VİDEO ASLININ VE BÜTÜNLÜĞÜNÜN FPGA İLE DOĞRULANMASI

VIDEO AUTHENTICATION AND VIDEO INTEGRITY USING FPGA'S

ÖZLEM UYANIK

Hacettepe Üniversitesi

Lisansüstü Eğitim - Öğretim ve Sınav Yönetmeliğinin

ELEKTRİK ve ELEKTRONİK Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ

olarak hazırlanmıştır.

2011

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma jürimiz tarafından ELEKTRİK ve ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI 'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Başkan	:……...
	Prof. Dr. H.Selçuk Geçim
Üye (Danışman)	:……...
	Doç. Dr. Ali Ziya Alkar
Üye	:……...
	Yrd. Doç. Dr. Kayhan İmre
Üye	:……...
	Yrd. Doç. Dr. Umut Sezen
Üye	:……...
	Yrd. Doç. Dr. Mehmet Demirer

ONAY

Bu tez Hacettepe Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği'nin ilgili maddeleri uyarınca yukarıdaki jüri üyeleri tarafından ...../...../ 2011 tarihinde uygun görülmüş ve Enstitü Yönetim Kurulunca ...../...../ 2011 tarihinde kabul edilmiştir.

Prof.Dr. Adil Denizli
FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

VİDEO ASLININ VE BÜTÜNLÜĞÜNÜN FPGA İLE DOĞRULANMASI

ÖZLEM UYANIK

ÖZ

İçinde çok sayıda video işlemini gerçekleştiren ve video sinyali üzerinde görüntü işleme algoritmaları çalıştıran güvenlik sistemlerinde, gelen video görüntüsünün beklenen video kaynağından geldiği varsayılmaktadır. Oysa video kaynağının bir başka kaynak ile yer değiştirilebilme, görüntü donması, görüntü kaynağı ile bağlantısının kesilmesi olasılıkları sonucunda güvenlik sistemi etkisiz hale gelmektedir. Bunun yanı sıra donanımsal hataların bazı kritik durumlarda farkedilememesi kötü sonuçlara sebep olabilmektedir. Bu nedenle, videonun doğrulanmasına yönelik farklı yöntemler kullanılmaktadır. Sayısal filigran basma ve sayısal imza görüntü iletimindeki hataları tespit etmede kullanılan yöntemlerdendir. Genellikle sistemlerde video kaynağı ile iletişimin tek yönlü olması, bilgisayar ağlarında sık kullanılan aslının doğrulanması yöntemlerinden farklı çözümler gerektirmektedir. Varolan çözümler yüksek çözünürlükler için uygulanamamaktadır. Ayrıca filigran basma uygulaması görüntünün en düşük bitlerinde değişiklik yaparak çok önemli olmasa da görüntünün orjinalini bozarak görüntü kalitesinden ödün vermektedir. Bu uygulama düşük çözünürlüklü video uygulama sonucu farkedilmese de yüksek çözünürlüklü video gönderiminde bozulma hissedilecektir. Bu çalışmada, güvenlik sistemine doğrudan eklenebilen, üzerinde video çözücü/kodlayıcı ve FPGA bulunan bir kart tasarlanabilip videonun aslının ve görüntü bütünlüğünün doğrulanması işlemlerinin özgün bir şekilde çerçeve numaralandırılması ve güvenlik anahtarlı özet alma algoritması (SHA-1) ile gerçekleştirilebileceği gösterilmiştir. Yapılan bu çalışmada çok yüksek çözünürlüklerde bile videonun kalitesi bozulmamaktadır.

Anahtar Kelimeler: Güvenlik sistemi, video bütünlüğü, videonun aslının doğrulanması (authentication), özet alma işlemi (hash), SHA-1, FPGA.

Danışman: Doç. Dr. Ali Ziya Alkar, Hacettepe Üniversitesi, Elektrik ve Elektronik Mühendisliği

VIDEO AUTHENTICATION AND VIDEO INTEGRITY USING FPGA'S

ÖZLEM UYANIK

ABSTRACT

In security systems which apply image processing algorithms on video signals and with many video processes running inside, the received video image is assumed to be coming from the expected video source. However, the security system might be compromised by changing the video source with other sources, or due to video freezing or termination of the video link. These kinds of systems require different authentication methods, unlike the ones mostly used in computer networks, due to limitations of the one way communication with the video source. Digital watermarking and digital signature are the most popular methods for video authentication. Owing to lack of processor speed, output performance could be low in high video resolutions, and video degredation at the bit level due to these authentication applications may be visually noticable. In this study, it is aimed to perform video authentication and image integrity verification operations by designing a FPGA-based card which can be added to enhance the security of system directly, using authentication techniques of assigning frame numbering in a unique way and security Hash-Based Message Authentication using SHA-1 algorithm.

The completed design contains an embedded video decoder/encoder. Functional tests indicate unnoticable video degredation even at very high video resolutions.

Keywords: Security system, video integrity, video authentication, The Keyed-Hash Based Message Authentication Code (HMAC), SHA-1, FPGA

Advisor: Assoc. Prof. Dr. Ali Ziya Alkar, Hacettepe University, Department of Electrical and Electronics Engineering

TEŞEKKÜR

Bu tezin gerçekleştirilmesinde bilgi ve deneyimlerini paylaşıp, destek ve yorumlarıyla katkıda bulunan tez danışmanım Doç. Dr. Ali Ziya ALKAR'a teşekkürü bir borç bilirim.

Tez süresince desteklerini esirgemeyen, bilgi ve tecrübesiyle beni en doğru şekilde yönlendiren Dr. İsmail Enis UNGAN'a teşekkürlerimi sunarım.

Hayatım boyunca belirlediğim hedeflere ulaşacağıma ve başarılı olacağıma inanan, her zaman yanımda olup, beni destekleyen aileme ve tez çalışmalarım sırasında bana destek olan bütün arkadaşlarıma çok teşekkür ederim.

İÇİNDEKİLER DİZİNİ:

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

KISALTMALAR DİZİNİ:

AHDL	**:**	Altera Hardware Description Language (Altera Donanım Tanımlama Dili)
CPLD	**:**	Complex Programmable Logic Device (Karmaşık Programlanabilir Mantık Aygıtı)
DE	**:**	Active Video Synchronization (Aktif Video Senkronizasyon)
DRM	**:**	Digital Rights Management (Sayısal Haklar Yönetimi)
DVI	**:**	Digital Visual Interface (Sayısal Görüntü Arayüzü)
FIPS	**:**	Federal Information Processing Standards Publications (Federal Bilgi İşleme Standart Yayınları)
FPGA	**:**	Field Programmable Gate Array (Alanda Programlanabilir Kapı Dizileri)
HDCP	**:**	High-bandwidth Digital Content Protection (Yüksek Bant Genişliğinde Sayısal İçerik Koruma)
HDL	**:**	Hardware Description Language (Donanım Tanımlama Dili)
HDMI	**:**	High Definition Multimedia Interface *(*Yüksek Tanımlı Çoklu Ortam Arayüzü)
HMAC	**:**	The Keyed-Hash Based Message Authentication Code (Anahtarlı Özet Fonksiyonu Tabanlı Mesaj Doğrulama Şifresi)
HSYNC	**:**	Horizontal Syncronization (Yatay Senkronizasyon)
IPAD	**:**	İnner Padding (İç Dolgulama)
LCD	**:**	Liquid Crystal Display (Sıvı Kristal Ekran)
LUT	**:**	Look Up Table (Başvuru Çizelgesi)
ITL	**:**	Information Technology Laboratory (Bilgi Teknolojileri Laboratuvarı)
LFSR	**:**	Linear Feedback Shift Register (Doğrusal Geri Bildirim Kaymalı Kaydedici)
NBC	**:**	National Broadcasting Company (Ulusal Radyo Yayın Şirketi)
NIST	**:**	National Institute of Standards and Technology (Ulusal Teknoloji ve Standartlar Enstitüsü)
NTSC	**:**	National Television Standards Committee (Ulusal Televizyon Standartları Komitesi)
OPAD	**:**	Outer Padding (Dış Dolgulama)
PAL	**:**	Phase Alternating Line (Faz Değişkenli Hat)
PRBS	**:**	Pseudo Random Binary Sequence (Sözde Rastgele İkili Dizi)

PLL	:	Phase Locked-Loop (Faz Kilitlemeli Döngü)
PC	:	Personal Computer (Kişisel Bilgisayar)
RAM	:	Random Access Memory (Rastgele Erişimli Bellek)
RCA	:	Radio Corporation of America (Amerika Radyo Kurumu)
RF	:	Radio Frequency (Radio Frekansı)
RGB	:	Red Green Blue (Kırmızı Yeşil Mavi)
S-Video	:	Separate Video (Ayrık Video)
VHDL	:	VHSIC HDL (Çok Yüksek Hızlı Tümleşik Devreler için Donanım Tanımlama Dili)
VHSIC	:	Very High Speed Integrated Circuits (Çok Yüksek Hızlı Tümleşik Devreler)
VGA	:	Video Graphic Array (Video Grafik Dizisi)
VSYNC	:	Vertical Syncronization (Dikey Senkronizasyon)
XGA	:	Extended Graphics Array (Genişletilmiş Grafik Dizisi)
XOR	:	Exclusive OR (Dışlayıcı Veya)

TÜRKÇE- İNGİLİZCE SÖZLÜK DİZİNİ:

Benek	**:**	Pixel
Dikey Senkronizasyon	**:**	Vertical Synchronization (VSYNC)
Yatay Senkronizasyon	**:**	Horizonal Synchronization (HSYNC)
Aslının Doğrulanması	**:**	Authentication
Görüntü Bütünlüğü	**:**	Video Integrity
Özet Alma	**:**	Hash
Filigran Basma	**:**	Watermarking
Yarı-kırılgan	**:**	Semi-fragile
Dolgulama	**:**	Padding
Güvenlik Kamerası	**:**	Surveillance Camera
Bellek	**:**	RAM
Çerçeve	**:**	Frame
Şifreleme	**:**	Cryptography
Mantık	**:**	Logic
Akış	**:**	Flow
Durum	**:**	State
Tümleşik	**:**	İntegrated
Anahtar	**:**	Key
Bağlayıcı	**:**	Connector
Karartma Aralığı	**:**	Blanking Interval
Parmak İzi	**:**	Finger Print
Atım	**:**	Pulse
Zayıf Çarpışma Direnci	**:**	Weak Collision Resistance
Kaymalı Kaydedici	**:**	Shift Register
Güçlü Çarpışma Direnci	**:**	Strong Collision Resistance

