCS sistemi içerisinde "Nabız Sorgulama" özelliği sayesinde hızlı bir şekilde tespit edilebilir. Böylece koordinatör düğümünde ortaya çıkabilecek sorunlar nedeniyle sistemin uzun süreli olarak devre dışı kalması önlenebilmektedir.

Geçit düğümü dışındaki algılayıcı düğümlerden birisinde ortaya çıkabilecek hatalar ise sadece hatalı düğümdeki işlevlerin icra edilememesine neden olur. ReMoniCS'in çalışır durumdaki düğümlerinin tamamı normal işleyişini sürdürebilir. Yukarıda bahsedildiği gibi hatalı düğümlerdeki sorun da nabız sorgulamasıyla hızlı biçimde tespit edilerek giderilebilmektedir.

Burada anlatılan sistemdeki donanımın benzerleri piyasada bulunmaktadır [4], [5]. ReMoniCS donanım bakımından ek bir değer veya özgünlük getirmemektedir. ReMoniCS'in kattığı değer yazılımın ve mimarinin ölçeklenmeye uygun şekilde tasarlanmış olması ve müşteri ihtiyaçlarına göre değiştirilebilmesidir. ReMoniCS'in yapısal esnekliği kullanıcıya maliyet bakımından avantajlar sunmaktadır.

Burada belli bir kullanıcı kesimi göz önüne alınarak geliştirilen bir çözüm özetlenmiştir. Fakat ReMoniCS'in oldukça ölçeklenebilen ve başka problemlere de çözüm sunabilecek genel bir platform olduğu da unutulmamalıdır. Makinalar arası haberleşme (M2M Communications [6], [7]) mobil haberleşme ve internet teknolojilerinin ucuzlayarak daha erişilebilir hale gelmesi ve tabana yayılması sonucu hızla büyüme potansiyelini yakalamış olan bir sektördür. Bu nedenle ReMoniCS ve benzeri çözümlerin sayısı ve kapsamı önümüzdeki yıllarda daha artarak günlük yaşantımızın bir parçası olacaktır.

IV. KAYNAKÇA

[1] ZigBee Alliance, "ZigBee Specification," www.zigbee.org, documentation 053474r17, January 2008.

[2] Cinterion, "Cinterion TC65 Terminal," www.cinterion.com, TC65T Datasheet, 2012.

[3] Atmel Corporation, "8-bit Atmel Microcontroller with 64K/128K/256K Bytes In-System Programmable Flash," www.atmel.com, Mayıs 2012.

[4] BlueRadios, "BR-BG-02 Cellular Gateway, Data Concentrator and Content Server", www.blueradios.com/BR-BG-02.pdf.

[5] Digi, "XBee ZB RF Modules", www.digi.com.

[6] Whizcomm Limited, "M2M Solutions and M2M Applications," www.whizcomm.com/m2m-solutions-and-m2m-applications/, 2012.

[7] M2M daily, http://www.m2mdaily.com/daily-news/m2m-communications.

Düşük Güçlü Kablosuz Algılayıcı Ağı ile Aydınlatma Kontrol Sistemi

Berk Baykal[1], Ali Temel Hacıhamzaoğlu[1], Sermin Kılıvan[2], Oğuzhan Urhan[1,3], Sarp Ertürk[1,3]

[1]Kocaeli Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü, Umuttepe, 41380, İzmit - Kocaeli
[2]VİKO Elektrik ve Elektronik Endüstrisi San.ve Tic. A.Ş.
Abdurrahmangazi Mah. Ebubekir Cad. No:44, 34887 Sancaktepe - İstanbul
[3]Pars Ar-Ge Bilgi Tekn. Elkt. Müh. ve Dan. Hiz. San. ve Tic. Ltd. Şti.
Kocaeli Üniversitesi Teknoparkı, Kocaeli
e-posta: berk@berkbaykal.com, alitemel61@gmail.com, skilivan@viko.com.tr , {urhano,sertur}@kocaeli.edu.tr

Özetçe—**Bu çalışmada kablosuz algılayıcı ağı şeklinde yapılandırılmış bir aydınlatma kontrol sistemi geliştirilmiştir. Bu sistemin temel bileşenleri anahtar ve röle ünitesidir. Anahtarlar ile kontrol ünitesi arasındaki iletişim normal şartlarda kablolu olarak gerçekleştirilmektedir. Bu çalışma kapsamında ise kontrol ünitesi kablosuz ağın koordinatörü (co-ordinator), anahtarlar ise son cihaz (end-device) olarak ele alınıp IEEE 802.15.4 standardı alt yapısı ile bir kablosuz ağ oluşturulmuştur. Çalışmada kullanılan haberleşme modülleri düşük güç tüketimi ile ön plana çıkmaktadır. Yapılan deneyler geliştirilen kablosuz modüllerin 500mAh'lik bir pil ile 5 yıldan fazla çalışmasının mümkün olduğunu göstermiştir.**

I. GİRİŞ

Otomasyon sistemlerinin üretim tesislerinde kullanımı verimlilik açısından artık rutin bir hal almıştır. Bu sistemlerin ev/bina otomasyonunda kullanımı her geçen gün artmaktadır. Ev otomasyon sistemlerinin temel olarak üç ana bileşeni bulunmaktadır. Bunlar aydınlatma, termostat ve jaluzi kontrolüdür. Geçmişte yaygın olarak kablolu sistemlerle kontrol edilmekte olan bu bileşenlerin kablosuz olarak kontrol edilebilmesi yakın geçmişte bir ihtiyaç haline dönüşmeye başlamıştır [1].

Viko Elektrik-Elektronik A.Ş. binaların iç tesisatlarında, alçak gerilimle çalışan anahtar, priz, kumanda ve besleme amaçlı ürünler, elektrik sayaçları ve bina otomasyon sistemlerinde otomatik kontrol amacı için kullanılan pek çok elektronik ürün üretmektedir. Viko'nun iş geliştirme ve pazarlama birimleri yeni iş alanı olarak, akıllı bina konseptini yerli üretim için firma gündemine taşımıştır. Ülkemizde yeni gelişmekte olan bina otomasyonu alanının, ithal ürünlerin hâkimiyetinde olduğu öteden beri bilinen ve firmanın yakından ilgilendiği bir iş alanıdır. Pazarın bu ihtiyacı sonucunda eksik kalan kablosuz anahtar kontrol sistemi için Kocaeli Üniversitesi iş birliği ile bu çalışmada detayları sunulan bir proje gerçekleştirilmiştir.

Bu çalışmada geliştirilen anahtar birimleri ile aydınlatma birimlerinin kablosuz olarak kontrolü hedeflenmektedir. Pazarda bu noktada, yeni projeler ve renovasyon projeleri olmak üzere iki farklı ihtiyaç vardır [2]. İlk olarak; yeni projelerde, anahtar ile aydınlatma arasına kablo çekilmesi istenmemektedir. Özellikle açık ofis tarzı uygulamalarda sonradan yapılan bölmelerin aydınlatma kontrolünde kablo çekilmesi mümkün olmamaktadır. İkinci olarak; renovasyon ve restorasyona tabii tutulacak yapılarda ise aydınlatmanın kontrolü mekanik anahtarla yapılmakta ve bu tesisata müdahale edilmeden mekanik anahtarın uzaktan kontrol edilmesi talep edilmektedir.

II. GELİŞTİRİLEN SİSTEM

Aydınlatma kontrolü için kablolu bir anahtarlama sistemi kullanıldığında, aydınlatma birimi doğrudan ilgili aydınlatma anahtarı tarafından kontrol edilmektedir. Kablosuz olup röle tipi kontrol ünitesi kullanmayan bir aydınlatma kontrol sisteminde ise her odadaki aydınlatma elemanın yakınına bir alıcı yerleştirilmesi gerekmektedir. Bu tip bir sistemde röle tipi bir kontrol ünitesi kullanıldığında ise aydınlatma kontrol anahtarları kablolarla röle tipi kontrol ünitesine bağlanmakta ve aydınlatma kontrolü röle tipi kontrol ünitesi tarafından yapılmaktadır.

Bu çalışmada aydınlatma kontrolü için kullanılması planlanan sistemin fiziksel yerleşimi Şekil-1'de gösterilmektedir. Bu şekilde aydınlatma kontrolü yapılacak her ortamda bir kablosuz anahtar modülü (KM) bulunmaktadır. Bu anahtar modülleri kablosuz ağ üzerinden kontrol ünitesine (KÜ) bağlanmaktadır. Aydınlatma elemanın açılması veya kapatılması istendiğinde KM'den bu bilgi KÜ'ye iletilmektedir. KÜ ise kablosuz ağ üzerinden kendisine ulaşan bilgiyi değerlendirerek ilgili aydınlatma elemanını açıp/kapatmaktadır.

Şekil-1: Kontrol ünitesi ve Kablosuz Modüllerin Yerleşimine Tipik Bir Örnek

(a) Tekli anahtar

(b) İkili anahtar

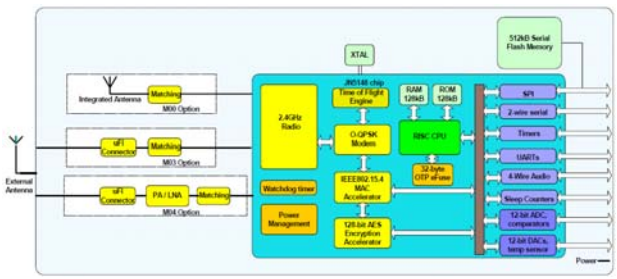
(c) DIN4 kutu [13 klemens] (d) DIN6 kutu [26 klemens]

Şekil-2: Farklı Özellikteki Anahtar ve Pano Tipi Kontrol Üniteleri

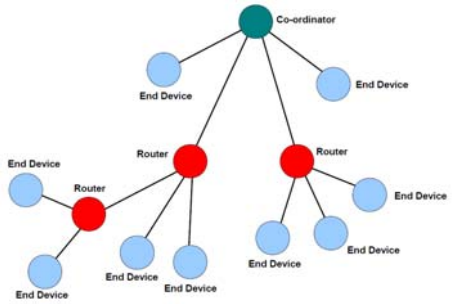
Şekil-2'de farklı konfigürasyondaki tekli (1 buton x 2 fonksiyon) ve ikili (2 buton x 4 fonksiyon) anahtar tipleri ve örnek iki pano tipi kontrol ünitesi görülmektedir.

Bu çalışma kapsamında geliştirilecek kablosuz ağın en önemli özelliği düşük güç tüketimi olarak belirlenmiştir. Kablosuz ağ alt yapısı için Wi-Fi [3], Zigbee [4], 6LoWPA [5] gibi alternatifler değerlendirilse de yapılan detaylı incelemeler sonucunda Philps-NXP firmasının dahili kablosuz haberleşme modülü içeren bir mikrodenetleyicisinin (JN5148 wireless microcontroller) kullanılmasına karar verilmiştir. SoC (Single on Chip) yapıdaki bu mikrodenetleyicinin blok yapısı Şekil-3'de gösterilmektedir. Bu SoC modül 32-bitlik RISC yapıdaki mikrodenetleyici ve tümleşik bir RF birimi içermektedir. Böylelikle hem fiziksel olarak küçük boyuta sahip hem de daha düşük güç tüketimine sahip olmaktadır. Bu SoC modül veri gönderirken kaynaktan 15mA, veri alırken 17.5mA çekmektedir.

IEEE 802.15.4 standardında yayın yapabilen bu modülün üzerinde ZigBee PRO protokolünün yanı sıra JenNet, JenNet-IP adı verilen daha düşük güç tüketimine sahip bir protokollerin kullanılması da mümkündür. Aydınlatma kontrol sisteminin sağlaması gereken özellikler ve güç tüketimi dikkate alınarak bu protokollerinde JenNet'in kullanılması karar verilmiştir. En fazla 500 uç noktanın aynı ağda çalışmasını destekleyen bu protokol ağaç (tree), yıldız (star) ve doğru (linear) topolojilerini kullanımına olanak sağlamaktadır.



Şekil-3: Kullanılan Modülün İç Yapısı



Şekil-4: Örnek Bir Ağaç Tipi Ağ Topolojisi

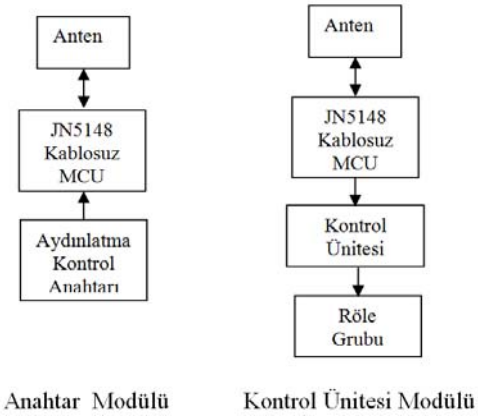
Bu çalışmada aydınlatma kontrol sistemi için Şekil-4'de detayları gösterilen ağaç topolojisinin kullanılması karar verilmiştir. Bu topolojide ağı başlatan ve koordine eden bir "Co-ordinator" bulunmaktadır. Ağdaki diğer birimler ise "Router" ve "End-Device"lardır. Bu çalışmada "Co-ordinator" birimi KÜ'nin içine konumlandırılmıştır. Anahtar üniteleri ise "End-Device"lara karşılık gelmektedir. Güç tüketiminin önemli olmadığı durumlarda "Router" kullanılmadan "End-Device"ların doğrudan "Co-ordinator"e bağlanabilmesi mümkündür. Ancak uzakta bulunan bir "End-Device"ın, "Co-ordinator"e bağlanması için yüksek güç harcaması gerekebilir. Bu batarya ile çalıştırılması gereken "End-Device"lar için önemli bir dezavantaj olabilmektedir. Bu çalışma ağaç topolojisinin kullanılmasına temel nedenleriden birisi de budur.