1. GİRİŞ

Günümüzde güvenliğe verilen önem her geçen gün artmaktadır. Güvenlik sistemlerinde video uygulamalarının kullanımı vazgeçilmez bir hal almıştır. Mobese kameraları ile hemen hemen bütün sokaklar izlenmekte, bütün mağazalar ve iş merkezlerinde güvenlik kamerası bulunmaktadır. Gözlenmek istenen bölgelere, hareketli ya da hareketsiz, çoğunlukla çok sayıda kamera yerleştirilerek video sinyalleri gözlem odasına ulaştırılmakta ve çok kanallı video alıcılarından geçirilerek bir ya da daha çok monitör üzerine aktarılmaktadır. Gözlemlerin insan kullanılarak yapılması çok kanallı video görüntülerinin sürekli izlenmesini zor hale getirmekte ve güvenliği azaltmaktadır. Bu sorunun önüne çoklu kanal giriş video sinyalleri üzerinde görüntü işleyen ve tehdit olabilecek oluşumları algılayan algoritmalarla geçilmektedir. Bunun yanısıra yüksek çözünürlüklü kameraların sağladığı video sinyalleri üzerinden nesne belirlemeye yönelik tanımlayıcı veriler elde edilebilmekte ve kayıt altına alınarak delil oluşturabilmektedir. Böylece gelişmiş donanım ve yazılım ile güvenlik seviyesi arttırılabilmektedir.

Güvenlik sistemlerinde video üzerinde yapılan uygulamaların karmaşık yapısına bakılmadan video sinyallerinin güvenilir olduğu varsayılmaktadır. Video sinyallerinin kaynakları olan kameralar çoğunlukla güvenlik merkezine uzak yerlerde bulunmaktadır. Böylece video sinyallerinin taşındığı kablolu ya da kablosuz iletim ortamına erişilebilmekte ve farklı video kaynak sinyalleri ile iletim hattına girilerek güvenlik sistemi etkisiz duruma getirilebilmektedir. Video sinyalinin beklenen kaynaktan geldiğini algılamak için video aslının doğrulanması gerekmektedir. Kaynak aynı bile olsa görüntünün bir bölümünde değişiklik yapıp güvenliğin etkisiz hale getirilebilmesi olasıdır. Bu durumu engellemek için videonun bütünlüğünün doğrulanması da vazgeçilmez bir gereksinimdir.

Video sinyallerinin farklı kaynaktan geldiğini algılamak ya da donanım kaynaklı bir problem olup olmadığına karar verebilmek için çeşitli algoritmalar kullanılmaktadır [18, 20]. Video üzerinde yapılan doğrulama çalışmaları iki kategoriye ayrılır. Bunlar veri saklama metodu olarak da adlandırılan sayısal filigran basma metodu [17, 18, 23, 24] ve sayısal imza metodlarıdır [7, 18, 25, 26]. Sayısal filigran basma uzun yıllardan beri bandrol, çek gibi değerli kağıtların üzerinde kullanılan, normal ortamda gözükmeyip sadece ışığa tutulduğunda gömülmüş işareti gösteren filigran basma

teknolojisinin sayısal ortamdaki karşılığıdır. Sayısal filigran basma teknolojisi ses, resim, görüntü dosyalarına uygulanabilmektedir. Filigran basma işleminin temelinde verinin değiştirilmesi yer almaktadır. Bu değişikliğin kullanıcı tarafından farkedilmediği kabul edilmiştir [19]. Teorik olarak filigran basmanın veri üzerindeki değişikliği kalıcıdır. Ayrıca filigran basma işlemi yüksek işlemci gücü gerektirmektedir. Özellikle yüksek çözünürlüklü video sinyallerinde kullanımı maliyet ve güç tüketiminin yüksek olması nedeniyle uygun değildir [17]. Özet algoritmasının mantık devreleriyle kolayca gerçeklenebilmesi nedeniyle yüksek çözünürlüklü video sinyallerinde kolaylıkla ve filigran basma yöntemine göre çok daha az maliyetle gerçekleştirilebilmektedir. Bu çalışmada video aslının doğrulanmasında özet algoritması kullanılarak çok yüksek çözünürlüklü video sinyallerinde kullanımı hedeflenmiştir.

Video aslının doğrulanmasında kullanılan bir başka yöntem ise sayısal imzadır. Video sinyalleri çeşitli paket yapısında ağ yada kablosuz iletim hattı üzerinden gönderilebilir. Ancak bu paketler şekilde yapılan veri gönderiminde video paketlerinin kaybolma olasılığı yüksektir. Video doğrulaması yapılarak bu problemin farkedilir hale gelmesi sağlanmıştır [26].

Hem video aslını hem de video bütünlüğünü doğrulayabilecek bir koruyucu sistemin gereksinimlerini ortaya çıkarmak için güvenliğe yapılabilecek olası saldırılar için çeşitli senaryolar düşünülmüştür. İlk olarak kameradan gelen görüntünün harici bir depolama aygıtına kaydedildiğini düşünelim. Kaydedilmiş görüntü üzerinde istenilen video çerçeveleri çıkarılıp yerine yenisi eklenebilir. Cezai durumun söz konusu olduğu bazı koşullarda kamera görüntüleri delil olarak kullanılmaktadır ve güvenilirliği oldukça önem taşımaktadır [8]. Algoritmanın entegre olduğu kameradan kayıt yapılması durumunda eklenilen ya da çıkarılan video çerçevelerinin tespiti mümkün olacaktır. Depolama aygıtından yürütülen görüntü monitöre gelmeden önce video aslının doğrulanmasının yapacağı kartta uğrayacaktır. Eğer özgün videonun üzerinde oynama yapılmışsa kullanılan özet fonksiyonu ile kolaylıkla tespit edilebilecektir [25, 26].

Kötü bir durumun yaşanmasına neden olabilecek en önemli problemlerden biri görüntülenmek istenen kameranın koparılması ve başka bir kameranın sisteme eklenmesidir. Hiçbir doğrulama algoritmasının çalışmaması durumunda değişiklik farkedilemez. Kamera ve monitör tarafında özet alma fonksiyonunun çalıştığı

durumda kamera değişikliği anında tespit edilebilecektir. Bu sayede gözlemlediğimiz görüntünün istenilen görüntü olduğu doğrulanacaktır [25].

Görüntü donması, bu çalışma sonrasında kamera ile entegre hale getirilen video güvenlik kartında meydana gelebilecek bir durumdur. Kamera sistemlerinde çeşitli video işleme algoritmaları çalışmaktadır. Bu algoritmaların kullanılabilmesi için genellikle harici bir belleğe ihtiyaç duyulmaktadır. Kartın donanımında oluşabilecek bir hata sistemin düzgün çalışmasını engelleyerek harici bellekte son kaydedilen çerçevenin devamlı olarak gönderilmesine neden olabilir. Bu hata durumunu engellemek amacıyla çerçeve sayacı ve sözde rastgele sayı dizisi (PRBS) kullanılarak hesaplanan mesaj özeti video işleme uygulamalarından önce tamamlanır ve video verisine eklenir. Görüntü donması durumunda çerçeve sayacı ve PRBS mesaj özeti sonucu değişmeyecektir. Üst üste aynı olduğu belirlenen bu değerler sonucunda kullanıcı uyarılacaktır. PRBS kullanılarak oluşturulan özet, sabit bir görüntü geldiğinde bellek kaynaklı bir donma olduğuna karar vermek ve sisteme tesadüfi durum katmak amacıyla sisteme eklenmiştir. Özellikle askeri sistemlerde görüntü donması durumunun anında tespit edilmesi gerekmektedir. Aksi durumda can ve mal kaybına neden olabilecek ortamlar oluşabilir.

Yukarıdaki senaryolar sonucu aşağıdaki gereksinimler ortaya çıkmış ve tasarlanacak sistemde gerçekleştirilmesi hedeflenmiştir.

- 1080p gibi yüksek çözünürlüklü video verilerine uygulanabilir.
- Video sinyaline özet fonksiyonu uygulanacak ve hesaplamalar gizli anahtar kullanılarak yapılacaktır.
- Video çerçevesinin ardışıklığı ve sürekliliği için çerçeve sıra numarası hesaplanacaktır.
- Video belleklerinde oluşacak video sinyalleri için PRBS kullanılacaktır.
- Var olan sistemlere kolay entegre edilebilmesi ve donanım kopyalamaya karşı güvenliği için sistem FPGA üzerinde gerçeklenecektir.

Kullanılan şifreleme algoritması, video standardı ve videonun taşınma yapısına ait genel bilgiler ikinci bölümde yer almaktadır. Sistem tasarımının parçaları üçüncü

bölümde incelenecektir. Tasarım bloklarının FPGA içerisinde nasıl gerçeklendiği, durum makinalarının çalışma yapısı dördüncü bölümde, gerçeklenen gösterim beşinci bölümde anlatılmaktadır. Yapılan çalışmalar sonucu elde edilen veriler ve çalışmanın kazanımlarının değerlendirilmesi altıncı bölümde anlatılmaktadır.

2. TASARIMDA KULLANILAN BİLGİLER

2.1. Video ile ilgili Temel Bilgiler

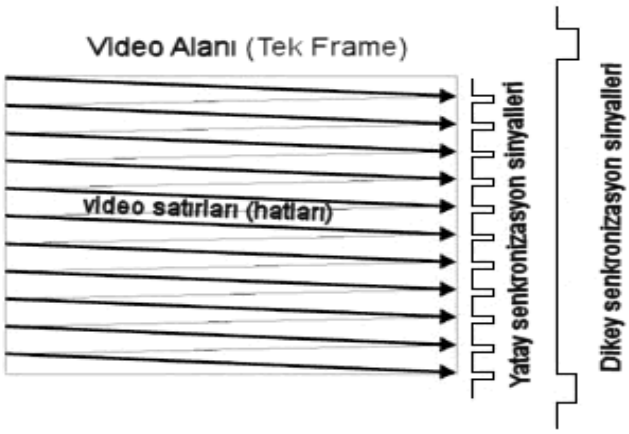
Video formatı, genellikle resimleri oynatmak için kullanılan çeşitli depolama formatı olarak adlandırılır. Dünyada kullanılan çeşitli video standartları bulunmaktadır. Video yapısında değişmeyen iki bileşen bulunmaktadır. Bunlar; senkronizasyonu sağlamak için kullanılan gömülü ya da ayrı olan senkronizasyon sinyalleri ve görüntü verileridir.