Ağaç topolojisinde "End-Device"lar doğrudan "Co-ordinator"e bağlı olabileceği gibi bir "Router" üzerinden de "Co-ordinator"e bağlanabilir. Böylelikle "Co-ordinator"e göre uzak bir yere konumlu "End-Device" yüksek güçle yayın yapmadan istediği bilgiyi "Router" aracılığı ile " Co-ordinator "e iletebilmektedir.

Şekil-5'de bu çalışmada kullanılan anahtar modüllerinin ve kontrol ünitesi modüllerinin blok yapıları gösterilmiştir. Kablosuz anahtar modülü, butonlar, kablosuz JN5148 mikrodenetleyicisi ve dahili anten biriminden oluşmaktadır. Kontrol ünitesi ise harici anten, JN5148 mikrodenetleyicisi ve röle gurubu ile bu röleleri kontrol eden ikinci bir mikrodenetleyiciden oluşmaktadır. KM'deki mikrodenetleyici zamanının çoğunu uyku modunda düşük güç tüketerek geçirmektedir. Herhangi bir butona basıldığında aktif moda geçerek basılan butonun kodunu "Co-ordinator" pozisyonundaki KÜ'ye kablosuz olarak iletmektedir. KÜ'ye iletilen bu bilgi JN5148 tarafından ilgili aydınlatmanın kontrol edilebilmesi için ikincil mikrodenetleyiciye iletilmektedir.

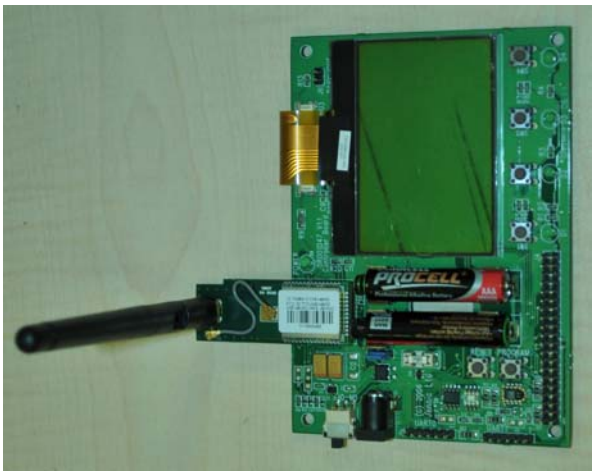
Alıcı verici modüller arasındaki haberleşmeye istenmeyen müdahaleleri engellemek amacıyla AES (Advanced Encryption Standard) 128-bit şifreleme yaklaşımı kullanılmaktadır.

Şekil-6'da bu çalışmada kablosuz aydınlatma kontrol sistemi için geliştirilen anahtar modülünün iki PCB'den oluşan sandviç yapıdaki devresi ve montajı gösterilmiştir. Devredeki ilk PCB dörtlü anahtarlara basıldığını algılamaya yarayan buton devresi ile bağlantı halindedir. Bu butonlara basılıp basılmadığı ikinci PCB'de bulunan JN5148 mikrodenetleyicisi ile algılanmaktadır. İkinci PCB'de hem JN5148 mikrodenetleyicisi hem de batarya bulunmaktadır.



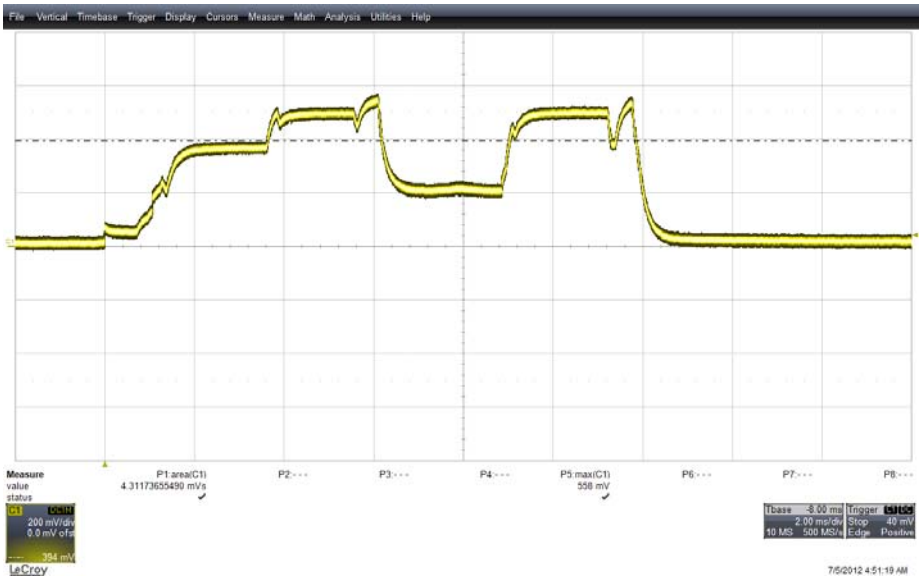
Şekil-5: Aydınlatma Kontrol Sistemde Kullanılacak Anahtar ve Kontrol Ünitesi Modülleri.

SoC modül RAM'deki bilgileri tutarak uyku modunda kullanıldığında 3.45µA harcamaktadır. Sonraki aşamada butondan veri okunacaktır. Butona basma süresince kaynaktan pull-up direncine bağlı olarak bir akım çekilecektir. Üçüncü aşamada CSMA/CA (Carrier sense multiple access with collision avoidance) yaklaşımı ile boşta uygun RF kanalına erişim hakkı sağlanacaktır. Bu işlem 1.088ms'de 17.5mA harcanarak gerçekleştirilmektedir. Son aşamada bir çerçevelik veri 2.58ms'de iletilip bu esnada kaynaktan 15mA çekilmektedir. Sistemin toplam güç tüketimini ölçmek için bataryaya seri bir direnç bağlanmış ve osiloskop ile bu direnç üzerideki gerilim ölçerek direnç değerine bölündüğünde tek bir veri aktarımı için kaynaktan ortalama 173µAs'lik bir akım geçtiği ölçülmüştür. Saatte ortalama 5 kez basma işlemi gerçekleştirildiği varsayılırsa sadece veri basma anlarında kaynaktan çekilen toplam akım 865µAs=0.24µAh olacaktır. Uyku modundaki akımda dikkate alındığında geliştirilen sistemin saatte ortalama 5 işlemin yapıldığı durumda kaynaktan ortalama 4µAh'den az bir güç gereksinimi bulunmaktadır. Bu durumda 250mAh'lik bir batarya ile teorik olarak 7 yıldan uzun bir süre kullanım mümkün olacaktır.

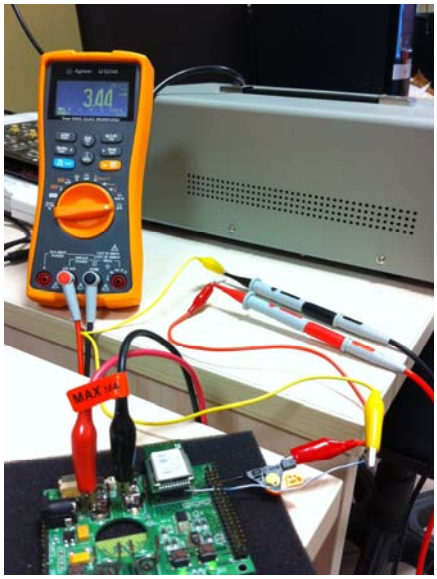


Şekil-7: Kontrol Ünitesindeki Kablosuz Alıcı Modül

Şekil-6: Kablosuz Anahtar Modülü ve bu modülün anahtar içine yerleşimi



Şekil-9: Veri aktarımı durumunda bataryadan kaynaktan çekilen akım



Şekil-8: Uyku modunda kaynaktan çekilen akım

Geliştirilen sistem ile yapılan ölçümlerde sistemin RAM'deki bilgiyi tutarak uyku moduna sokulması durumunda kaynaktan 3.44µA'lik akım çekildiği görülmüştür. Bu ölçüm Agilent'ın U1273A model numaralı multimetresi ile alınmıştır. Bu ölçüm Şekil-8'de gösterilmektedir. Şekil-9'da kablosuz anahtarın herhangi bir butonuna basıldığında kaynaktan çekilen akım görülmektedir. Görüldüğü gibi uyku modunda kaynaktan çekilen akım oldukça sınırlıdır. Butona basılması sonrasında JN5148 uyku modundan uyanmakta ve kesmeyi işlemeye başlamaktadır. Veriyi göndermeden önce yayın yapmak istediği kanalda halihazırda bir yayın olup olmadığını dinlemektedir. Bu kanalda herhangi bir yayın olmadığına karar verdikten sonra yayın yaparak veriyi iletmektedir. Bu işlem sonrasında tekrar uyku moduna geçerek butona basılmasını beklemeye başlamaktadır.

Geliştirilen sistemde mikrodenetleyiciler RAM'deki bilgi aktif tutularak uyku moduna sokulduğundan sistemin uyku modundan kalkıp gerekli bilgili KM'ye iletmesi birkaç milisaniye içerisinde geçekleşmektedir.

III. SONUÇLAR

Bu çalışmada kablosuz algılayıcı ağı yaklaşımı kullanılarak bir aydınlatma kontrol sistemi geliştirilmiştir. Geliştirilen sistem temel olarak SoC yapıdaki bir kablosuz mikrodenetleyiciler tarafından kontrol edilmektedir. Anahtar ve kontrol ünitesi modülleri arasındaki kablosuz iletişim JenNet adı verilen, IEEE.802.14.5 standardı üzerinde çalışan ve herhangi bir telif hakkı ödemesi gerektirmeyen bir protokole ile sağlanmaktadır. Batarya ile çalışan anahtar modülünün güç tüketimini olabildiğince düşük tutabilmek için bu modül sürekli olarak uyku modunda tutulmakta sadece butona basıldığında veri aktarımı için aktif moda geçmektedir. Yapılan deneyler geliştirilen sistemin 250mAh'lik bir batarya ile 7 seneden daha uzun süre çalışabileceğini göstermiştir.

KAYNAKÇA

[1] Reinisch, W. Kastner, G. Neugschwandtner, and W. Granzer, "Wireless Technologies in Home and Building Automation," in 5th IEEE International Conference on Industrial Informatics, Vienna, 2007, pp. 93 - 98.

[2] A.J.D. Rathnayaka, V.M. Potdar, S.J. Kuruppu, "Wireless Technologies in Home and Building Automation," 5th IEEE International Conference on Digital Ecosystems and Technologies, Daejeon, Korea, 2011,pp. 76 – 81.

[3] R.J.Smith,"Wi-Fi Home Networking," McGraw-Hill Companies, 2003.

[4] K. Gill, S.-H. Yang, F. Yao, and X. Lu, "A ZigBee-Based Home Automation System," IEEE Transactions on Consumer Electronics, vol. 55, pp. 422-430, May 2009.

[5] G. Mulligan, "The 6LoWPAN architecture," in Proceedings of the 4th workshop on Embedded networked sensors CorkIreland, 2007, pp. 78-82

Panoramik Kamera Sistemi: PAN-KAM

Ahmet Tekyıldız[1], Çağrı Güvenel[1], Ramazan Duvar[1], Anıl Çelebi[1,2], Oğuzhan Urhan[1,2], Kemal Güllü[1,2], Sarp Ertürk[1,2]

[1]Kocaeli Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü, Umuttepe, Kocaeli
[2]PARS AR-GE Bilgi Teknolojileri Elektronik Mühendislik ve Danışmanlık Hizmetleri Sanayi ve Ticaret Ltd. Şti.
Kocaeli Üniversitesi Teknoparkı, Kocaelie-posta: {ahmet.tekyildiz, cagriguvenel35, ramazanduvar} @gmail.com , {anilcelebi, urhano, kemalg, sertur}@kocaeli.edu.tr

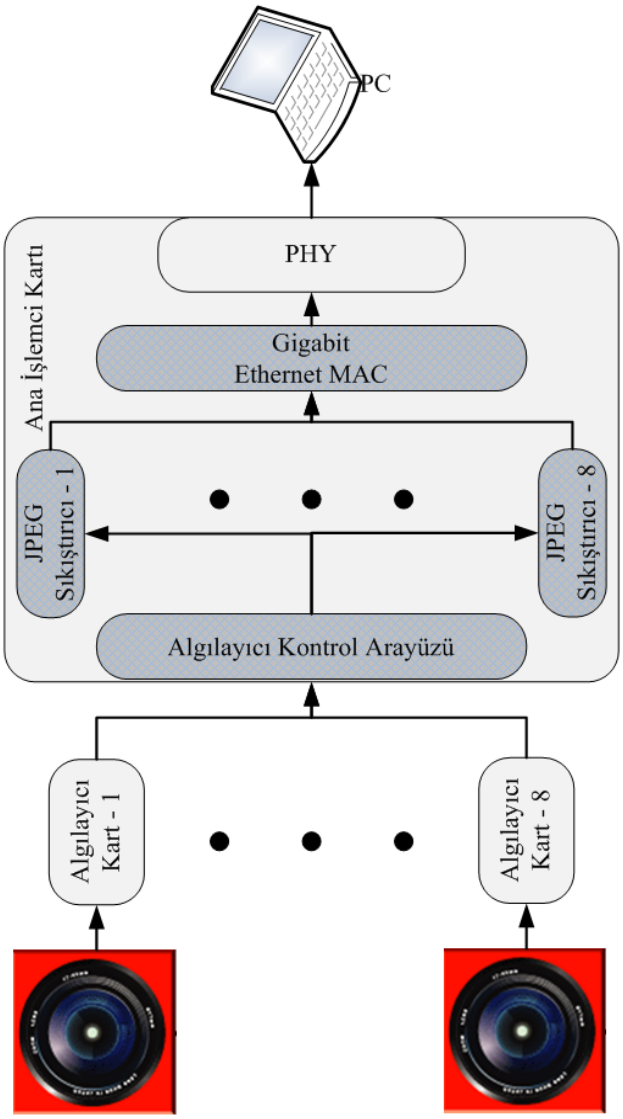
Özetçe—**Bu çalışmada Kocaeli Üniversitesi İşaret ve Görüntü İşleme Laboratuarında (KULIS) geliştirilen panoramik kamera sistemi sunulmaktadır. Geliştirilen sistem 8 kameradan alınan görüntüyü gömülü olarak sıkıştırıp Gigabit Ethernet (GigE) üzerinden göndermektedir. PC üzerinde koşan uygulama ile de GigE hattından alınan sıkıştırılmış imgeler birleştirilerek panoramik görüntü oluşturulmaktadır. Tasarlanan sistem gerçek zamanlı olarak saniyede 15 çerçeveyi sıkıştırıp gönderebilmektedir.**