Bu bölümde dünyada kullanılan standartların bir kısmı ve videonun genel çalışma prensibinden bahsedilmiştir.

2.1.1. Video Zamanlaması

Videonun ekranda gösterilebilmesi için görüntü verisi haricinde senkronizasyon sinyallerine de ihtiyaç vardır. Senkronizasyon bilgileri görüntünün doğru zamanlamalarla ekranda düzgün bir şekilde görüntülenmesini sağlarlar. Üç adet senkronizasyon bilgisi vardir [27]. Bunlar:

- Dikey senkronizasyon (VSYNC)
- Yatay senkronizasyon (HSYNC)
- Aktif alan senkronizasyon (DE)'dir.



Şekil 2.1.1-1 Yatay ve dikey senkronizasyon sinyalleri

Şekil 2.1.1-1'de görüldüğü üzere her yeni çerçevenin geldiğine dair bilgi dikey senkronizasyon (VSYNC) sinyalinden elde edilir. Dikey senkronizasyon sinyali ilgili

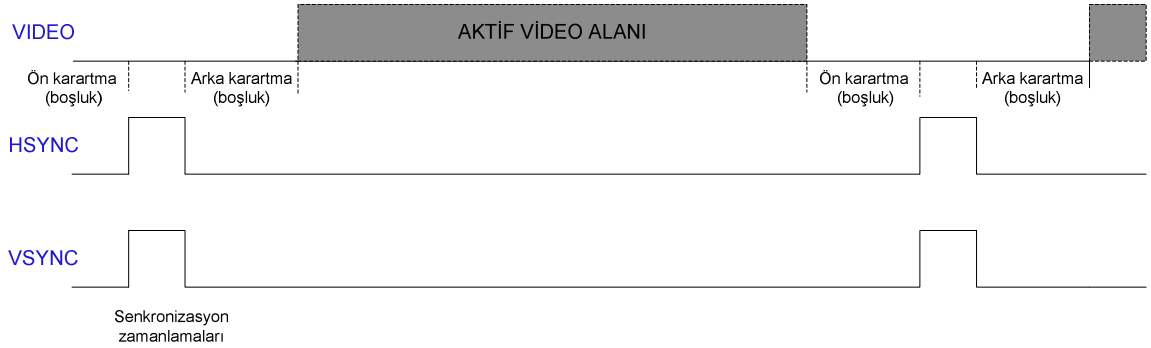
standarda göre mantık '0' ya da mantık '1' olabilir. Dikey senkronizasyona ait satır sayısı da kullanılan standarda bağlıdır.

Her yeni satırı belirten senkronizasyon sinyali yatay senkronizasyon (HSYNC) olarak adlandırılır. Yatay senkronizasyon sinyali ilgili standarda göre mantık '0' ya da mantık '1' olabilir. Yatay senkronizasyona ait benek sayısı aynı şekilde kullanılan standarda bağlıdır.

Ekrana basılacak görüntünün alanının belirlemesi için aktif video senkronizasyon sinyali kullanılır. Aktif video alanı ilgili standartlarda tanımlanmıştır. Aktif video sinyali (DE) görüntünün ekranda gösterilmek istendiği kısımda her zaman '1' dir.

Yatay ve dikey senkronizasyon bilgileri video çerçevesinin ekranda görülmeyen karartma aralığında bulunur. Karartma alanı satır ve benek bazında olmaktadır. Kullanılan standarda göre bu alanın genişliği değişmektedir.

Video zamanlama yapısı Şekil 2.1.1-2'de genel olarak gösterilmiştir.



Şekil 2.1.1-2 HSYNC ve VSYNC sinyalleri ve karartma aralıkları

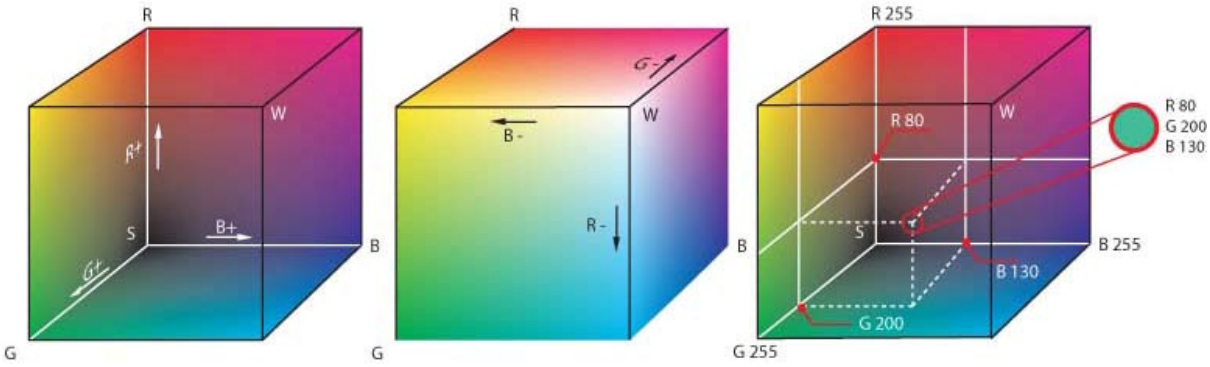
Senkronizasyon sinyallerinin videoya eklenmesi farklı şekillerde olabilmektedir [9]. Örneğin kompozit video sinyali, yatay ve dikey senkronizasyon, parlaklık ve renk sinyallerinin bileşiminden oluşan tek bir sinyal olduğundan senkronizasyon bilgisi, bütün sinyalin bir bölümü olarak taşınmaktadır. Ayrıca kompozit videoda yatay senkronizasyon bilgisi kısa ve negatif yönlüdür. Parlaklık ve renk sinyali olmak üzere iki tip sinyalin bileşiminden oluşan S-Video standardında ise senkronizasyon sinyali parlaklık sinyalinin bir parçasıdır. Sayısal HDMI, DVI ve Analog VGA video arayüzlerinde ise renk sinyalleri ve senkronizasyon sinyalleri birbirlerinden ayrı

taşınmaktadır .

RGB renk uzayı, ingilizcedeki 'Red' 'Green' 'Blue' ('Kırmızı' 'Yeşil' 'Mavi') kelimelerinin baş harflerinden ismini alan ve en yaygın olarak kullanılan bir renk uzayıdır.

Işığı temel alarak, doğadaki tüm renklerin kodları bu üç temel renge referansla belirtilmektedir. Her renk %100 oranında karıştırıldığında beyaz ve %0 oranında karıştırıldığındaysa siyah elde edilir. Bu uzayda, ana renkler olan kırmızı, mavi ve yeşil belirtilmediği için, bu ana renklerin tanımı değiştikçe, tüm renkler değişir. Renklerin oranlarla birlikte değişimi Şekil 2.1.1-3'de gösterilmektedir.

İnternette kullanılan renk sistemi RGB renk sistemidir. Bunun sebebi, 1953'te ilk renkli fotoğraf makinesi olan Polaroid'te ve ondan sonra da televizyonlarda standart kabul edilmiş olmasıdır. Günümüzde tüplü ekranlarda, tarayıcılarda, televizyon ve manuel fotoğraf makinelerinde standart olarak kullanılmaktadır.



Şekil 2.1.1-3 RGB uzayına göre renklerin oluşturulması

2.1.2. Video Arayüz Standartları

Bu bölümde yaygın olarak kullanılan bazı sayısal ve analog video standartlarından bahsedilmektedir.

S-VIDEO

S-Video, video bilgisini iki ayrı sinyal (parlaklık ve renk) şeklinde taşıyan bir video sinyalidir. Y (parlaklık bilgisi)/ C (renk bilgisi) olarak da bilinir. S-Video ile ses sinyali taşınmaz. S- Video, projeksiyon için uygun değildir. VGA'dan çok düşük kalitede görüntü verir. S-Video girişi, genelde eski bilgisayarlarda bulunur. S-Video, komponent video standardında yer almaktadır [9].

NTSC & PAL

NTSC (Ulusal Televizyon Standartları Komitesi), dünyanın birçok yerinde kullanılan bir renk kodlama sistemidir. RCA ve NBC firmaları tarafından geliştirilen formatın, daha sonraları birçok farklı versiyonu geliştirilmiştir. NTSC sistemlerinin ortak özelliği 525 yatay çizgi (45'i boş hat), 30 görüntü/saniye yapısında olmasıdır.

PAL (Faz Değişkenli Hat), dünyanın en çok kullanılan renk kodlama sistemidir. PAL, analog bir format olup televizyon yayın sistemlerinde kullanılır. Birçok farklı versiyonu bulunan PAL'ın ortak özelliği 625 satır ve 25 görüntü/saniye olmasıdır. NTSC ve PAL kompozit video standardındadır.

VGA

Bilgisayar donanımı ve bazı tüketici ekipmanları tarafından kullanılan VGA arayüzü analog RGB sinyalleri aktarmak içindir. Analog RGB sinyalleri senkronize bilgileri içerir.

DVI

"Sayısal Görüntü Arayüzü (DVI)", LCD ve sayısal projektörler gibi cihazlardaki görüntü kalitesinin artırılması için tasarlanmış bir standarttır [27]. Sıkıştırılmamış sayısal video verisinin taşınmasını amaçlamaktadır. HDMI (Yüksek Tanımlı Çoklu Ortam Arayüzü) ile kısmen uyumludur.

DVI, aynı zamanda analog ve sayısal monitörlerin tek bir bağlayıcı vasıtasıyla

kullanılabilmesi için geliştirilmiş bir spesifikasyondur. 3 farklı çeşidi vardır:

- DVI-A (Analog) – Analog sinyaller için tasarlanmıştır.
- DVI-D (Sayısal) – Sayısal sinyaller için tasarlanmıştır. DVI-D, analog arabirimlere göre daha hızlıdır. Analog bağlantıda titreşim olurken DVI-D'de olmaz. DVI-D, düşük sinyal değişimlerinden kaynaklanan kaybı ve sinyaldeki gürültüyü azaltmaktadır.
- DVI-I (Entegre) – Hem analog hem de sayısal monitörler için tasarlanmıştır.