I. GİRİŞ

Haberleşme standartlarının geçen zaman içinde hızlanması, sayısal veri sıkıştırma teknolojilerindeki gelişme ve paralel işlem kapasitesi yüksek yongaların ortaya çıkması ile birlikte yüksek miktarda sayısal verinin bir noktadan diğer bir noktaya aktarımı pratik olarak mümkün olmaya başlamıştır. Gerçek zamanlı panoramik görüntüleme teknolojileri günlük hatta sıkça rastlanmaktadır. Buna en tipik örnek Google'ın "Street View" uygulamasıdır [1]. Panoramik görüntüleme sistemleri günümüzde özellikle güvenlik, şehir ve bölge planlaması, video konferans uygulamalarında tercih edilmektedir. [2]'de konuşmacı konum kestirimi özellikli bir panoramik telekonferans sistemi 3 kamera ile yatayda yaklaşık 180 derecelik bir görüş alanı sağlayacak şekilde gerçekleştirilmiştir. [3]'de alan programlanabilir kapı dizisi (FPGA-Field Programmable Gate Array) temelli gerçek zamanlı çalışabilecek bir panoramik görüntü oluşturucu tasarlanmıştır. [4]'de yine [3]'e kıyasla daha başarılı bir panoramik görüntü oluşturucu tasarımı yapılmıştır. Bu tür bir sistemin gerçek zamanlı olarak çalışabilmesi için yüksek boyutlardaki imge verisini başarılı bir biçimde işleyebilmesine bağlıdır. Bu çalışmada 8 adet 2Mpiksel çözünürlüklü görüntü algılayıcıdan gelen imge verisini sıkıştırıp GigE ortamından gönderebilen bir sistem FPGA üzerinde 15 çerçeve/saniye hızında çalışabilecek şekilde gerçekleştirilmiştir. PC üzerinde de bu görüntülerde lens bozulmalarını ortadan kaldırıp panoramik görüntüleri oluşturan bir uygulama geliştirilmiştir. Geliştirilen sistemde tüketici seviyesinde bir alan programlanabilir kapı dizisi kullanılmıştır dolayısıyla geliştirilen sistemin olası bir ürünleştirme çalışmasında maliyeti çok fazla olmayacaktır.

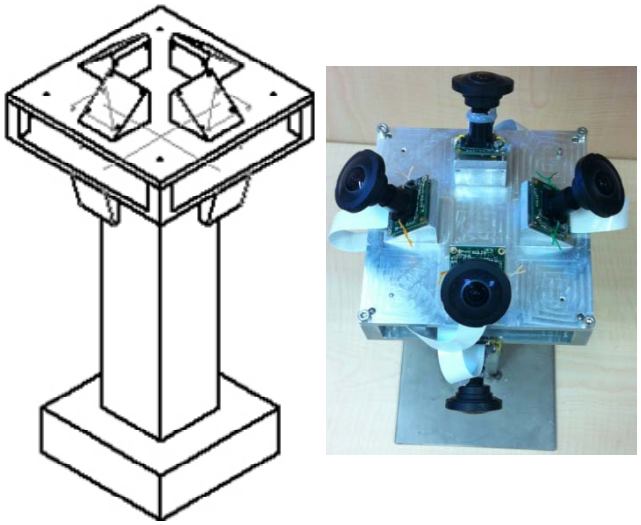
II. PANAROMİK KAMERA DONANIMI

Geliştirilen sistem 8 algılayıcı kart, 1 ana işlemci kartı, her bir algılayıcı kart için optik düzenek ve bütün algılayıcı kartların optik olarak uyumlu olmasını ve ana işlemci kartını bütünleştirecek bir mekanik yapıdan oluşmaktadır. Algılayıcı kartlar 2Mpiksellik görüntü algılayıcı elemanları ve gerekli elektronik devreyi barındırmaktadır. Ana işlemci kartı bir adet FPGA ve haberleşme ara yüz devrelerinden oluşmaktadır. Geliştirdiğimiz panoramik kameranın fonksiyonel şeması Şekil 1'de görülmektedir. Koyu gri şekilde gösterilen bloklar FPGA'nın içerisinde geriye kalan gri bloklar ise baskılı devre kartlar üzerinde bulunmaktadır.



Şekil 1. Geliştirilen panoramik kameranın blok şeması

Geliştirilen panoramik kameranın mekanik yapısı ve son halinin fotoğrafı Şekil 2'de görülmektedir. Bu yapı 8 adet algılayıcı kartın üzerlerinde bulunacak lenslerin optik merkezleri aynı noktada olacak şekilde tasarlanmıştır. Şekil 3'te de sistemin tamamının fotoğrafı görülmektedir.



(a) (b)

Şekil 2. Geliştirilen panaromik kameranın a) mekanik yapısı, b) üretim sonrası sistemin üstten görünüşü

A. Algılayıcı Kartlar

Algılayıcı kartlarda 2Mpiksel çözünürlüğe sahip standart bir algılayıcı kullanılmıştır. Bu kartlar algılayıcılar ile FPGA arasındaki kontrol ve veri haberleşmesini sağlayacak temel elektronik bileşenleri içermektedirler. Veri hızına bağlı olarak kart üzerinde çeşitli noktalarda daha özenli bir tasarım yapılmıştır.

B. Ana İşlemci Kartı

Bu kart kameranın işlem merkezidir. Bu kart üzerindeki elektronik 8 adet algılayıcı kart ile gerçekleştirilecek veri aktarımı, kontrol sinyalleşmesi ve PC ile GigE bağlantısını sağlayacak elektronik çevreseller ve güç devrelerinden oluşmaktadır.

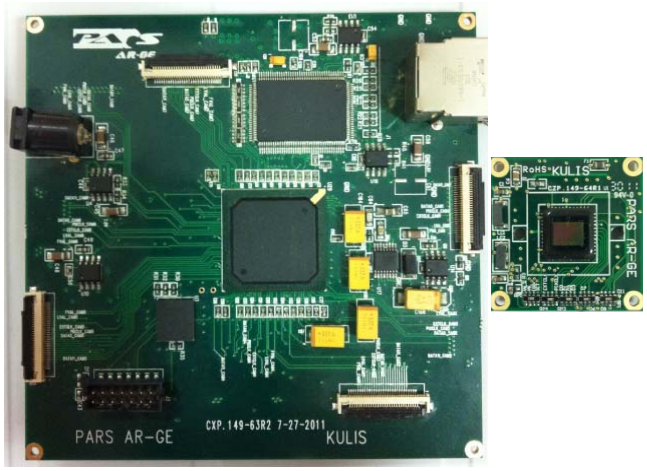
Bu kart üzerinde bulunan FPGA algılayıcıların düzgün çalışması, bu kartlardan gelen görüntü verisinin JPEG algoritması ile sıkıştırılması, GigE haberleşmesi için her bir kanal verisinin hizalanması ve GigE haberleşmesinin gerçekleştirilmesinden sorumludur. Şekil 3'de geliştirilen ana işlemci kartının ve bir adet algılayıcı kartının fotoğrafı görülmektedir.

C. PC Uygulaması

PC uygulaması GigE hattından gelen paketlerin açılması, JPEG dosyalarının oluşturulması ve geliştirilen birleştirme yöntemi ile 8 kanaldan 360 derecelik panaromik imgenin oluşturulmasından sorumludur.

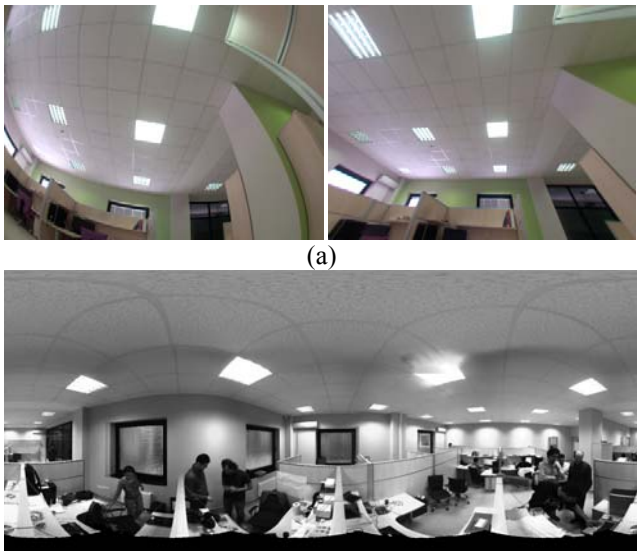
I. SONUÇLAR

Geliştirilen sistem ile 8 kanaldan gelen 2Mpiksel boyutundaki bilgi JPEG algoritması ile sıkıştırılıp GigE hattından 15 çerçeve/saniye hızında başarılı bir şekilde PC'ye aktarılmıştır. PC'de çalışan uygulama da bu imgeleri başarılı bir biçimde birleştirmiştir. Şekil 4'te geliştirilen sistem ile birleştirilen imgeler görülmektedir.



(a) (b)

Şekil 3. Geliştirilen panaromik kameranın a) Ana işlemci kartı, b) Algılayıcı kartı



(a)

(b)

Şekil 4. a) Geliştirilen PC uygulaması ile bir kameradan alınan imge ve lens bozulması giderilmiş hali, b) PC uygulaması tarafından oluşturulmuş panoramik görüntü

II. KAYNAKÇA

[1] www.google.com/streetview.

[2] T. Uesugi, T. Kawamura, T. Shimizu, K. Sugahara, "Hardware realization of panoramic camera with direction of speaker Estimation and a panoramic image generation function," Proc. of the 7th WSEAS Int. Conf. on Sim. Mod. and Opt., pp. 268-273, Beijing, China, September 2007.

[3] L. Chen, M. Zhang, B. Wang, Z. Xiong, G. Cheng, "Real-time FPBA-based panoramic unrolling of high-resolution catadioptric omnidirectional images," Proc. of Int. Conf. on Measuring Technol. and Mechatronics Automation, vol.1, pp. 502-505, Zhangjiajie, Hunan, China, April 2009

[4] L.-D. Chen, M.-J. Zhang, Z.-H. Xiong, "Series-parallel pipeline architecture for high-resolution catadioptric panoramic unwrapping," Image Processing, IET , vol.4, no.5, pp.403-412, October 2010

Sayısal Analog Dönüştürücülerde Kullanılan Ara Değerleme ve Modülasyon Sistemi Doğrulaması

Gürer Özbek, Ömer Kerem Karaali ve Türker Küyel
İstanbul Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Maslak, 34469, İstanbul
e-posta: {ozbekgu, karaalio, tkuyel}@itu.edu.tr

Özetçe—**Bu çalışmada, yüksek hızlı sayısal analog dönüştürücülerde (Digital to Analog Converter, DAC) kullanılan bir sayısal ara değerleme (enterpolasyon) ve modülasyon sisteminin test süreci anlatılmaktadır. Çalışma kapsamında, sistemin tanıtılması, test durumlarının oluşturulması, testlerin yapılması ve sonuçların yorumlanması yer almaktadır.**

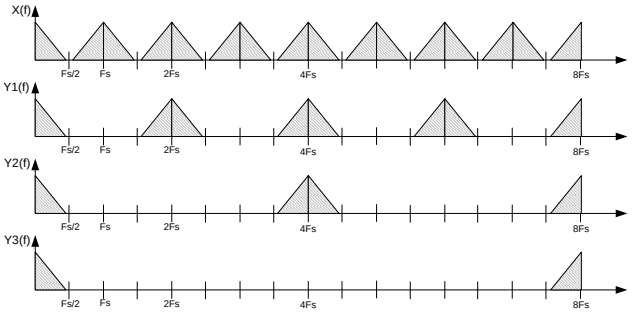
I. GİRİŞ

Teknolojinin gelişmesine paralel olarak yüksek hızlı veri transferi ve işaret işleme gereksinimlerinin artması, yüksek hızlı sayısal analog dönüştürücülere olan ilgiyi de gün geçtikçe arttırmaktadır [1], [2]. Baz istasyonlarından sayısal televizyon yayını sistemlerine kadar pek çok kullanım alanı bulunan DAC'lar, tümleştirme gereksinimlerinden dolayı sayısal işaret işleme işlevselliğine de sahip kırmıklar halinde tasarlanmaktadırlar [1], [2]. Gerekli olan sayısal işaret işleme özellikleri; ara değerleme (enterpolasyon), sayısal modülasyon, I/Q modülasyonu olarak sıralanabilir [1]-[4]. Bu tür kırmıkların ülkemizde tasarlanması da önem taşımaktadır. Sayısal analog dönüştürücü kırmıklarında bulunan sayısal işaret işleme birimlerinin tasarımı kadar test edilmesi de önem kazanmıştır. Uluslararası Yarı-iletken Teknoloji Yol Haritası (ITRS), kırmık üzeri sistemlerdeki (system on chip, SOC) sayısal işlevsellik test tekniklerinin önemine açıkça değinmiştir [5]. Bu nedenle tasarımı devam eden yüksek hızlı DAC kırmığında yer alan ara değerleme ve modülasyon sisteminin benzetim ve ölçüm sonuçlarının ilişkisi, bu çalışmanın konusunu oluşturmuştur.

II. SİSTEM TANITIMI

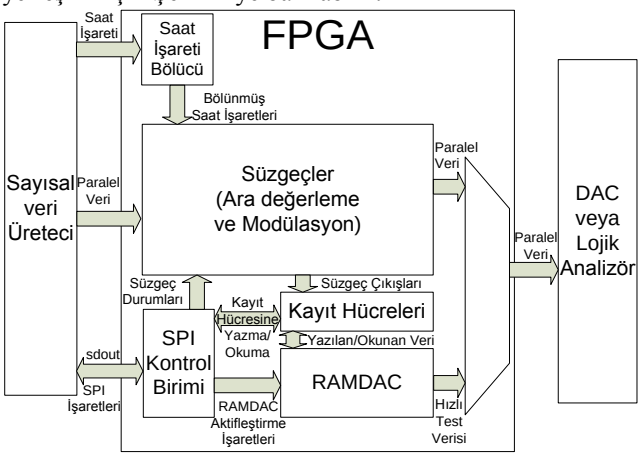
Sayısal analog dönüştürücülerde işaret bandının resimlerinin örnekleme frekansının katlarında ortaya çıkması sorunu, DAC çıkışına analog süzgeç eklenerek çözülür [3]. Resim frekanslarının yüksek bir başarımla bastırılmasının istendiği, ya da resim frekanslarının işaret bandına yakın olduğu uygulamalarda ise; kullanılması gereken analog süzgecin derecesi çok yüksek olmakta ve böyle bir süzgecin tasarımı pratik olmamaktadır. Ara değerleme yardımıyla giriş işaretinin örnekleme frekansını arttırmak, analog süzgecin derecesinin daha düşük yapılabilmesini sağlamakta ve analog süzgeci gerçeklemeye imkân vermektedir [3], [4].

Yüksek hızlı DAC'ların kullanıldığı sistemlere bakıldığında, çoğunlukla dönüştürücünün arkasına bir analog modülatör bağlanarak girişteki işaretin istenilen bir frekans aralığına ötelendiği görülmektedir. Analog modülatörün hatalarından dolayı, orta frekans bandında (Intermediate Frequency, IF) sayısal modülatörler tercih edilmektedir. Tasarımımızda sayısal ara değerleme ve modülasyon birimleri bulunmaktadır. Bu birimlerin durumları değiştirilerek girişteki işaret, frekans bandında istenilen yere taşınabilmektedir. Modülasyonsuz 8x ara değerleme için frekans spektrumu Şekil 1'de verilmiştir.



Şekil 1: 8x Ara değerleme işlevi.

Kırmık üstü sistemimizde yer alan bir diğer birim ise, kırmık iç devrelerine veri yazılıp okunmasını sağlayacak olan ve çok sayıda kayıt hücresine sahip sayısal kontrol birimidir. Bu birim, seri çevresel arayüz (Serial Peripheral Interface, SPI) üzerinden mikroişlemci ile haberleşir. Kontrol birimine ek olarak, sayısal analog dönüştürücüyü giriş padlerinin destekleyebileceğinden daha yüksek hızlı veri ile deneyebilmek için, kırmık içine RAMDAC isimli bir sayısal bellek döngüsü yerleştirilmiştir. Bu birime SPI ile düşük hızda veri yazılmakta ve veri yazımı bittikten sonra, hafızaya yazılmış olan veriler yüksek hız ile döndürülerek DAC devresini sürebilmektedir. Sistemin alt birimlerinin yerleşimi için Şekil 2'ye bakılabilir.



Şekil 2: Test edilen sistem ve alt birimleri.

III. TEST DURUMLARI

Kırmık tasarımına başlamadan önce Verilog diliyle yapılan tasarım, alanda programlanabilir kapı dizisi (Field Programmable Gate Array, FPGA) ile gerçeklenmiştir. Gerçeklenen tasarımda test edilmesi gereken fonksiyonellikler aşağıdaki ana başlıklar altında toplanmıştır.

- Kontrol birimindeki adreslere veri yazılması ve okunması
- Seçilen bir sigortanın yakılması için veri yazılması
- RAMDAC'ın doldurulması ve okunması
- Sayısal süzgeçlerin çalışma durumlarının seçilmesi
- 16-bit paralel verinin süzgeçlere uygulanması ve süzgeç/modülatör çıkışlarının okunması

A. Kontrol Birimindeki Adreslere Veri Yazılması ve Okunması

Seçilen bir adrese veri yazılması ve sonra da bunun okunması, SPI kullanılarak yapılacak en temel işlevdir. Bu nedenle de testler arasında ilk sırada yer almıştır. Bu test, sırasıyla bir yazma komutu ve bir okuma komutunun SPI'dan gönderilmesi ile yapılır. Yazılan veri ile okunan veri aynı ise sistemin doğru çalıştığı görülmüş olur.

B. Seçilen Bir Sigortanın Yakılması İçin Veri Yazılması

Kırmık içerisinde bir kez programlanabilir (One Time Programmable, OTP) kayıt hücreleri bulunduğunda bunların programlanması ve programlanan OTP nin değerinin geri okunması işlemi de SPI tarafından gerçekleştirilmektedir. FPGA ortamında sigortayı modellemek için "latch" elemanı kullanılmıştır.

C. RAMDAC'ın Doldurulması ve Okunması

128 kelime derinlikte ve 16 bit genişlikte olan RAMDAC kayıt hücrelerine yazma ve kontrol amaçlı okuma işlemleri, SPI ile yapılmaktadır. Yazılmak istenen veri, RAMDAC'ın örnekleme yapacağı kayıt hücresine yazılır ve RAMDAC'ın bu veriyi örneklemesi sağlanır. Okuma işlemi ise RAMDAC çıkışını örnekleyen kayıt hücrelerinin SPI'dan okunması ile sağlanır. Sonuç olarak okunan veri yazılan veri ile karşılaştırılarak doğrulama yapılır.

D. Sayısal Süzgeçlerin Çalışma Durumlarının Seçilmesi

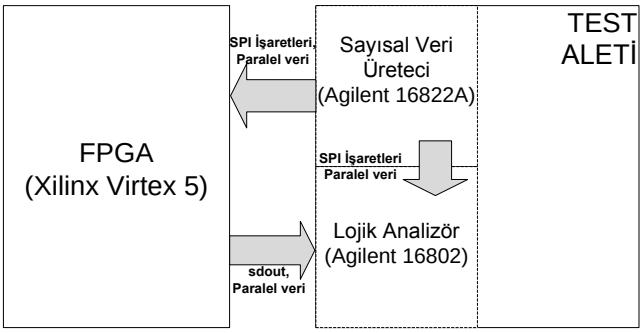
Kırmıkta, 41 değişik ara değerleme ve modülasyon durumu bulunmaktadır. Bu durumlar kullanıcı tarafından SPI ile seçilmektedir. Bu durumlara örnek olarak, 2X, 4X, 8X üst örnekleme ve ara değerleme durumları, Hilbert dönüştürücüleri ve işareti Fs/16 aralıklar ile değişik frekanslara öteleme durumları sayılabilir. Durumlara, RAMDAC'ın devreye sokulması gibi işlevler de eklenir. Bütün bu durumların seçimi, önceden belirlenmiş bir kayıt hücresine uygun bitlerin yazılması ile yapılır Bu test grubundaki amaç, uygun değerlerin SPI ile yazılıp, sistemin istenen duruma getirildiğinin doğrulanmasıdır.

E. 16-bit Paralel Verinin Süzgeçlere Uygulanması ve Süzgeç/Modülatör Çıkışlarının Okunması

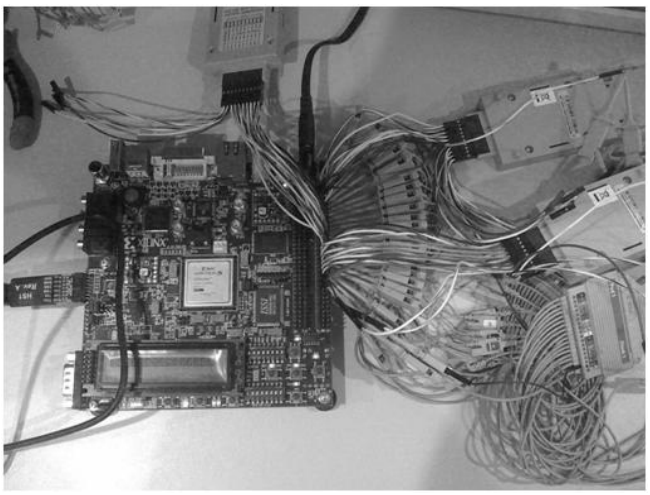
16-bit paralel veri dizisinin sistem girişine uygulanmasıyla süzgeç işlevselliği 41 işaret işleme durumu için test edilmiştir. Sistemde ara değerleme ve modülasyon işlemlerini gerçekleştiren seri bağlı üç adet süzgeç bulunmaktadır. Bu süzgeçlerin doğru çalıştığının tespit edilebilmesi için her bir süzgeç çıkışına istendiğinde SPI'dan okunabilen kayıt hücreleri yerleştirilmiştir. Herhangi bir anda süzgeç çalışması durdurulup SPI'dan okuma yapıldığında, çıkışlar gözlenebilmektedir. Bu verilerin davranışsal testler ile karşılaştırılması ile doğrulama yapılır.

IV. TESTLERİN YAPILMASI

Doğrulamalar iki aşamada gerçekleştirilmiştir. İlk aşamada, benzetim ile yapılan doğrulamalar Cadence'ın NCSim Verilog simülatörü kullanılarak gerçekleştirilmiştir. Bu testlerde her bir iç kayıt hücresinin değerleri gözlenmiş ve istenen özellikleri sağlayıp sağlamadığı kontrol edilmiştir. Bu testler, donanım testleri için de referans kabul edilmiştir. İkinci aşamada ise donanım testleri yapılmış, Xilinx Virtex 5 FPGA içine gömülü olan sayısal devre, Agilent 16822A sayısal veri üreteci ile sürülmüş ve Agilent 16802A logic analyzer test aleti ile ölçülmüştür. Kurulan test düzeni Şekil 3 ve 4'te verilmiştir.



Şekil 3: Test düzeninin blok diyagramı.

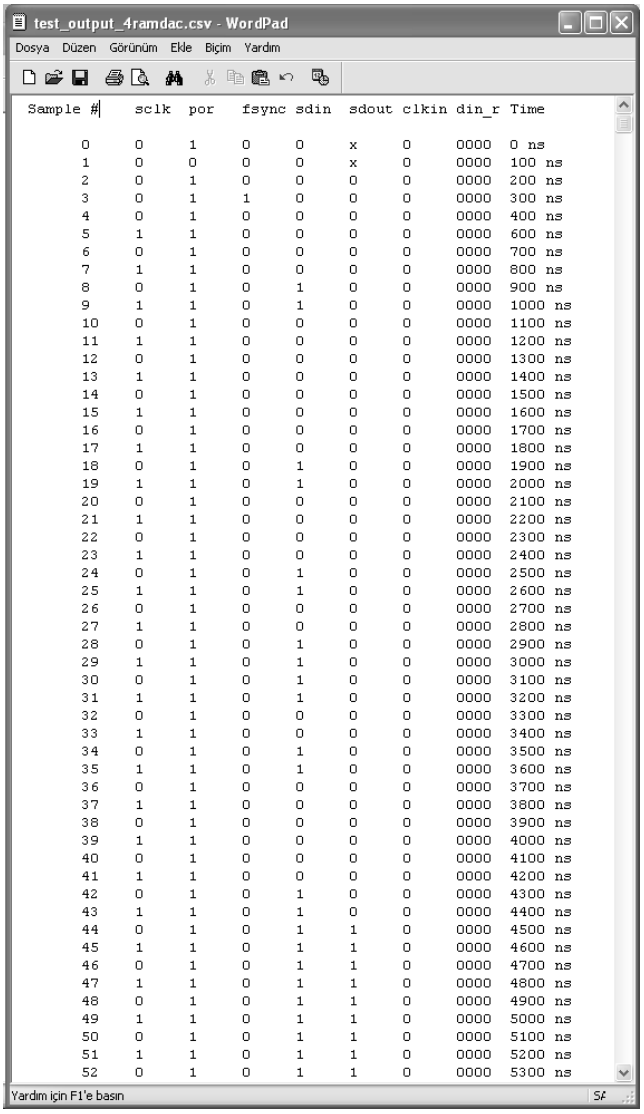


Şekil 4: Test düzeninin fotoğrafı.

FPGA testlerinde davranışsal benzetimler referans alındığı için, FPGA girişlerine uygulanacak işaretlerin

davranışsal benzetimlerde uygulananlarla birebir aynı olması gereklidir. Bunun hızlı ve güvenilir şekilde oluşturulabilmesi için, benzetimlerde kullanılan Verilog yazılımı, uygulanan işaretlerin bir kopyasını, 16822A veri üretecinin anlayacağı formatta bir dosyaya yazmaktadır.

Davranışsal benzetim sonuçları ile FPGA testlerinin sonuçlarının hızlı ve doğru biçimde karşılaştırılması gereklidir. Bunun sağlanması için, C diliyle bir karşılaştırma programı yazılmıştır. Bu programa giriş olarak davranışsal testler ile FPGA testlerinin çıkış işaretlerinin yazılı olduğu iki dosya verilir (Şekil 5 ve 6). C programı iki dosyadaki çıkışların aynı olup olmadığını kontrol eder ve bir hata olması durumunda hangi anlarda farklı çıkışların geldiğini gösterir. Programın her iki duruma karşı verdiği çıkışlar, Şekil 7 ve 8'de verilmiştir.