Bazı yüksek çözünürlüklü ekranların DVI bağlayıcılarında yüksek görüntü kalitesinin sağlanabilmesi amacıyla ek olarak ikinci bir veri bağlantısıyer almaktadır. Bu tip bağlayıcılar, "çiftli bağlantılı DVI-D bağlayıcıları" olarak adlandırılmaktadırlar. Şekil A.4-2'de DVI bağlayıcı tipleri gösterilmektedir.

Bir çok LCD monitör sayısal arabirimi kullanır ve DVI-I veya DVI-D portlarına bağlanarak ekran kartından görüntü aktarılır. Sayısal arabirim kullanıldığında herhangi bir sinyal dönüştürme işlemi yapılmaz. Günümüz ekran kartlarının çoğu, standart 15-pin'li VGA portuyla birlikte DVI bağlantı noktası da taşımaktadır.

HDMI

HDMI ya da tam adıyla High Definition Multimedia Interface *(*Yüksek Tanımlı Çoklu Ortam Arayüzü*)* ses ve görüntü verilerini sıkıştırılmadan dijital olarak aktarmak için 2003 yılında geliştirilmiş bir arabirimdir. HDMI; blu-ray disk çalar, HD-DVD disk çalar, bilgisayar, oyun konsolu, dijital uydu alıcısı gibi cihazlarla uyumlu ses ve görüntü cihazlarını (LCD televizyon) bağlar. 2006 yılında HDTV kameralar ve dijital fotoğraf makinelerinde de görülmeye başlamıştır.

RF (Radyo Frekansı) koaksiyel kablo, kompozit video, komponent video, S-Video, SCART, VGA gibi analog görüntü arayüz standartları ile DVI gibi dijital arayüz standartlarına alternatif olarak telif haklarını güvence altına almayı amaçlayan ve erişim kontrolü sağlayan DRM (Sayısal Haklar Yönetimi) teknolojisini beraberinde getirmektedir. DVI ile geri uyumlu olup, ek olarak ses de taşımaktadır.

Standart ile beraber protokol, elektriksel karakteristik, kablo ve konnektör için mekanik ve elektriksel özellikler tanımlanmıştır. HDCP (Yüksek Bant Genişliğinde

Sayısal İçerik Koruma) uyumluluğu HDMI standardına uyumluluğun bir parçası olarak tanımlanmıştır. 4 çeşit HDMI sürümü vardır. Bunlar; HDMI 1.0, HDMI 1.1, HDMI 1.2, HDMI 1.3'dür.

2.2. Anahtarlı Özet Fonksiyonu Tabanlı Mesaj Doğrulama

Özet fonksiyonlarını temel alan mesaj doğrulama yöntemine Anahtarlı Özet Fonksiyonu Tabanlı Mesaj Doğrulama Şifresi (HMAC (FIPS PUBS 198)) denilmiştir [2]. Federal Bilgi İşleme Standart Yayınları (FIPS PUBS 198) olarak Amerikan Ulusal Teknoloji ve Standartlar Enstitüsü (NIST) tarafından standart olarak yayınlanmıştır. Bu standart mesaj doğruluğunu onaylamak için geliştirilmiş bir algoritmadır.

Anahtarlı Özet Mesaj Doğrulama Şifresi Standardı, Bilgisayar Güvenlik Standardı kategorisinde yer almaktadır. Standardın alt kategorisi kriptolojidir. Ulusal Teknoloji ve Standartlar Enstitüsü'ne ait Bilgi Teknolojileri Laboratuarı (ITL) tarafından kabul edilmiştir.

MAC'lerin genel amacı; mesajın kendisini ve bütünlüğünü herhangi başka bir mekanizmaya ihtiyaç duymadan doğrulayabilmektir. HMAC'lerin en belirgin fonksiyonel parametreleri mesaj, mesajı şifreleyecek ve mesajın şifresini çözecek tarafın bildiği ortak anahtardır.

HMAC fonksiyonunda gönderici taraf anahtarı ve şifrelenecek metini kullanarak şifreyi oluşturur. Oluşturulan şifre metin ile birlikte gönderilir. Alıcı taraf gelen mesajı anahtarla şifreler. Gelen şifreli metni kendi elde ettiği şifrelenmiş metin ile karşılaştırıp doğrulama işlemi gerçekleştirir. Eğer karşılaştırma sonucu her iki şifrelenmiş veri aynı ise mesaj doğrulanmış demektir. Gönderici taraf istediği sayıda alıcı tarafla şifresini paylaşabilir.

Taşınan bilginin bütünlüğünü sağlamak, günümüz iletişim dünyasında başlıca bir zorunluluktur. HMAC uygulamasında mesajın bir bitinde bile değişim olsa elde edilecek şifrelenmiş metin farklı olacaktır.

Bütünlük kontrollerini sağlayan gizli bir anahtarla şifreleme işlemi gerçekleştiren mekanizmalar mesaj belgeleme şifreleridir (MACs). Tipik olarak, mesaj belgeleme kodları; bilginin, belirlenen kullanıcılar arasında paylaşıldığını belgelemek amacıyla

kullanılır. Bu standart, genellikle gizli anahtarla birlikte kullanılan kriptografik özet fonksiyonu olarak tanımlanır.

HMAC kullanım amaçları :

- Sık kullanılan bir uygulamadır. Ayrıca kullanılan fonksiyonlar yüksek performanslıdır.
- Önemli bir performans kaybı olmadan etkili bir güvenilirlik sağlar.
- Anahtar kullanımı kolaydır.
- Özet fonksiyonları gücü nedeniyle mesaj doğrulama yöntemlerinde kabul görmüştür.
- Kullanılan özet algoritması, daha güvenli ve hızlı bir versiyonunun çıkması halinde yenisiyle kolayca değiştirilebilir.

Bu standartta kullanılan terimlerin açıklamaları aşağıda yer almaktadır.

Kriptografik anahtar: Kriptografik algoritma ile kullanılan ve bu algoritmayı uygulama için özel hale getiren parametredir. Bu uygulamada anahtar HMAC algoritması tarafından kullanılmaktadır ve bilgi doğrulamasını sağlamaktadır.

Anahtarlı-Özet Fonksiyonu Tabanlı Mesaj Doğrulama Şifresi (HMAC): Bilgiyi doğrulamak için anahtarlı kullanılan özet fonksiyonudur.

Mesaj Doğrulama Şifresi (MAC): Bilgi doğrulama algoritmasından geçtikten sonra oluşan özet bilgidir. Bu standart HMAC, HMAC sonucunda elde edilen özet bilgi ise MAC'dir.

Gizli Anahtar: Kriptografik tek anahtar, bir ya da birden çok kuruluş arasında kullanılır. Bu kullanım şekli tam olarak "gizli" kavramını taşımamaktadır.

Özet fonksiyonu: Sabit mesaj bloklarından oluşan herhangi uzunluktaki bilginin onaylı bir matematiksel algoritmayla haritalanmasıdır. Bu algoritma sayesinde çok uzun bir metinden özet oluşturulmuş olur. Bölüm 2.2.1'de daha detaylı olarak anlatılacaktır.

2.2.1. Özet Fonksiyonu

Büyük miktarda verinin bütünlüğünün ve doğrulanmasının az miktarda veri ile yapılabilmesi için özet fonksiyonu kullanılır.

Özet değeri h, H fonksiyonu kullanılarak h = H(M) formatında elde edilir. Burada M, herhangi bir boyuttaki mesaj, H(M) ise sabit boyuttaki mesaj özetidir. Hesaplanan ve doğru olduğu bilinen özet ilgili mesajın sonuna eklenerek gönderilir. Alıcı taraf mesajı doğrulamak için tekrar özeti hesaplar. Çünkü kullanılan özet fonksiyonu gizli olmayabilir. Bu nedenle özet değerini korumak gerekmektedir.

Özet fonksiyonları mesaj doğrulama (message authentication) için kullanılır [1].

Özet fonksiyonlarının asıl amacı dosyanın, mesajın ya da blok halindeki verinin parmak izini oluşturmaktır. Özet fonksiyonunun gereksinimleri aşağıdaki sıralanabilir.

1. Özet fonksiyonu herhangi bir boyuttaki blok şeklindeki veriye uygulanabilir.
2. Özet fonksiyonları sonucunda sabit uzunlukta mesaj özeti oluşur.
3. H(x) her x için hesaplaması kolaydır. Yazılım ve donanım gerçekleştirmesi yapılabilir.
4. Herhangi bir h için, geri dönüşümlü H(x) = h hesaplaması yapılamaz. Bu duruma tek-yön özelliği denir.
5. Herhangi bir x bloğu için, H(x) = H(y) ise $x \neq y$ hesaplaması bulunamaz. Bu özelliğe zayıf çarpışma direnci denir.
6. Çift x ve y bloğunu H(x) = H(y) olarak hesaplamak mümkün değildir. Bu durum güçlü çarpışma direnci olarak adlandırılır.

İlk üç özellik pratik oluşu nedeniyle mesaj doğrulamada kullanılır. Dördüncü özellikten anlaşılan durum ise mesajdan özet oluşturmanın kolay olması ama özetten özet fonksiyonunu tekrar kullanılarak mesajı elde etmenin mümkün olmamasıdır. Beşinci özellik alternatif mesajın aynı özet fonksiyonundan geçirilmesiyle asıl mesaja ulaşılamayacağı garantilemektedir. Altıncı özellik ise özet fonksiyonunun saldırılara karşı olan direnci hakkında bilgi vermektedir.

Özet fonksiyonun kullanıldığı yerlerden bazıları aşağıda listelenmiştir.

- Şifre doğrulama
- Mesaj doğrulama şifrelerinin oluşturulması
- Sayısal imzalar
- Bir bilginin duyurulmadan doğrulanması
- Verilen bir elemanın bir kümeye ait olup olmadığının sabit zamanlı testi
- Olasılıksal veri yapısı-Bloom testi.

Yinelemeli özet fonksiyonu olarak MD5, SHA-1, SHA-512... kullanılabilir.

2.2.2. SHA-1

Standardın adı, Güvenli Özet İmzadır (FIPS PUB 180-2). Güvenli Özet İmza Standardı, Bilgisayar Güvenlik standardı kategorisinde yer almaktadır. Standardın alt kategorisi kriptolojidir [4].