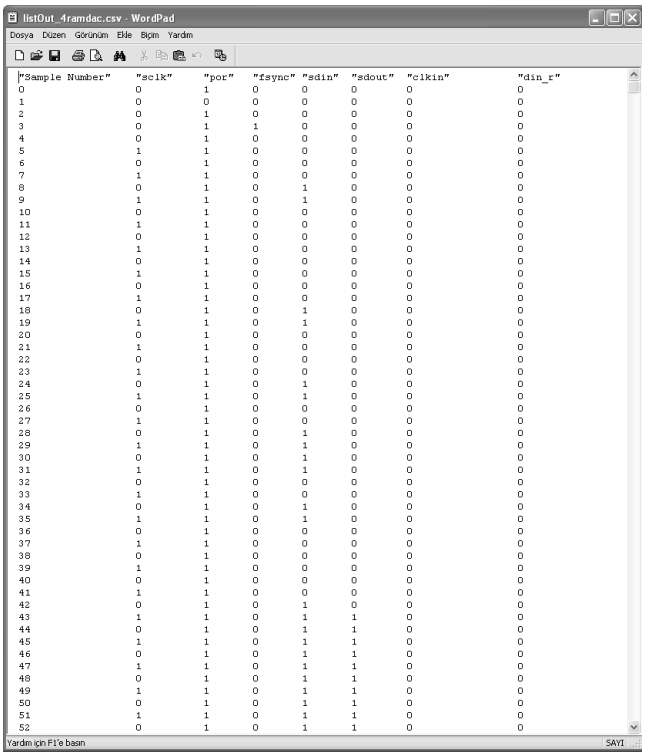

test_output_4ramdac.csv - WordPad

Dosya Düzen Görünüm Ekle Biçim Yardım

Sample #	sclk	por	fsync	sdin	sdout	clkin	din_r	Time
0	0	1	0	0	x	0	0000	0 ns
1	0	0	0	0	x	0	0000	100 ns
2	0	1	0	0	0	0	0000	200 ns
3	0	1	1	0	0	0	0000	300 ns
4	0	1	0	0	0	0	0000	400 ns
5	1	1	0	0	0	0	0000	600 ns
6	0	1	0	0	0	0	0000	700 ns
7	1	1	0	0	0	0	0000	800 ns
8	0	1	0	1	0	0	0000	900 ns
9	1	1	0	1	0	0	0000	1000 ns
10	0	1	0	0	0	0	0000	1100 ns
11	1	1	0	0	0	0	0000	1200 ns
12	0	1	0	0	0	0	0000	1300 ns
13	1	1	0	0	0	0	0000	1400 ns
14	0	1	0	0	0	0	0000	1500 ns
15	1	1	0	0	0	0	0000	1600 ns
16	0	1	0	0	0	0	0000	1700 ns
17	1	1	0	0	0	0	0000	1800 ns
18	0	1	0	1	0	0	0000	1900 ns
19	1	1	0	1	0	0	0000	2000 ns
20	0	1	0	0	0	0	0000	2100 ns
21	1	1	0	0	0	0	0000	2200 ns
22	0	1	0	0	0	0	0000	2300 ns
23	1	1	0	0	0	0	0000	2400 ns
24	0	1	0	1	0	0	0000	2500 ns
25	1	1	0	1	0	0	0000	2600 ns
26	0	1	0	0	0	0	0000	2700 ns
27	1	1	0	0	0	0	0000	2800 ns
28	0	1	0	1	0	0	0000	2900 ns
29	1	1	0	1	0	0	0000	3000 ns
30	0	1	0	1	0	0	0000	3100 ns
31	1	1	0	1	0	0	0000	3200 ns
32	0	1	0	0	0	0	0000	3300 ns
33	1	1	0	0	0	0	0000	3400 ns
34	0	1	0	1	0	0	0000	3500 ns
35	1	1	0	1	0	0	0000	3600 ns
36	0	1	0	0	0	0	0000	3700 ns
37	1	1	0	0	0	0	0000	3800 ns
38	0	1	0	0	0	0	0000	3900 ns
39	1	1	0	0	0	0	0000	4000 ns
40	0	1	0	0	0	0	0000	4100 ns
41	1	1	0	0	0	0	0000	4200 ns
42	0	1	0	1	0	0	0000	4300 ns
43	1	1	0	1	0	0	0000	4400 ns
44	0	1	0	1	1	0	0000	4500 ns
45	1	1	0	1	1	0	0000	4600 ns
46	0	1	0	1	1	0	0000	4700 ns
47	1	1	0	1	1	0	0000	4800 ns
48	0	1	0	1	1	0	0000	4900 ns
49	1	1	0	1	1	0	0000	5000 ns
50	0	1	0	1	1	0	0000	5100 ns
51	1	1	0	1	1	0	0000	5200 ns
52	0	1	0	1	1	0	0000	5300 ns

Yardım için F1'e basın

Şekil 5: Karşılaştırılan dosya 1: benzetim çıktısı.

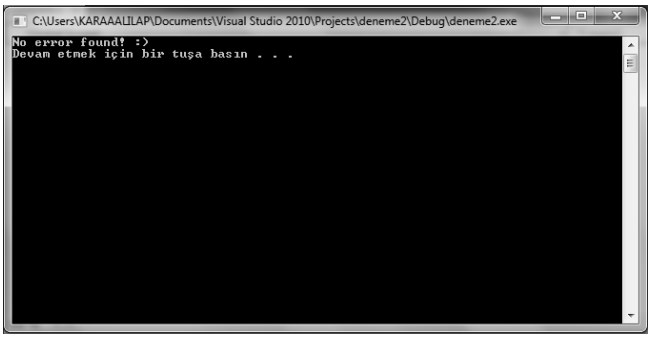

listOut_4ramdac.csv - WordPad

Dosya Düzen Görünüm Ekle Biçim Yardım

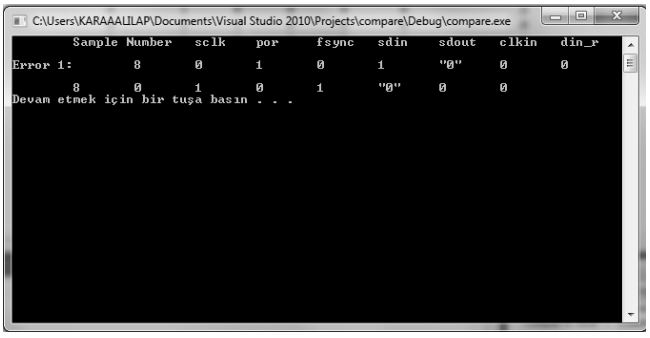
"Sample Number"	"sclk"	"por"	"fsync"	"sdin"	"sdout"	"clkin"	"din_r"
0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0
3	0	1	1	0	0	0	0
4	0	1	0	0	0	0	0
5	1	1	0	0	0	0	0
6	0	1	0	0	0	0	0
7	1	1	0	0	0	0	0
8	0	1	0	1	0	0	0
9	1	1	0	1	0	0	0
10	0	1	0	0	0	0	0
11	1	1	0	0	0	0	0
12	0	1	0	0	0	0	0
13	1	1	0	0	0	0	0
14	0	1	0	0	0	0	0
15	1	1	0	0	0	0	0
16	0	1	0	0	0	0	0
17	1	1	0	0	0	0	0
18	0	1	0	1	0	0	0
19	1	1	0	1	0	0	0
20	0	1	0	0	0	0	0
21	1	1	0	0	0	0	0
22	0	1	0	0	0	0	0
23	1	1	0	0	0	0	0
24	0	1	0	1	0	0	0
25	1	1	0	1	0	0	0
26	0	1	0	0	0	0	0
27	1	1	0	0	0	0	0
28	0	1	0	1	0	0	0
29	1	1	0	1	0	0	0
30	0	1	0	1	0	0	0
31	1	1	0	1	0	0	0
32	0	1	0	0	0	0	0
33	1	1	0	0	0	0	0
34	0	1	0	1	0	0	0
35	1	1	0	1	0	0	0
36	0	1	0	0	0	0	0
37	1	1	0	0	0	0	0
38	0	1	0	0	0	0	0
39	1	1	0	0	0	0	0
40	0	1	0	0	0	0	0
41	1	1	0	0	0	0	0
42	0	1	0	1	0	0	0
43	1	1	0	1	1	0	0
44	0	1	0	1	1	0	0
45	1	1	0	1	1	0	0
46	0	1	0	1	1	0	0
47	1	1	0	1	1	0	0
48	0	1	0	1	1	0	0
49	1	1	0	1	1	0	0
50	0	1	0	1	1	0	0
51	1	1	0	1	1	0	0
52	0	1	0	1	1	0	0

Yardım için F1'e basın SAYI

Şekil 6: Karşılaştırılan dosya 2: donanım testi çıktısı.



Şekil 7: Dosyalardaki çıkışların aynı olması durumu için program çıktısı.



Şekil 8: Dosyalardaki çıkışların farklı olması durumu için program çıktısı.

V. SONUÇLAR

FPGA testlerinden elde edilen sonuçlara bakıldığında, davranışsal testlerle tamamen aynı sonuçların alındığı ve bit hata oranının sıfır olduğu görülmüştür. Tasarımın donanımda doğru bir şekilde çalıştığı gözlenmiştir. Verilog, FPGA ve lojik analizör kullanarak, işlevsel test otomasyonu sağlanmıştır.

VI. KAYNAKÇA

[1] Analog Devices, "Dual, 12-/14-/16-Bit, 1 GSPS Digital-to-Analog Converters", AD9779A kataloğu, alındığı tarih: 01.08.2011, adres: http://www.analog.com/static/imported-files/data_sheets/AD9776_9778_9779.pdf

[2] Texas Instruments, "16-Bit, 1.0 GSPS 2x-4x Interpolating Dual-Channel Digital-to-Analog Converter (DAC)", DAC5682z kataloğu, alındığı tarih: 06.04.2012, adres: http://www.ti.com/lit/ds/symlink/dac5682z.pdf

[3] A.V.Oppenheim ve R.W.Schafer, "Discrete-Time Signal Processing", Prentice-Hall, 1998.

[4] P.P.Vaidyanathan, "Multirate Systems and Filter Banks", Prentice-Hall, 1992.

[5] ITRS, "Test and Test Equipment", 2009 Edition, alındığı tarih: 11.10.2012, adres: http://www.itrs.net/Links/2009ITRS/2009Chapters_2009Tables/2009_Test.pdf

GÖMÜLÜ SİSTEM İLE YEREL METEOROLOJİ İSTASYONU GELİŞTİRİLMESİ

Ahmet ALBAYRAK
Sinop Üniversitesi
Ayancık Meslek Yüksekokulu
Ayancık, 57400 Sinop
e-posta: aalbayrak@sinop.edu.tr

Eslem Erva Yılmaz
Sinop Üniversitesi
Ayancık Meslek Yüksekokulu
Ayancık, 57400 Sinop
eposta: eervay@gmail.com

Gamze Gündoğdu
Sinop Üniversitesi
Ayancık Meslek Yüksekokulu
Ayancık, 57400 Sinop
eposta: ggundogd@gmail.com

Özetçe— **Bu çalışmada, uzman sistem yardımıyla yerel olarak hava tahmini yapan gömülü sistem tasarlanmıştır. Uzman sistemin tahmin yapması için gerekli olan sıcaklık, nem, rüzgar hızı ve yönü ve basınç parametreleri ölçülmüştür. Ölçülen bu parametreler ile günde 3 defa hava tahmini yapılmaktadır. Gerçek zamanlı olarak yapılan ölçümler sunucu üzerinden web sayfasında anlık olarak gösterilmektedir.**

I. GIRIŞ

Hava sürekli, dinamik, kaotik ve çok fazla parametre içeren lineer olmayan bir süreçtir. Yakın zaman periyodunda bile tahmin yaparken çok miktarda karmaşık veriye gereksinim duyulmaktadır. Bu özelliklerinden dolayı hava durumu tahmini yapmak oldukça zor bir iştir.

Hava durumu tahmini ile aşırı yağışların neden olduğu su taşkını ve doğal afetlerin önüne geçilebilir. Hava tahmininde bulutların sahip olduğu sıcaklıklar çok önemli yer tutar. Yağmurun şiddetini ve süresini bu sıcaklık farkından elde edebiliriz.

Uzman sistem bir yapay zeka yazılımı olup lineer problemlerin çözümünde rahatlıkla kullanılabilir. Hava tahmini lineer olmayan ve çok karmaşık bir süreç olduğundan dolayı uzman sistemin ne kadar başarı sağlayacağı bilinmemektedir. Yapılan deneyler sonucunda uzman sistemin tahminleri meteoroloji müdürlüğünün verileri ile karşılaştırılmıştır.

Gömülü sistemler özel amaçlar için tasarlanmış, içerisinde uygulamaya özel işlemci ve çevre birimleri bulunan bir tasarımdır. Gömülü sistemler kendisi için önceden özel olarak tanımlanmış görevleri yerine getirir.

Bu çalışma şu basamakları içermektedir. Giriş bölümünden sonra, günümüzde hava durumu tahmininin nasıl yapıldığı açıklanmaktadır. Üçüncü bölümde hava durumu tahmininde kullanılan uzman sistemden bahsedilmiştir. Dördüncü bölümde ise kullanılan materyaller ve gömülü sistemi içeren geliştirme kartı hakkında bilgi verilmiştir. Beşinci bölüm yapılan deneylerle ilgili olup altıncı bölümde sonuç ve öneriler bulunmaktadır.

II. HAVA DURUMU TAHMİNİ

Hava durumu tahmini, belirli bir bölge veya merkezde belirli bir zaman dilimi içerisinde olası meteorolojik olayların gözlem ve analizlere dayanılarak subjektif veya objektif yöntemler kullanılarak önceden öngörülme çalışmaları olarak adlandırılabilir. Hava tahmini 3 aşamalı bir süreçtir. Bu sürecin ilk aşaması verilerin toplanması için yapılan gözlem deneylerini kapsar. Gözlem deneyleri, yer gözlemleri, yüksek hava gözlemleri ve uydu gözlemleri olarak bilinir.