Bu standart dört güvenli özet algoritması (SHA-1, SHA-256, SHA-384 ve SHA-512) ile belirtilen elektronik veri (mesaj) özeti elde etmek için kullanılır. Herhangi bir uzunlukta mesaj geldiğinde seçilen algoritma sonucunda sabit uzunlukta bir mesaj özeti oluşturulur. Mesaj özet aralığı seçilen algoritmaya bağlı olarak 160-512 bit arasında değişir. Özet algoritmaları genellikle sayısal imza ve anahtarlı-karma ileti kimlik doğrulama kodları veya rastgele sayı dizinleri gibi kriptografik algoritmalar ile kullanılır.

Mesaj özetinden mesajı elde edememe ve aynı mesaj özetine sahip iki farklı mesaj olma durumu imkansız olarak hesaplandığı için bu dört özet fonksiyonu güvenilir kabul edilmiştir.

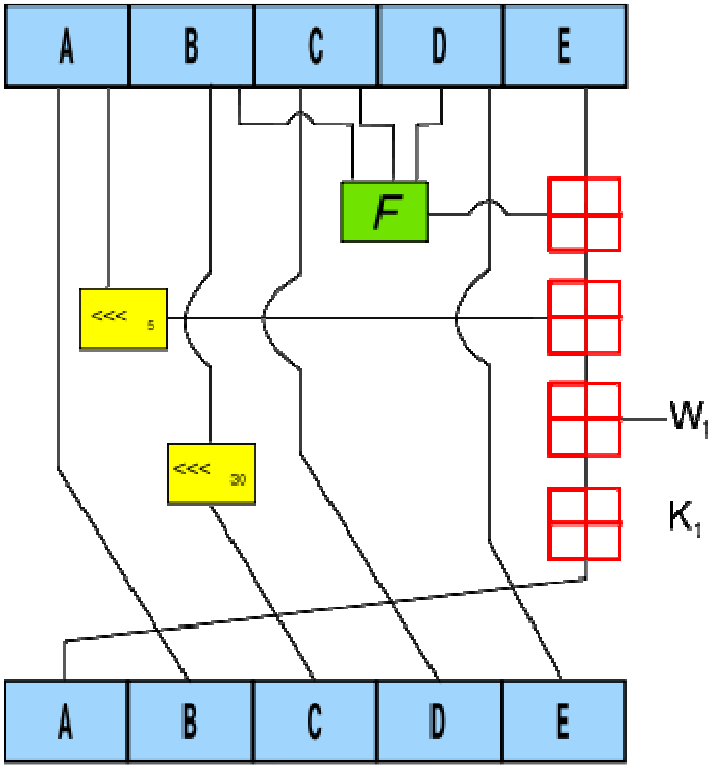
SHA-1 yapısını çalıştırabilmek için yapılması gereken adımlar vardır. SHA-1 blokları 512-bit'ten oluşur. İlk aşamada 512-bit olmayan mesajı SHA-1 bloğu haline getirmek için dolgulama işlemi yapılır. Dolgulama işleminde mesajın sonuna '1' eklenir. Son 64-bit mesaj uzunluk bilgisidir. Aradaki bitlerin tümü mesaj uzunluğu 512'nin katı olana kadar sıfır ile doldurulur.

Mesaj özetinin hesaplanmasından önce başlangıç değeri verilir. H0, H1, H2, H3, H4 32-bit'lik özet yazmaçlarıdır.

H0 = 67452301
H1 = efcdab89
H2 = 98badcfe
H3 = 10325476
H4 = c3d2e1f0 onaltılık sayı tabanında ifade edilmiş başlangıç değerleridir.

Sonrasında dolgulama işlemi yapılan mesaja 512-bit'lik bloklar halinde SHA-1 işlemi uygulanır. Algoritma 16 adrese yazılan 32-bit'lik verileri kullanarak toplam 80 adet 32-bit'lik veri oluşturulmaktadır [5].

$$W_t = (W_{t-3} \oplus W_{t-8} \oplus W_{t-14} \oplus W_{t-16}) <<< 1 \qquad W_t \geq 16 \qquad (1)$$



Şekil 2.2.2-1 SHA-1 çalışma Yapısı

A, B, C, D ve E isminde 5 adet yazmaç tanımlanır. Güncellemeler ve hesaplamalar bu yazmaçlarla gerçekleştirilir. Başlangıçta bu yazmaçlardan A'ya H0, B'ye H1, C'ye H2, D'ye H3, E'ye H4 değerleri atanır.

80 adet 32-bit'lik veri oluşturduktan sonra belli katsayılar ve farklı hesaplamalar uygulanarak 160-bit'lik mesaj özeti elde edilir. Aralıklara ait katsayılar aşağıdaki gibidir [11].

$$K_t = \begin{cases} x"5a827999" & 0 \le t \le 19 \\ x"6ed9eba1" & 20 \le t \le 39 \\ x"8\ f\ 1bbcdc" & 40 \le t \le 59 \\ x"ca62c1d6" & 50 \le t \le 79 \end{cases} \quad (2)$$

Kullanılan fonksiyon t değerine göre değişmektedir [6].

$$F(X,Y,Z) = \begin{cases} (X \wedge Y) \oplus (\neg X \wedge Z) & 0 \le t \le 19 \\ X \oplus Y \oplus Z & 20 \le t \le 39 \\ (X \wedge Y) \oplus (X \wedge Z) \oplus (Y \wedge Z) & 40 \le t \le 59 \\ X \oplus Y \oplus Z & 60 \le t \le 79 \end{cases} \quad (3)$$

Her t değişiminde aşağıdaki atamalar gerçekleştirilir.

$T = (A<<<5) + F(B,C,D) + W_t + K_t + E$

A ←T

B ←A

C ←B <<<30

D ←C

E ←D

80 adım gerçekleştirildikten sonra yeni mesaj özeti aşağıdaki gibi hesaplanır.

H0 ← H0 + A

H1 ← H1 + B

H2 ← H2 + C

H3 ← H3 + D

H4 ← H4 + E

Şekil 2.2.2-1' de SHA-1 bloğunun çalışma yapısı gösterilmiştir.

Kullanılan Semboller:

$\wedge$: Ve kapısı

$\vee$: Veya kapısı

$\neg$: Tümleme operasyonu

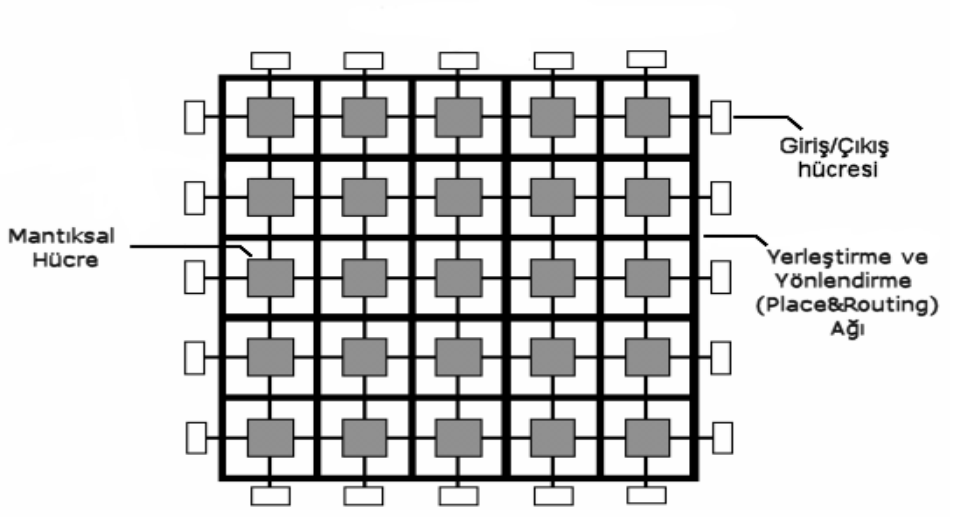
<<< : Sola döndürme işlemi

$\oplus$: XOR kapısı

2.3. FPGA'lere Genel Bakış

FPGA Tanımı ve Genel özellikleri

FPGA (Alanda Programlanabilir Kapı Dizileri), programlanabilir mantık blokları ve bu bloklar arasındaki ara bağlantılardan oluşan ve geniş uygulama alanlarına sahip olan sayısal tümleşik devrelerdir. Tasarımcının ihtiyaç duyduğu mantık işlevlerini gerçekleştirme amacına yönelik olarak üretilmiştir. Dolayısıyla her bir mantık bloğunun işlevi kullanıcı tarafından düzenlenebilmektedir. FPGA kullanılarak temel mantık kapılarının ve yapısı daha karmaşık olan devre elemanlarının işlevselliği arttırılmaktadır. Alanda programlanabilir ismi verilmesinin nedeni, mantık bloklarının ve ara bağlantıların imalat sürecinden sonra programlanabilmesidir.



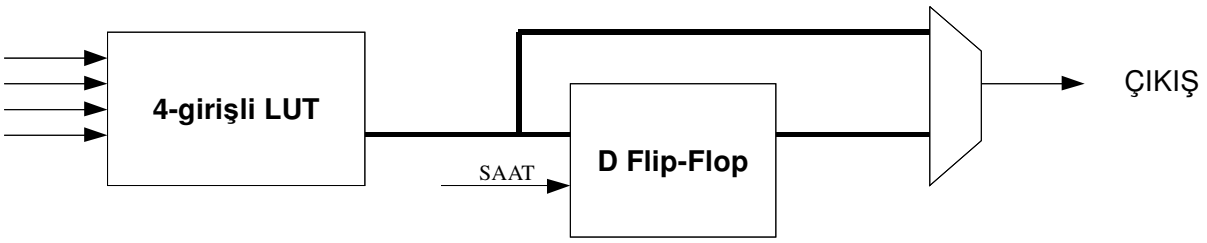
Şekil 2.3-1 FPGA Yapısı[1]

[1] http://www.eecg.toronto.edu/~vaughn/challenge/fpga_arch.html

Şekil 2.3-1'de görüldüğü gibi FPGA, programlanabilir mantık blokları, bu blok dizisini çevreleyen giriş-çıkış blokları ve ara bağlantılar olmak üzere üç ana bölümden oluşur. Programlanabilir mantık blokları, ara bağlantılar içerisine gömülü şekilde bulunur. Programlanabilir mantık bloklarının yapılandırılması ve bu bloklar arasındaki iletişim ara bağlantılar sayesinde gerçekleşir. Giriş çıkış blokları, ara bağlantılar ile bütünleşmiş devrenin paket bacakları arasındaki ilişkiyi sağlar.

Tipik FPGA Mantık Bloğu Yapısı

Tipik FPGA mantık bloğu, Şekil 2.3-2'de gösterildiği gibi 4 girişli LUT yapısı ve flip-flop gibi diğer mantık elemanlarından oluşur.



Şekil 2.3-2 FPGA Mantık Bloğunun Yapısı

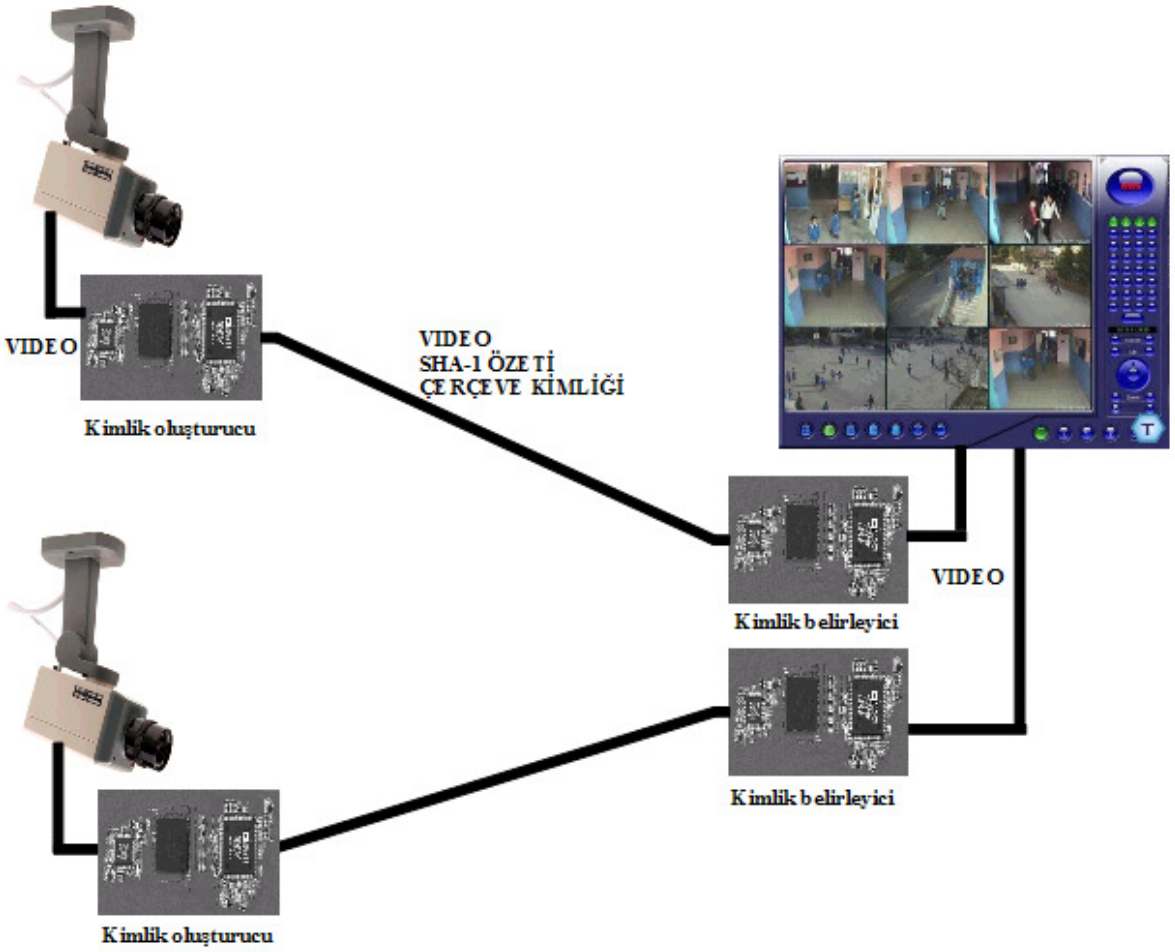
4 girişli LUT yapısı, değişik mantıksal işlemleri yürütür. Çıkış verileri isteğe bağlı olarak yazmaçta saklanır.

Uygulama Alanları

Düşük maliyetli olması ve tasarım sırasında kullanıcıya esneklik sağlaması sebebiyle FPGA kullanımı gittikçe yaygınlaşmıştır. Savunma, sayısal sinyal işleme, uzay, tıbbi görüntüleme ve otomotiv, FPGA'nın uygulama alanlarından bazılarıdır. FPGA'ler özellikle paralel işlem gerektiren uygulamalarda yaygın olarak kullanılmaktadır.

3. SİSTEM YAPISI

Bu kısımda, giriş kısmında bahsedilen senaryolara uygun olarak oluşturulan sistemin daha ayrıntılı bir şekilde anlatımı yapılmaktadır. Sistemde iki adet kart bulunmaktadır. Aslında bahsedilen iki kart donanım olarak birbirinin aynısıdır. Kartları farklı kılan içerlerinde yer alan HDL kodlarıdır. Bu kartlardan birisi kamera tarafında yer alır. Sonraki çalışmalarda kartın kamera ile tümleşik hale getirilmesi amaçlanmaktadır. Diğer kart ise monitör tarafında bulunur. Monitör tarafındaki kart her zaman modüler halde kullanılacaktır. Kamera tarafında yer alan kart, kamerayı tanımlayan yapıda olduğu için "Kimlik Oluşturucu" olarak adlandırılmıştır. Monitör tarafındaki kart ise gelen görüntünün doğruluğunu tespit etmesi sebebiyle "Kimlik Belirleyici" olarak adlandırılmaktadır. Bundan sonraki kısımlarda kartlar için bu isimler kullanılmaktadır. Sisteme ait yapı Şekil 3-1'de gösterilmektedir.

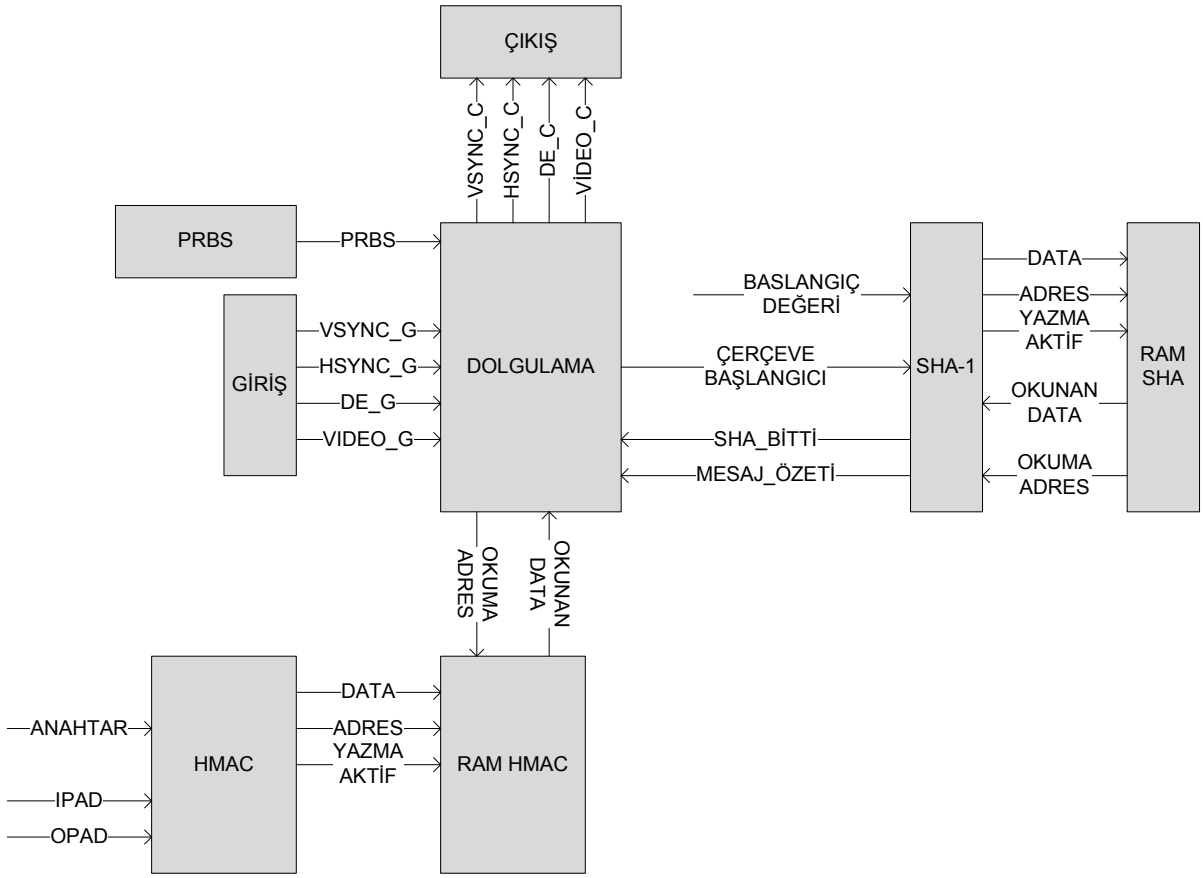


Şekil 3-1 Video kartı kurulum

Tasarımın sistem yapısı, görüntünün kameradan alınmasından monitöre ulaştırılmasına kadar olan sıralamayla anlatılacaktır.