Yer gözlemleri sinoptik olarak yapılır. İngiltere'deki Grinwich'ten geçen boylam derecesi başlangıç kabul edilir ve bu başlangıç boylamında 12.00 GMT'de (Grinwich Mean Time) yapılan bir sinoptik rasat mahalli olarak Türkiye'de öğleden sonra 15.00'de, Hindistan'da akşam 18.00'de, Avustralya'da gece 22.00'de ve Orta Amerika'da ise sabah 05.00'de yapılır. Bu gözlemlerin hepsi de 12.00 GMT gözlemi olarak isimlendirilir. Bu gözlem de çok sayıda parametre ölçülebilir. Hava basıncı, rüzgar yönü-hızı, hava sıcaklığı, nem oranları, bulutluluk oranı, günlük buharlaşma oranı vs. ölçülen parametreler arasında sayılabilir.

Yüksek hava gözlemleri atmosferin üst tabakaları için gözlem yapan istasyonlarda radyo vericili gözlem aleti, hidrojen veya benzeri hafiflikte gazla doldurulmuş bir balona bağlanarak atmosfere bırakılır. Bu balonlarla 30-40 km yüksekliğe kadar çıkabilen ölçüm cihazı; Belirli basınç seviyelerinin yüksekliğini, Bu seviyelerdeki sıcaklık ve nemi, rüzgar yön ve şiddetini ölçerek radyo sinyalleri ile yer istasyonuna gönderir. Bu işlem 00:00 ve 12:00 UTC'de olmak üzere günde iki kez tekrarlanır. Türkiye'de 7 ve dünyada 1000 meteoroloji istasyonu tarafından yapılmaktadır.

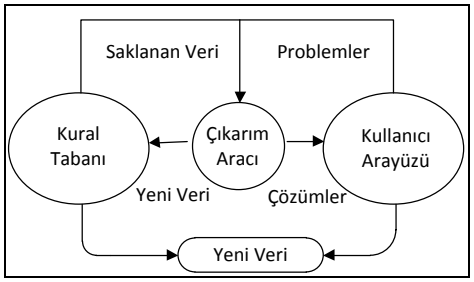
Uydu gözlemleri ise uzayda bulunan uydularla yapılmaktadır. Uydular sensörlerden gelen bilgileri ve fotoğrafları belirli aralıklarda yer istasyonlarına göndererek küresel hava tahminine yardımcı olmaktadırlar.

Hava tahmininin ikinci basamağında ise analiz yer almaktadır. Analiz toplanan tüm verilerin işlenmesi ve basınç merkezlerinin tespitini sağlamaktadır. Hava tahmini ise analiz sonucu elde edilen verilerin de eklenmesiyle yapılır. Tahmin günlük olabileceği gibi haftalık da olabilmektedir.

III. UZMAN SİSTEMLER

Uzman sistem temelde insan düşüncelerini gerçekleştirmek amacıyla bilgisayar tarafından işletilen bir yazılımdır. Uzman sistem geliştirilirken, uzmanların belli bir konudaki bilgi ve deneyimlerinin bilgisayara aktarılması amaçlanmaktadır [1]. Uzman sistem insan bilgisini ve akıl yürütme becerilerini kullanarak gerçek dünya problemlerini çözen zeki bilgisayar yazılımıdır [2]. Uzman sistem yapay zeka prensiplerini kullanır. Uzman sistemin kalbi bilgi tabanıdır. Bilgi tabanı sezgisel ve gerçek verilerden oluşan

bir veri tabanıdır. Uzman sistem basit olarak çözülecek olan problem ile ilgili bilgilerin tutulduğu bilgi tabanı ve bu bilgileri kullanarak çıkarım yapılacak olan karar mekanizmasından oluşur. Çıkarım kuralları basit IF kuralları ile oluşturulur. Uzman sistem blok diyagramı Şekil 1'de verilmektedir.



Şekil 1: Uzman sistem blok diyagramı.

Uzman sistem yeterlilik değerlendirmesinde [2] şirketler için personel alım aşamalarında kullanılabilir. Uzman sistem tarımsal uygulamalarda bitkilerdeki hastalık teşhisi ve denetiminde [3, 4], elektronik cihazların arıza teşhisinde yapay sinir ağı tabanlı uzman sistem kullanılmıştır [5]. Sun Xianglin yüksek gerilim devre kesici tasarımında yapay sinir ağı tabanlı uzman sistem kullanmıştır [6]. Wang Jiangping sondaj işlemlerinde hata teşhisi [7], Li Zhigang su tasarrufu sağlayan sulama sisteminde [8] sinir ağı tabanlı uzman sistem kullanmışlardır. Erken uyarı sistemi olarak uzman sistemi Zhang Weigong kullanmıştır [9]. Fırın sıcaklığının ayarlanmasında [10] ve daha birçok uygulamada uzman sistem başarı ile uygulanmaktadır. Coğrafi bilgi sistemleri uygulamalarında uzaktan bitki hastalık ve zararlısı arama uygulamalarında uzman sistem sinir ağı ile birlikte başarıyla uygulanmıştır [11].

IV. MATERYAL VE YÖNTEM

Bu çalışmada gömülü sistem olarak Micro Framework desteği bulunan Fez Panda II geliştirme kartı kullanılmıştır. Microsoft .Net Micro Framework'ü Smart Personal Object Technology (SPOT) adı altında geliştirmiştir. Microsoft'un .Net Framework, C# dili ve Visual Studio ile birlikte gömülü sistemlerle uğraşan mühendis ve programcılar için daha fazla seçenek ve tasarımlar sunmaktadır. Window CE (Compact Edition) ve diğer gömülü işletim sistemleri aksine .Net Micro Framework düşük güç ve daha düşük maliyet ile ARM7, ARM9 ve Blackfin işlemcilerde kullanılabilir. Micro Framework sadece birkaç yüz kbyte Ram ve Flash/Rom belleğe gereksinim duymaktadır.

.Net Micro Framework programlama için C# programlamadilini sunmaktadır. C# dili kolay öğrenilebilir olması ve C temelli olması sebebiyle programcılar için son yıllarda daha fazla tercih edilmektedir. Gömülü sistemler için uygun mimari olan .Net Micro Framework giderek yaygınlaşan bir şekilde kullanılmaktadır. Gömülü sistem geliştirme kartları tasarlayan firmalarda Micro Framework teknolojisine uygun kartlar geliştirmeyi hızlandırmışlardır[11].

GHI elektronik firmasının ürettiği Fez Panda II geliştirme kartı 72 Mhz çalışma hızında LPC2387 mikrodenetleyicisini barındırmaktadır. 32 bit ARM7 mikrodenetleyici 512 Kb belleğe sahiptir. Dijital girişler haricinde 6 adet donanımsal PWM (Pulse Width Modulation), 6 adet analog giriş ve 2 adet Can (Control Area Network) girişlerine sahiptir. FAT (File Allocation Table) dosya sistemi barındıran Panda II geliştirme kartı 4 adet UART (Universal Asynchronous Receiver/Transmitter) porta sahiptir. Visual Studio 2010 platformunda C# yazılım diliyle kolaylıkla programlanabilir. İşlemcinin kendi kaydedicilerine erişim imkânı sunması da daha kapsamlı kod yazımına imkân vermektedir[4].

Hava tahmini için rüzgar hızı ve yönü ölçümü, nem, sıcaklık ölçümü ve basınç ölçümü yapılmıştır. Rüzgar hızı havanın hareketini ifade eden önemli bir parametredir. Rüzgarlar hızlarına göre çarptıkları yüzeyde basınç oluştururlar. 1m/s hızla esen rüzgarın çarptığı 1 metrekarelik yüzeye yaptığı basınç 0.076 kg'dır.

Rüzgarın hızını ölçmek için el yapımı anemometre kullanılmıştır. Anemometrenin dönüşü için kullanılan motorun devrini okumak için ayrı bir mikrodenetleyici kullanılmıştır. Fez Panda üzerinde encoder okuyucu olmadığı için PIC18F4331 entegresi kullanılmıştır. QEI(Quadrature Encoder Interface) arayüzü bulunan bu mikrodenetleyici ile motorun devri okunarak rüzgar hızı ölçülmektedir. 16 bitlik sayıcısı bulunan denetleyici 2x modunda kullanılmıştır. Sistemin ana kontrol elemanı olan Fez Panda ile Can modülü üzerinden haberleşme sağlanmaktadır. Şekil 2'de bu çalışmada kullanılan kepçe tipi anemometre gösterilmektedir.

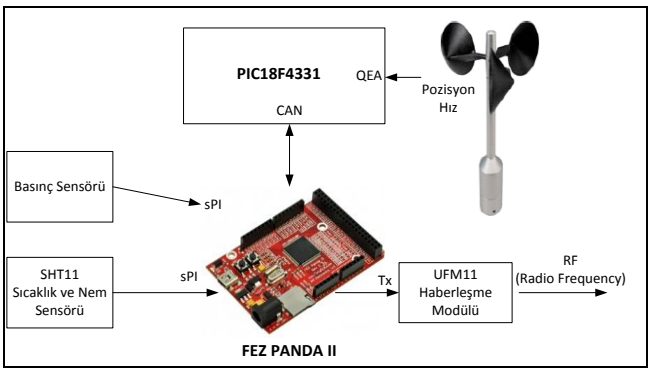
Şekil 2: Kepçe tipi anemometre.

Rüzgarın bulunduğumuz yere doğru geldiği yöne rüzgar yönü denilmektedir. Rüzgar yönü bilgisi yine encoderdan gelen pozisyon bilgisine göre belirlenmektedir.

Hava tahmini için dış ortamın sıcaklık ve nem değerlerini de bilmek gerekmektedir. Bunun için sıcaklık ve nemi birlikte ölçebilen SHT11 algılayıcısı kullanılmıştır. SHT11 algılayıcısı çalışma gerilimi 3.3V - 5V'dur. Sıcaklık hassasiyeti %4 nem hassasiyeti ise %3' tür. Nem ölçüm aralığı 0-100 %, sıcaklık ölçüm aralığı ise -40 ile 100 °C'dir [13].

Hava tahmininde diğer önemli bir parametre olan basınç değişimleri için BMP085 basınç sensörü kullanılmıştır. Bu sensör 0.03 hPa'dan düşük değerde mutlak bir doğrulukla, 300-1100 hPa aralığında ölçüm sunmaktadır. Sensör uygun zemin üzerinde hava basıncı ölçmek için kullanılmıştır.

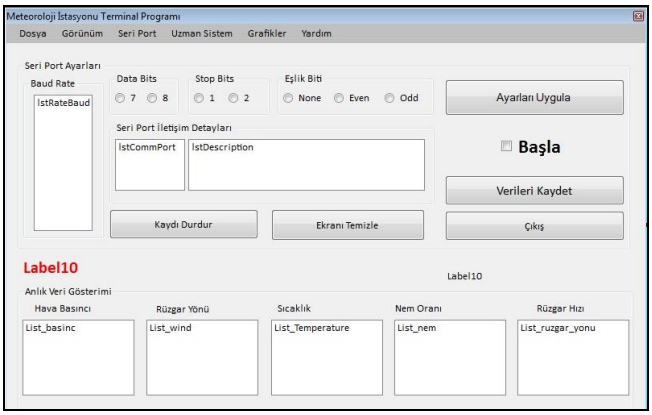
Ölçülen parametrelerin uzakta bulunan sunucu bilgisayara gönderilmesi için Udea firmasını ürettiği kablosuz iletişim modülleri kullanılmıştır. UFM11 iletişim modülü 433 MHz UHF bandında haberleşme sağlamaktadır. Ölçülen parametreler Fez Panda ile UFM11 modül aracılığıyla sunucu bilgisayara gönderilmektedir. Sistemin blok şeması Şekil 3'de verilmektedir.



Şekil 3: Blok diyagram.

Ölçülen parametreler aynı zamanda sürekli olarak 5 Gb'lık SD (Security Digital Memory Card) kart üzerine yazılmaktadır.

Sistemin verici (transmitter) olarak tanımlayabileceğimiz kısmı Şekil 3'de verilmektedir. Alıcı (Receiver) tarafında ise bir web sunucu ve asp.net platformunda hazırlanmış web sayfası bulunmaktadır. Sunucu bilgisayar alıcıdan gelen verileri okumak için seri portuna alıcı bir devre tasarlanmış ve bağlanmıştır. Alıcı devre UFM11 kablosuz haberleşme modülünden oluşmaktadır. Sunucuda çalışan ve seri portu sürekli okuyan bu terminal program aracılığıyla veriler veri tabanına kontrol edilerek kaydedilmektedir. Terminal program C# dilinde kodlanmış bir form uygulamasıdır. Ölçülen parametreler bu program aracılığıyla veri tabanına kaydedilirken aynı zamanda uzman sistem yazılımından da geçirilerek tahmin işlemi de veri tabanına uygun alana kaydedilmektedir. Haberleşme esnasında herhangi bir problem oluşmaması halinde dakikada 5 defa bu işlem gerçekleşmektedir. Yeterli veri toplandığı zaman (günde 3 defa) hava tahmini yapılmakta ve web sayfasında gösterilmektedir. Şekil 4'de terminal program görülmektedir.



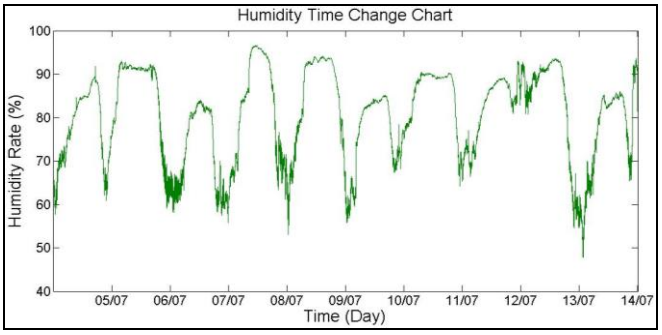
Şekil 4: Terminal programı.

Asp.Net platformunda hazırlanan web sayfasında günde 3 defa yapılan tahmin sonuçları uygun şekilde gösterilmektedir. Web sunucu olarak windows server 2008 kullanılmıştır. Web sayfasında yapılan tahminler şekil 5'de verilmektedir.

Şekil 5: Meteorolojik tahmin web sayfası.

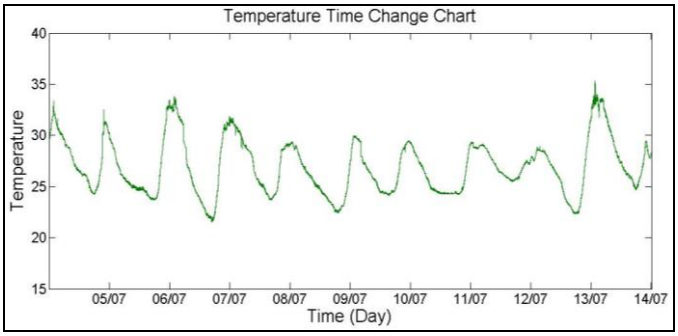
V. DENEYSEL ÇALIŞMA

Uzman sistem için öncelikle gerekli olan uzman bilgisini sağlamak gerekmektedir. Bunun için belirli periyotlar aralığında sıcaklık, nem, basınç ve rüzgar hızı-yönü ölçümü yapılmıştır. Uzman bilgisi için ayrıca Sinop meteoroloji istasyonu yetkililerinden de bilgi alınmıştır.



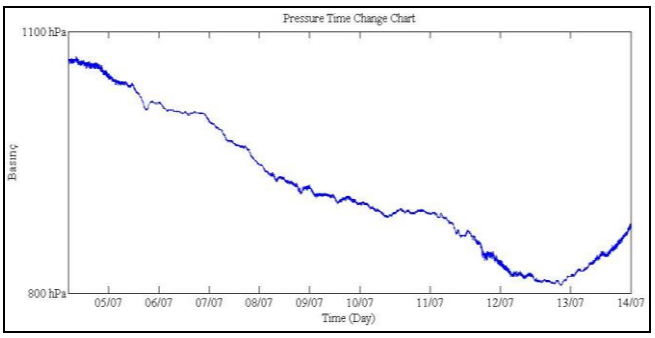
Şekil 6: Nem değişim grafiği.

10 günlük alınan ölçümler sonucunda nem değişimlerinin grafiği şekil 6'da verilmektedir. Nem oranı ortalama %20 oranında değişim göstermektedir.



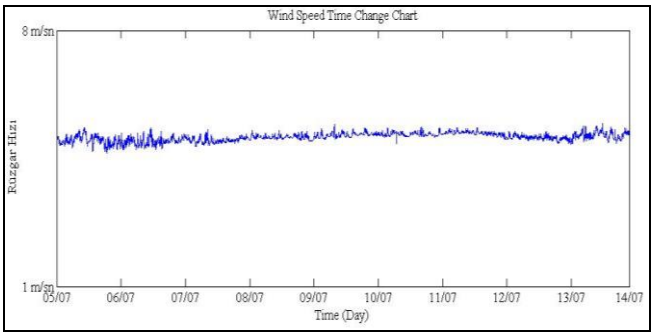
Şekil 7: Sıcaklık değişim grafiği.

Sinop ili Ayancık ilçesinde yapılan bu ölçümlerde sıcaklık değişimi verilen şekil 7'de ortalama değişim 8ºC olarak görülmektedir.



Şekil 8: Basınç değişim grafiği.

Basınç değişim verileri şekil 8'de verilmektedir. Bu verilere göre basınç değişimi genel olarak belirli bir aralıkta sabit olmuştur.



Şekil 9: Rüzgar ölçümü.

Şekil 9'de anemometre ile ölçülen rüzgar hızları ortalama 4,7 m/sn olmaktadır. Yaz aylarında sıcak havanın etkisiyle rüzgar hızı düşük olmaktadır. Rüzgar yönü ise ağırlıklı olarak güney doğu yönünden esmiştir.

Uzman sistem ile gözlem deneyinden elde edilen veriler ışığında ileri zincirlemeli kurallar oluşturulmuştur. Uzman sistem standart if kuralarından oluşmaktadır. Örneğin bir kural açık hava basıncı yüksek ve sıcaklık yaz ortalamaları seviyesinde ise şu şekilde yazılabilir.

if (acik hava basinci = normal and sıcaklık =mevsim normallerinde and nem = mevsim normallerinde and rüzgarYonu != Kuzey Doğu) Then sıcaklık = mevsim normallerinde

şeklinde oluşturulmuştur.

Bu kurallar matris şeklinde düşünüldüğünde çizelge 1'deki gibi olmaktadır.

Çizelge 1: Uzman sistem kuralları.

Parametreler	Yarın yağmur var	Yarın yağmur yok
Soğuk	P(Yağmur | Soğuk)	P(Kuru | Soğuk)
Nem	P(Yağmur | Nem)	P(Kuru | Nem)
Rüzgarlı	P(Yağmur | Rüzgarlı)	P(Kuru | Rüzgarlı)
Hava Basıncı	P(Yağmur | Rüzgarlı)	P(Kuru | Rüzgarlı)
Rüzgar Yönü	P(Yağmur | Soğuk)	P(Kuru | Soğuk)

Bu kurallara bağlı kalarak oluşturulan tahminler yarın havanın nasıl olacağı ile ilgilidir. Gözlem deneyleri kullanılarak yapılan tahminlerde %74 oranında doğru tahmin yapılabilmiştir.

VI. SONUÇ VE ÖNERİLER

Bu çalışma için geliştirilen uzman sistem yazılımında toplam 37 adet kural oluşturulmuştur. Kurallar veya bağlacı ile tasarlanmıştır. Doğru tahmin oranını artırmak için kural sayısı artırılabilir.

Yerel olarak ölçüm yapan ve yerel olarak tahmin yapılan bu çalışmada veriler ve tahmin sonuçları cep telefonları ve diğer taşınabilir cihazlar için uygulama şeklinde hazırlanabilir.

Uzman sistem kullanılarak yapılan hava tahmininde doğru tahmin oranının düşük olması sistemin yeterince uzmanlaşmadığı sonucunu çıkarmaktadır. Girilen verilerin yetersizliği ve tahminde kullanılacak parametrelerin artırılmasıyla sistem daha doğru tahminlerde bulunabilecektir.

VII. KAYNAKÇA

[1] Elmas, Ç., "Yapay Zeka Uygulamaları", Seçkin Yayınevi, Ankara, 21-22 (2007).

[2] Tripathi, P., Islam, S.N., Ranjan, J., Pandeya, T., "Developing Computational Intelligent method for Competence Assessment through Expert System: An Institutional Development Approach", (ICCIC), 2010 IEEE International Conference on Computational Intelligence and Computing Research , India, 1-5, (2011).

[3] Devraj, Renu J., "PulsExpert: An expert system fort he diagnosis and control of diseases in pulse crops", Expert Systems with Applications, 38(9), 11463-11471 (2011).

[4] Zhang Y., Ye Y., and Hu W., "Application of Artifical Neural Network in Agriculture Expert System", Agricultural Machinery Research, 10, 151-153, 2008.

[5] Sun X. and Li J., "Study of High-voltage Breaker Fault Diagnosis Based on Neural Network and Expert System", Coal Mine Machinery and Electron, 6, 42-44, 2008.

[6] Wang J., Bao Z., Meng X., "Application of neural network based expert system in drilling fault diagnosis", Journal of Computer Applications, 29(1), 277-280, 2009.

[7] Li Z. and Zhang W., "The Application Research of the Technique of Neural Network in Water Saving Irrigation", Journal of Shanxi Agriculture Sciences, 35(8), 70-72, 2007.

[8] Zhang W., He J. and Ding D., "Enterprise Risk Premonition System of Property Insurance Company in China Based on BP Neural Network and Expert System", Journal of Xidian University, 19(1), 27-32, 2009.

[9] Nagabhushana K., Nagabhushan R. K., Bhaskar P. and Parvathi S., "An Integrated Expert Controller for the Oven Temperature Control System", Sensor & Transducers Journal, 126(3), 101-109, 2011.

[10] Xiao L., Wang Z., Peng X. and Wu M., "Remote Diagnosis and Control Expert System for Citrus Agricultural Diseases and Insect Pests Based on BP Neural Network ang WebGIS", Second International Conference on Intelligent Computation Technology and Automation, 4, 88-93, 2009.

[11] Song B., Zhang Y., Cheng J. ve Wang J., "Path Following Control of a Mobile Robot via Line of Sight Method", 2th International Conference on Intelligent Human Machine Systems and Cybernetics, 2010.

[12] Fez Panda II Geliştirme Kartı, http://www.ghielectronics.com/downloads/USBizi/USBizi_User_Manual.pdf, (Erişim Tarihi : 05.10.2012).

[13] Albayrak, A., Bayır, R., "Zeki denetimli arı kovanı", 2.Uluslararası Muğla Arıcılık ve Çam Balı Kongresi, Muğla, 177-186 (2010).

Nano/Mikro-Uydular için FPGA Tabanlı 2-FSK Tasarımı

Seyyid M. Dilek, Anilcan Ayranci, Anıl Şeker, Osman Ceylan, H. Bulent Yagci
İstanbul Teknik Üniversitesi
Elektronik ve Haberleşme Müh. Bölümü
Maslak, 34469, İstanbul
e-posta: {dileks, ayrancia, sekera, ceylanos,bulent.yagci}@itu.edu.tr

Özetçe — **Bu bildiride, nano / mikro uydularda modem olarak kullanılabilecek bir yazılım tanımlı radyo (YTR) çalışması sunulmuştur. Yazılımsal radyo işlevleri FPGA üzerinde VHDL ile tasarlanmıştır. Projede küçük uydularda sıklıkla kullanılan 2-FSK (İkili frekans kaydırmalı anahtarlama) modülatör gerçeklenmiştir. Altera Cyclone IV FPGA ile 1200 baud iletişim hızını destekleyen sistemde; UART devresi, DDS yapıları, modülatör gibi bütün alt sistemler yazılımsal olarak geçeklenmiş, sayısal çıkış işareti sayısal-analog (DAC) çevirici ile analog işarete çevrilmiştir. Gerçeklenen bu sistemde PLL ile arttırılan frekans kullanılarak sisteme 150 MHz saat işareti verilmiştir ve sayısal analog çevirici çıkışında 37.6 MHz taşıyıcı işaret elde edilmiştir. Bu işaret analog devre elemanları ile serbest kullanıma uygun olan 437.5 MHz'e çevrilmiştir.**

I. GİRİŞ

Yazılım tanımlı radyo; analog radyolardaki fiziksel süzgeç, karıştırıcı, modülatör gibi yapıların işlevlerinin yazılım yardımıyla gerçeklendiği radyo türüdür. Sistemdeki bütün basamakların yazılım tabanında gerçeklenmesi; uygulama üzerinde kolaylıkla değişiklik yapmaya olanak sağlamakta, farklı fonksiyonları ek bir donanıma gerek kalmadan gerçeklemeye imkan vermektedir. Yazılım tanımlı haberleşme sistemlerinin parametrelerinin de değiştirilebilmesi yazılım tanımlı radyoları uzaktan denetlenebilir hale getirmektedir. Bu da sistem hatalarının düzeltilebilmesine, sistemin geliştirilebilmesine ve iyileştirilmesine olanak sağlamaktadır. Bu özellikler dikkate alındığında fırlatıldıktan sonra sisteme müdahale şansı olmayan uzay sistemleri için büyük esneklik sağlamaktadır.

YTR karasal sistemler de sıklıkla kullanılmaya başlamış, sadece bilimsel değil, endüstriyel uygulamalarda da giderek ivmelenen bir şekilde kullanımı artmaktadır. Buna örnek olarak denizcilikte AIS (otomatik tanımlama sistemi), frekans atlaması askeri haberleşme sistemleri, GPS alıcıları verilebilir. Sayısal sistemler üzerinde tasarım yeteneklerinin ve tecrübelerinin artmasıyla uzayda kullanılacak sistemlerde de YTR teknolojilerinin daha fazla yer bulması beklenmektedir.

Yazılım tanımlı radyonun küçük uydularda kullanılması özellikle iletişim sistemleri açısından aşağıdaki konularda fayda sağlayacaktır:

- Toplam kütle ve hacimde azalış
- Güç tasarrufu
- Maliyet ve tasarım süresinde azalış
- Daha kararlı ve uzun ömür
- Yüksek işlem kabiliyeti
- Uyumluluk (Esneklik)

Tümdevre teknolojilerindeki gelişmeler de bu YTR üzerinde olumlu katılar sağlamaktadır. Daha hızlı ama daha düşük güç tüketen işlemciler sayısal radyoların da yeteneklerinin artmasına izin vermektedir.

Mikro ve nano uydular düşük maliyetleri ve hızlı bir şekilde tasarlanabilmeleriyle sebebiyle sadece bilim dünyasının değil son zamanlarda endüstrinin de dikkatini çekmeye başlamıştır. Tasarımı yıllar süren ve çok yüksek maliyetleriyle genelde sadece hükümetler tarafından desteklenen büyük uyduların yerine yetenekli küçük uydular kullanılması fikri herkes için önemli bir hedef ve istek haline gelmiştir.

Mikro ve nano uydular 300 km'den 2000 km'ye kadar olan yüksekliklerde bulunurlar [1]. Küçük uydular ağırlıklarına göre (Tablo 1) sınıflandırılmaktadır. Uydunun görevine göre ağırlığı ve bulunacağı yörünge yüksekliği değişebilir.