Öncelikle kameradan alınan görüntü yüksek frekansta gelen differansiyel videodur. Gelen görüntü ilk olarak kimlik oluşturucuya girmektedir. Kimlik oluşturucu üzerinde yer alan DVI çözücünün çıkışı 24-bit RGB şeklindedir. 24-bit görüntü bilgisinin renk içeriği 8-bit kırmızı (R), yeşil (G) ve maviden (B) oluşmaktadır. Video ile ilgili bütün sinyaller (veri, senkronizasyon, benek frekansı,…) kart üzerinde yer alan FPGA'in bacaklarına bağlanmıştır. FPGA içerisinde çerçeve sayma, PRBS kullanılarak özet oluşturma ve gerçek zamanlı videoyu kullanılarak özet oluşturma [23] işlemleri gerçekleştirilmektedir. İşlemlerin tümü gerçek zamanlı olarak gerçekleştirilmekte ve herhangi harici bir belleğe ihtiyaç duyulmamasıyla yapılan birçok çalışmadan daha farklı bir yerde durmaktadır. Hesaplanan bütün değerler videonun önceden belirlenen satırlarına eklenerek DVI şifreleyiciye gönderilmektedir. DVI şifreleyici, hesaplanan değerleri de içeren sayısal video verilerini differansiyel hale getirerek göndermektedir. Belirlenen sistemde video verisi DVI kablosu üzerinden taşınmaktadır. Kimlik oluşturucu, 3.1 Kimlik Oluşturucu başlığı altında detaylandırılmıştır. Kimlik belirleyici kısmında yer alan DVI çözücünün çalışma prensibi de aynıdır. Fakat FPGA içerisine yüklenen HDL kodların çalışma yapısı farklıdır. Öncelikle kimlik oluşturucu kısmında video verisine eklenen hesaplamaların hangi satırlarda yer aldığı kimlik belirleyicide tanımlıdır. Gelen veriler, karşılaştırma yapabilmek için bir süreliğine ilgili yazmaçlara kaydedilir. FPGA içerisinde özet oluşturma ve çerçeve sayma işlemleri gerçekleştirilir. Yazmaçlara kaydedilen değerlerle hesaplanan değerler karşılaştırılır. Özetlerin ve çerçeve sayacının birbirinin aynı olması durumunda sistemde problem olmadığına karar verilir. Görüntü monitöre olduğu gibi basılır. Karşılaştırma sonucunda hatalı bir durum olduğuna karar verilirse inceleyen kişiyi bilgilendirmek amacıyla ekrana X basılır. Kimlik oluşturucu ve kimlik belirleyici kısımlarına ait FPGA'ler genel çalışma yapısı açısından birbirine oldukça benzemektedir. Kimlik oluşturma işleminde her video çerçevesi için 16-bit çerçeve sıra numarası atanır. Her video çerçevesi gönderiminde çerçeve sıra numarası bir artmaktadır. Ayrıca çerçeve sıra numarası ile bir adet SHA-1 bloğu oluşturabilmek için 503-bit sözde rastgele ikili dizi (PRBS) üretilip 496-bit kısmı alınır [16]. Kimlik oluşturucu kısımda hesaplanan PRBS değerine ait SHA-1 bloğu, oluşan mesaj özeti ile birleştirilip bir satırda 672-bit olarak kimlik belirleyiciye gönderilmektedir.

3.1. Kimlik Oluşturucu



Şekil 3.1-1 Kimlik oluşturucu yapısı

Şekil 3.1-1'de kimlik oluşturucu yapısına ait blok çizim verilmiştir. Aslinda bu kısım kimlik oluşturucu ve kimlik belirleyici için ortak olan çok sayıda blok içermektedir. Ortak olan blokların açıklaması kimlik oluşturucu kısmında gerçekleştirilecektir.

Yeni bir çerçevenin başlangınıcından itibaren değerlendirirsek sistemin başlangıç noktası giriş bloğudur. Bu bloğun DVI çözücüden çıkıp FPGA'e bağlanan video verileri ile doğrudan bağlantısı bulunmaktadır. Gelen video üzerinde yapılan bütün hesaplamaların uyumlu çalışması dolgulama bloğu tarafından gerçeklenir. Dolgulama bloğunda gelen video verileriden 512-bit' lik SHA-1 bloğu oluşturmak için her satırdan 512-bit alınmaktadır. Tasarım bloğunda IPAD (içsel dolgulama) ve OPAD (dışsal dolgulama) hesaplamalarının yapılması takip edilmektedir. Özet mesajının oluşturulabilmesi için mesaj bloklarından önce IPAD için SHA-1 hesaplaması, sonrasında ise OPAD hesaplaması gerçekleştirilmesi gerekmektedir. Hesaplamaların sadece kimlik oluşturucu ve kimlik belirleyici kısım arasında kalmasını istediğimiz için

sistemde her iki taraf için ortak olan 64-bit'lik anahtar kullanılmaktadır. Bu anahtar, tekrarlama işlemi yapılarak birer adet SHA-1 bloğu haline getirilen IPAD ve OPAD değerleriyle XORlanır. Bu işlem HMAC bloğu içerisinde gerçekleştirilir. Hesaplanan değerler ram_hmac belleğine kaydedilir. SHA-1 uygulaması kullanılacağı zaman dolgulama bloğu tarafından ilgili bellekten okunarak özet oluşturulur.

PRBS bloğu gelen her yeni video çerçevesine sıra numarası ve rastgele sayı verip çerçeve bilgisinden özet bilgi çıkarması için SHA-1 bloğuna bellek üzerinden iletir. Özet bilgilerin FPGA içine gömülü gizli bir anahtar kullanılarak çıkarılmasını HMAC bloğu sağlar. PRBS üreteci ve çerçeve sayacı ile üretilen yeni mesaj için farklı anahtar kullanılarak farklı bir özet oluşturulur. Elde edilen iki özet, gelen video çerçevesinin görünmeyen satırlarına video çıkış bloğunca yerleştirilir.

Video çerçevesinin görünen bölgesinin belirlenen satırlarından p benekte bir alınan toplam 512-bit veri SHA-1 döngüsünde bir blokluk giriş verisine denk gelir [3]. Görüntü bütünlüğü işlemlerinde video çözünürlüğüne bağlı olarak p değişir:

$$p = d * b / 512 \quad (4)$$

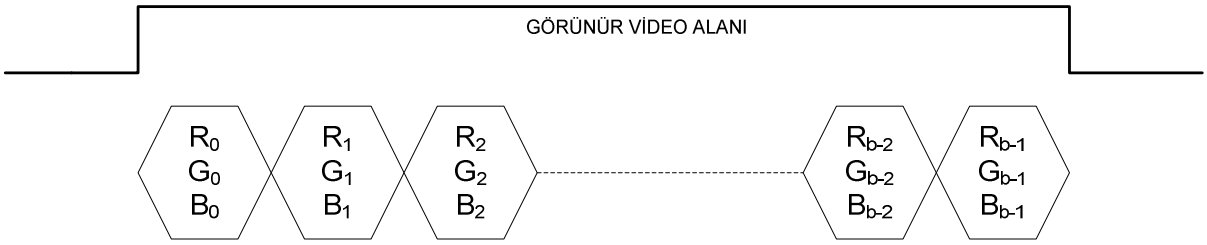
p; satırın görünen bölgesinde atlanacak benek sayısını

d; beneğin G (yeşil renk) verisi için bit sayısını ve

b; satırın görünen bölgesinde benek sayısını

ifade etmektedir.

Yukarıdaki (4) eşitliğini kullanan video giriş (GIRIS*)* bloğu görünen bölgenin verilerini toplayıp SHA-1 belleğine yazar ve 512-bit'lik veri bloğunu SHA-1 döngüsü için oluşturur. Bu işlem, görünen bölgede satır sayısı s kere tekrarlanır ve sonucunda 160-bit görüntü özeti ortaya çıkar. Belleğe yazma ve okuma sıklığı video benek saat sıklığı ile doğru, p değeriyle ters orantılıdır. Giriş bloğu için p, d ve s değişken girişlerdir. Şekil 3.1-2'de yer alan video verisinin taşınma yapısını inceleyerek SHA-1 bloğunda kullanılacak verilerin nasıl elde edildiğini tekrar inceleyebiliriz.



Şekil 3.1-2 Video veri yapısı

Her benekten sadece yeşil renk bilgisi taşıyanlar toplanacaktır. Atlanacak benek sayısı 1 olursa (p=1) G_0, G_2, G_4….., $G12_6$ bitleri alınacaktır.

Kimlik bilgisini her çerçeve için oluşturan video çerçeve sıra numarası verisi ve buna bitiştirilen sözde rastgele ikili dizi (PRBS) verisi toplamı olan 512-bit veri SHA-1 döngüsünde bir blokluk veridir.

$$k = ç \,||\, r \qquad (5)$$

k: Kimlik belirleyici kod (512-bit'lik veri)

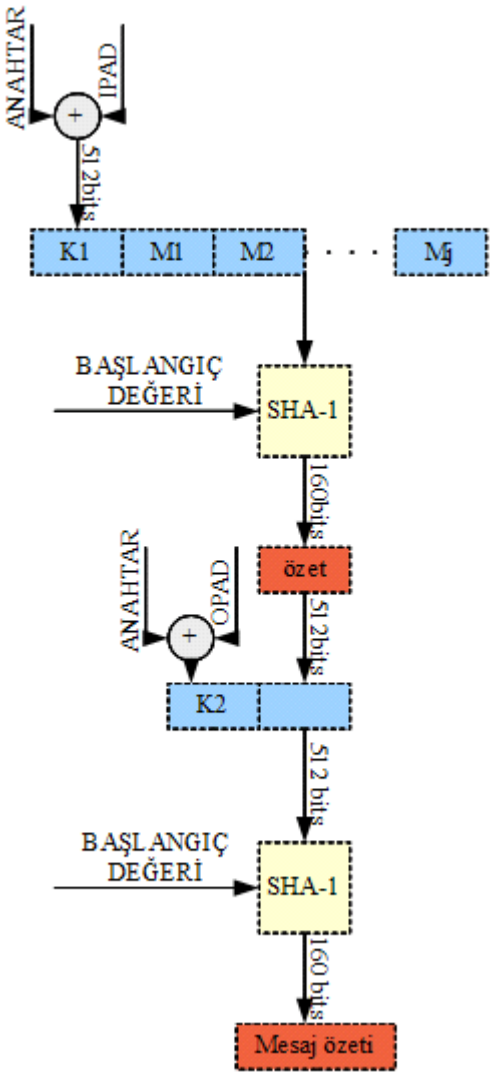
ç: Video çerçeve sayısı (16-bit'lik veri)

r: Rastgele sayı (496-bit'lik veri)

||: Bitiştirme işlemi

Yukarıdaki (5) eşitliğini PRBS giriş (PRBS_URETECI*)* bloğu tarafından gerçekleştirilir. Blok içinde çerçeve sayısı için doğrusal 16-bit sayaç ve "linear feedback shift register (LFSR)" ile gerçekleştirilmiş 503-bit PRBS üreteci bulunur. 16-bit uzunluğundaki çerçeve sıra numarası ve 503-bit'lik PRBS sonucunun en değerli 496-biti kullanılarak bir adet SHA-1 bloğu oluşturulur.