Tablo 1: Küçük uydu sınıflandırması

Sınıf	*Ağırlık (kg)*
Piko Uydular	<1
Nano Uydular	1 - 10
Mikro Uydular	10 - 100

Küçük uydulardaki en önemli sorun hacim kısıtıdır. Bu sorun 2 şekilde karşımıza çıkmaktadır: Çok sayıda akü yerleştirilememesi, güneş paneli sayısının kısıtlı olması. Akü sayısının fazla olmaması uydu ömrünü azaltır ve sistemlerin kesintisiz olarak çalışabileceği süreyi kısıtlar. Güneş panelleri de sisteme sağlanacak enerjiyi belirlediği için özellikle güç tüketimi konusunda önemli sınırlar getirir. Bunun en önemli etkisi iletişim yapısında uydu üzerinde yüksek güçlü çıkış alınamaması olarak gösterilebilir. Ayrıca uyduların hızlarının çok yüksek oluşu da (5-15 km/saniye) uydu iletişiminin başarımını etkiler. Doppler Etkisi'nden dolayı meydana gelen frekans kayması bant genişliğini ve modülasyon tipinin seçimini kısıtlar. Esnek bir yapı ile bu olumsuz etkiyi en aza indirmek, önemli bir katkı sağlayacaktır.

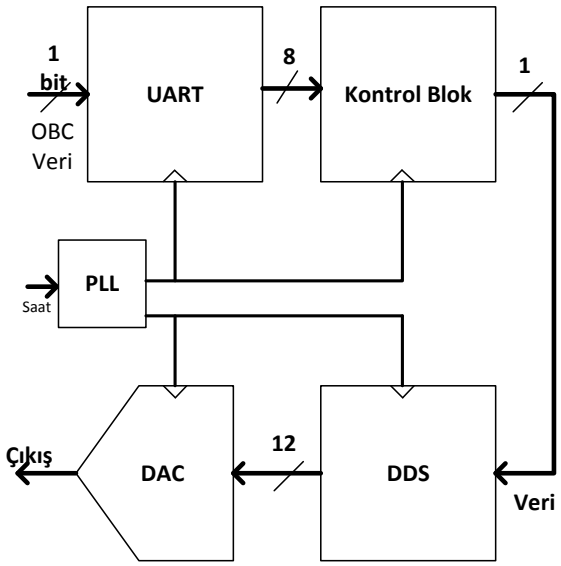
Projenin genel amacı nano uydular için; küçük hacimli ve düşük enerji tüketimine sahip, etkili ve hızlı iletişim kurabilen ve yeniden programlanabilir bir yazılım tanımlı radyo modem tasarımını gerçeklemektir.

II. 2-FSK MODÜLATÖR YAPISI

2-FSK, sayısal bilgi işaretinin önceden tanımlanmış taşıyıcı iki frekansın kaydırılması sonucunda oluşan modülasyon işlemidir. Taşıyıcı frekanslar 1 ve 0'a karşılık düşmektedir. Nano/mikro uydu uygulamlarında 2-FSK modülasyonu aşağıdaki sebeplerden dolayı tercih edilmektedir;

- FSK işareti amatör radyocular tarafından kolaylıkla çözülebilmesi,
- Doppler kaymasına daha dayanıklı olması,
- Daha az band genişliği ihtiyacı,

FSK modülatör yapısı, PLL (Faz kilitlemeli çevrim), UART, Kontrol ve DDS (Doğrudan Sayısal Sentezleyici – Direct Digital Synthesizer) bloklarından oluşmaktadır (Şekil 1).



Şekil 1: Modülatör Yapısı.

A. PLL

Sayısal tasarımlarda, saat darbesi problemlerini çözebilmek için, PLL kullanılmaktadır. PLL, saat darbesi çarpımı ve bölümü, faz kaydırması, programlanabilir görev döngüsü ve harici saat darbesi çıkışı gibi işlemleri saat darbesi yönetimini ve kontrolünü sağlar. Altera firmasının Quartus II yazılımı [2], FPGA'ın PLL özelliğini, harici bir yapı kullanmadan aktif hale getirmemizi sağlar. PLL genellikle dahili cihaz saat darbesini harici saat darbesi ile uyumluluğu konusunda, dahili saat darbesini, harici saat darbesinden daha yüksek frekansla üretmek amacıyla ve saat darbesi gecikmelerini azaltmak için kullanılan bir yapıdır.

B. UART

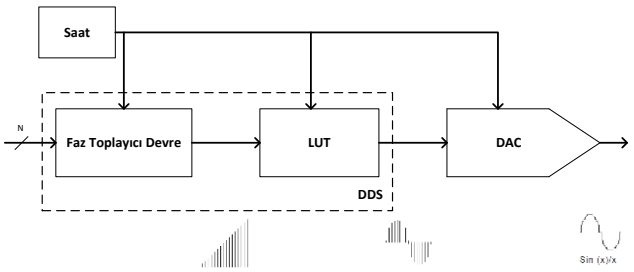
UART (Universal asynchronous receiver/transmitter) nano uydular için kullanılan temel bir seri iletişim arayüzüdür. Uydu bilgisayarı, almaç gibi bilgi kaynaklarından gelen sayısal verinin haberleşmesi için uyduların birçok alt sisteminde kullanılmaktadır. Bu projede, iletilmek istenilen veri UART modülü tarafından alınmaktadır.

C. Kontrol Bloğu

Kontrol Bloğu, içerisinde paralel-seri dönüştürücü devre, saat darbesi bölücüsü ve MUX bulunmaktadır. Blok, UART bloğundan gelen 8 bitlik verinin 1 bitlik veriye dönüştürülmesinde kullanıdı.

D. DDS

DDS, modülatörün temel yapılarından birisidir. DDS tekniği, gerektiğinde kullanılmak üzere örneklenmiş sinüs ve kosinüs işaret dalgasının ROM'da saklanan bir tarama tablosundan (LUT) sırayla okunarak oluşturulur. Saat darbesinin her bir periyodunda, faz algılayıcısı ile belirlenen değere göre veri kontrolünü sağlar (Şekil 2).

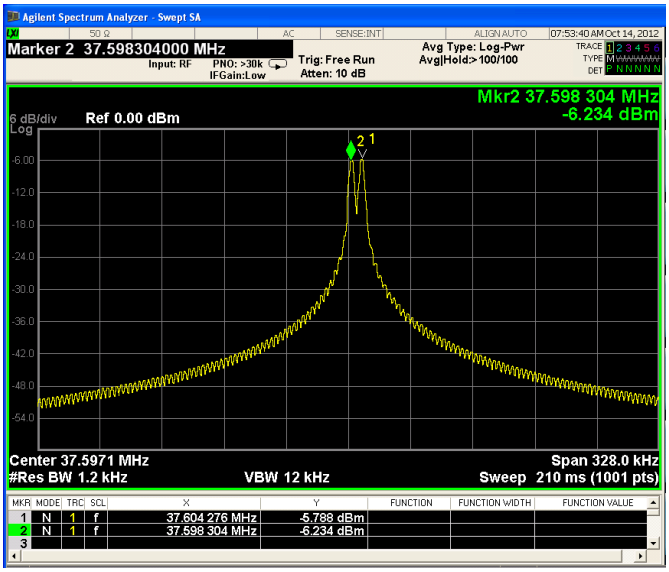


Şekil 2: DDS yapısı.

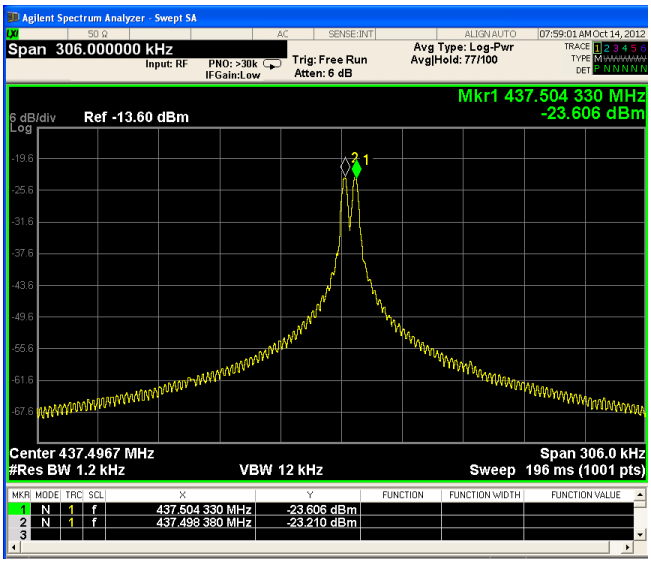
III. MODÜLATÖR SİSTEMİNİN UYGULANMASI

Uydu bilgisayarı ya da almaçlardan alınan seri 1200 baud hızındaki veri, UART arayüzü yardımı ile kontrol bloğuna 8 bitlik olarak aktarılır. Kontrol bloğu alınan 8 bitlik veriyi 2-FSK yapısına uygun bir şekle sokmak için paralel-seri dönüştürücüler, saat darbesi yönetimi ve MUX kullanır. FPGA entegresindeki son işlem DDS bloğudur. DDS bloğu alınan 1 ve 0'lara göre önceden belirlenen taşıyıcı frekansları üretir.

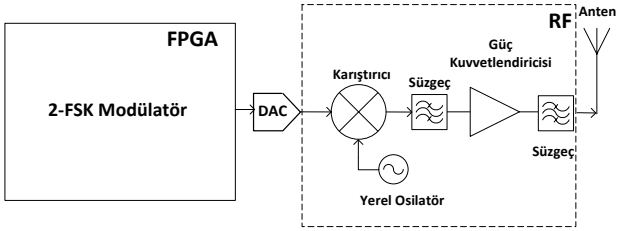
DDS yardımı ile üretilen 2-FSK modülasyonlu işaret 12 bit çözünürlük ve 165 MSPS örnekleme hızı ile paralel işlem özelliğine sahip olan DAC [3] entegresine aktarıldıktan sonra, DAC'ın çıkışından 37.6 MHz modüleli işaret elde edilmiştir (Şekil 3). DAC'ın çıkışından alınan analog işaret, bir yerel osilatör ve karıştırıcı kullanılarak lisanssız banda getirilmiştir (Şekil 4). DAC çıkışının yeteri kadar yüksek frekanslı olmasından dolayı taşıyıcı frekans tek bir ara kat ile istenilen frekansa kolaylıkla yükseltilmiştir, bu sayede devre elemanı sayısı ve güç tüketimi azaltılmıştır (Şekil 5). Test aşamasında olan analog yapı bu çalışmada modüller kullanılarak hazırlanmıştır. Düşük gürültülü kristal osilatör ve pasif karıştırıcı tercih edilmiştir. Bu modüllerin devre elemanları da mevcut olup projenin ilerleyen aşamalarında aynı kart üzerine yerleştirilmesi planlanmaktadır. Testlerde ara kat ve güç kuvvetlendiricisi kullanılmamıştır, bu nedenle ilgili grafiklerde çıkış seviyesi düşüktür.



Şekil 3: DAC'ın Çıkışı (Agilent N90101A EXA)



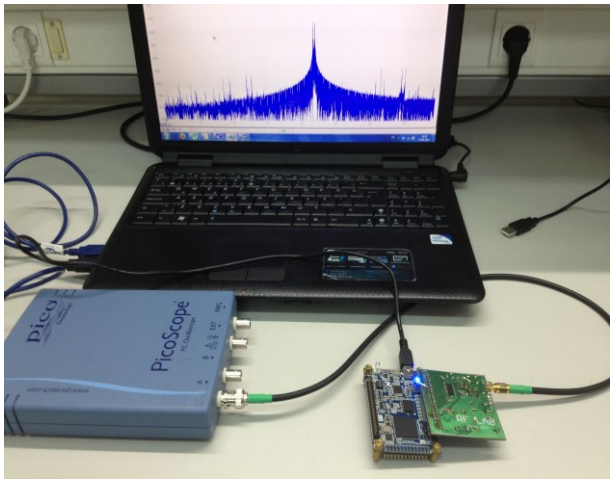
Şekil 4: RF ön uç çıktısı



Şekil 5: Sistem blok şeması

IV. SONUÇLAR

Yazılım tanımlı radyo çalışması FPGA üzerinde başarılı bir şekilde gerçeklenmiş, sayısal işaret analoğa dönüştürülerek işaret analizi yapılmıştır. Sistemin genel başarımını test etmek için modüller yardımıyla işaret taşıyıcı frekansı serbest banda (437.5 MHz) taşınmıştır. Çalışmanın ilerleyen aşamasında FPGA, DAC ve diğer analog RF devre elemanlarını tek bir elektronik kart üzerinde tasarlanması planlanmaktadır. Sistem masaüstü model olarak başarılı bir şekilde çalışmaktadır (Şekil 6). Çalışmanın ilerleyen aşamasında FPGA, DAC ve diğer analog RF devre elemanlarını küçük boyutlu tek bir elektronik kart üzerinde tasarlanması planlanmaktadır. Daha düşük güç ile daha küçük boyulu bir yazılım hazırlanması da hedefler arasındadır. Ayrıca sistemin düşük yörüngeli bir uyduda kullanılacak olması çeşitli devre elemanlarının kullanımına imkan vermediği için devrenin normal şartlarda daha iyi gerçeklenebilecek olmasına rağmen tasarımcıları bir miktar kısıtlamaktadır.



Şekil 6. Masaüstü Model

V. KAYNAKÇA

[1] Committee on Implications of Emerging Micro- and Nanotechnologies, National Research Council. "1. Introduction." *Implications of Emerging Micro and Nanotechnology*. Washington, DC: The National Academies Press, 2002, p.28.

[2] Altera Corporation, Using PLLs in Cyclone Devices, Mayıs 2008.

[3] Burr-Brown Product. SpeedPlus 12-bit, 165 MSPS Digital to Analog Converter, Mayıs 2002.

[4] Agilent Technologies, N9010A EXA X Series Signal Analyzer, Nisan 2012.

Yazar İndeksi