Yukarıda tanımlanan GIRIS ve PRBS_URETECI bloklarının oluşturduğu 512-bit veri blokları 64-bit uzunluğunda gizli anahtar değeri ile HMAC ve SHA-1 bloklarınca işlenir ve sonucunda iki adet 160-bit HMAC oluşur. Gizli anahtarın kullanımı [5] için işleyiş Şekil 3.1-3'de yer almaktadır.



Şekil 3.1-3 Dolgulama blok yapısı

K1 = ipad64 *xor* (a || sfr56) (6)

K2 = opad64 *xor* (a || sfr56)

K1: İlk oluşturulan anahtar değeri (512-bit'lik veri)

K2: İkinci oluşturulan anahtar değeri (512-bit'lik veri)

Mj: Mesaj blokları (512-bit'lik veri)

a: Gizli anahtar değeri (64-bit'lik veri)

ipad64: 64 kez tekrarlanmış IPAD (36h) değeri

opad64: 64 kez tekrarlanmış OPAD (5Ch) değeri

sfr56: 56-bayt sıfır değeri

dolgu: Mesaj dolgulama işlemi

b : Dolgulama yapılan mesaj özeti

h : Geçici 160-bit “özet” değeri

h_init : Standart başlangıç “özet” değeri

Bu işlemleri HMAC bloğu gizli anahtar a değerini işleyerek gerçekler ve ram_hmac adlı 1024-bit belleğe K1 ve K2 değerlerini yazar. SHA-1 bloğu ram_hmac belleğini K1 ve K2 değerleri için okur ve diğer 512-bit'lik veri blokları ile birlikte işler.

Video aslının doğrulanmasında kullanılacak hmac1 özet değerinin gizli anahtar kullanılarak elde edilmesi için aşağıda işlem sırası verilmiştir:

h1 <= sha-1(K1, hinit)

hj <= sha-1(Mj, h1)

h2 <= sha-1(K2, hinit)

M(j+1) <= dolgu(hj)

hmac1 <= sha-1(M(j+1), h2)

mac <= sha-1(prbs, h)

h2 <= sha-1(K2, hinit)

b <= dolgu(mac)

hmac2 <= sha-1(b, h2)

Her video çerçevesi için toplam satır sayısı (s) kadar veri bloğu okur. İşlem sırası;

h1 <= sha-1(K1, hinit)

mac <= sha-1(video(i), h1)

i <= 1, 2, 3... s için tekrarla.

h2 <= sha-1(K2, hinit)

b <= dolgu(mac)

hmac2 <= Sha-1(b, h2)

olarak belirlenmiştir.

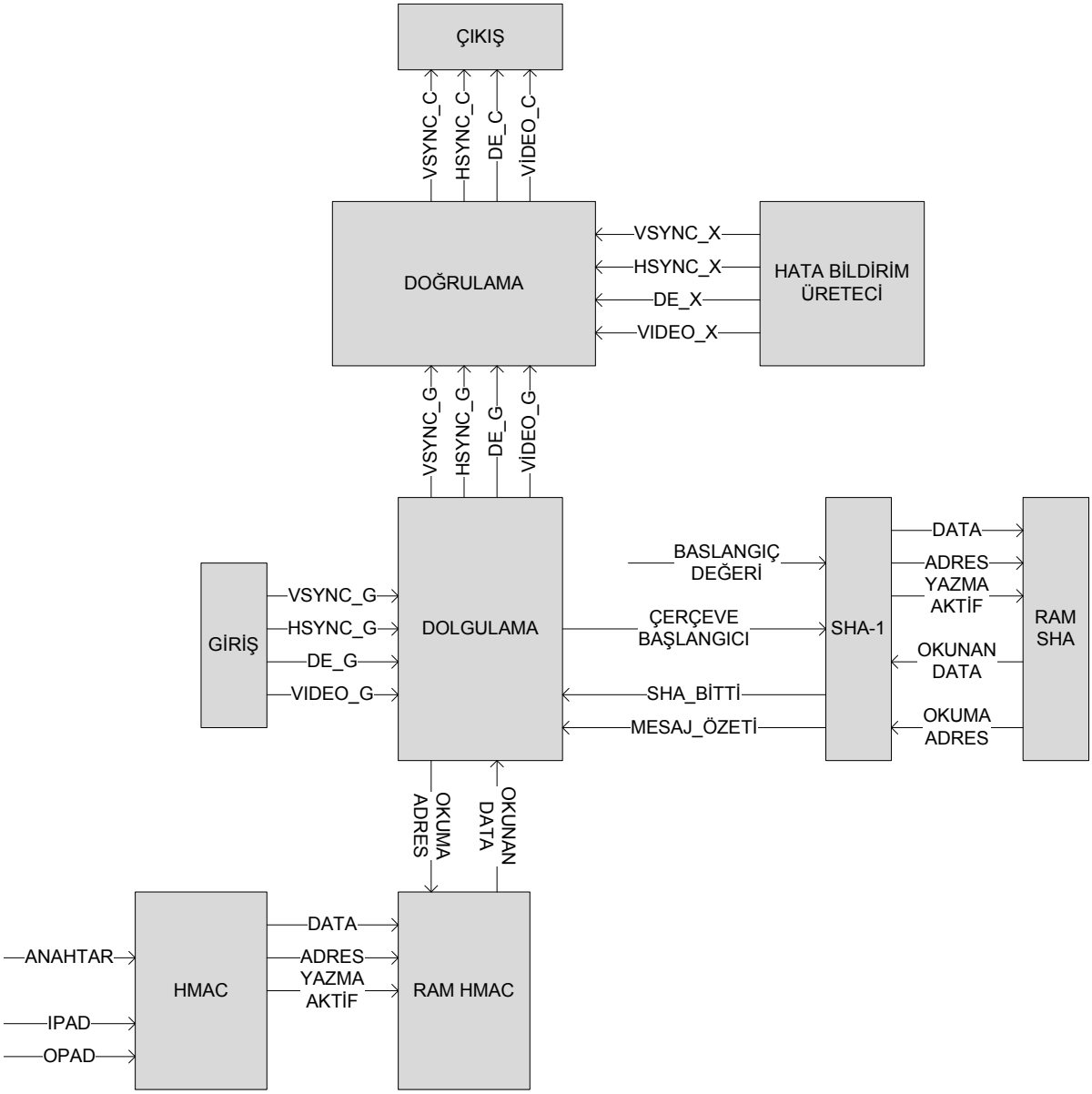
SHA-1 bloğu video ve prbs belleklerinden okunan 16 adet 32-bit sözcükten 80 adet 32-bit sözcük elde eder. 17'nci ve daha sonraki sözcükler kendisinden önce 16'ncı, 14'ncü, 8'inci ve 3'üncü sözcüklerin kullanılmasıyla elde edilir. Bu işlemi gerçekleştirmek için 80*32-bit bellek (sha_bllk*)* kullanır. Türetilen her yeni sözcük *sha_bllk* belleğinde adresi bir arttırarak yazılır. SHA-1 bloğu her video çerçevesi için bulduğu iki adet hmac1 ve hmac2 özet değerinin her birini 124 döngü sonunda ortaya çıkarır ve yazmaçlarına kaydeder.

Video çıkış bloğu (CIKIS), PRBS_URETECI bloğundan PRBS değerini, çerçeve sıra numarasını, SHA-1 bloğundan hmac1 ve hmac2 değerlerini alıp çıkacak video çerçevesi içinde görünmeyen video bölgelerine yerleştirir. Blok, d ve p değişkenlerine göre benek adreslerini ve benek değerlerini oluşturur. Kimlik oluşturmada ve görüntü doğrulamada kullanılan veriler için d=8 alınmıştır. Hesaplanan verilerin gönderilmesinde ise 24-bit'in tamamı kullanılmıştır. Buna göre 672-bit'lik kimlik bilgisi için 672 / 24 = 28 benek kullanılmıştır. Görüntü doğrulama için kullanılan 160-bit hmac2 değeri için 7 benek kullanılır. CIKIS bloğu toplam 35 beneği, görünen video bölgesinin son satırından sonraki görünmeyen video bölgesindeki satıra yerleştirir.

3.2. Kimlik Belirleyici

Kimlik belirleyicinin yapısı kimlik oluşturucu yapısıyla benzerlik göstermektedir. Kimlik oluşturucu kısmında açıklanan HMAC uygulaması aynı şekilde gerçekleştirilir. Kimlik belirleme işlevini gerçekleştirmek için gizli anahtarı (a) kullanarak gelen veri bloklarından oluşturduğu hmac1 ve hmac2 özet bilgilerini gelen özet bilgileri ile karşılaştırır ve farklılık durumunda video çerçevesinin görünen bölgesine hata iletisi çıkarır.

Kimlik belirleyicinin yapısı Şekil 3.2-1'de gösterilmiştir. Video giriş (GIRIS) bloğu görünen video bölgesinden 512-bit'lik bilgiyi SHA-1 bloğuna ait belleğe aktarır. Aynı zamanda görünmeyen video bölgesinden PRBS ve çerçeve sayısını okuyup ilgili yazmaca yazar. Aynı bölgeden karşılaştırılacak hmac1 ve hmac2 özet bilgilerini okur ve yazmaçlara kaydeder. Gizli anahtarı kullanıp HMAC bloğu ile oluşturduğu K1 ve K2 değerlerini ram_hmac belleğine kaydeder. SHA-1 bloğu belleklerden verileri okuyup işleyerek hmac1 ve hmac2 değerlerini elde eder. Kimlik belirleme ve görüntü bütünlüğü işlemlerinin son basamağı CIKIS bloğunda gerçekleşir. Blok, GIRIS bloğundan gelen hmac1 ve hmac2 değerlerini SHA-1 bloğundan gelen değerlerle karşılaştırır. Karşılaştırma sonucunda kimlik bilgisinin özeti olan hmac1 değerinde tutarsızlık olması durumunda hata verir. Görüntü bütünlüğünün doğruluğu için kullanılan hmac2 özet bilgisinde tutarsızlık olması durumunda ise görüntü bütünlüğü hatası verir. Sadece kimlik belirleyicide yer alan uygulamayla hata olması durumunda ekrana X basılmaktadır.



Şekil 3.2-1 Kimlik belirleyicinin yapısı

4. VHDL KODLARININ AÇIKLANMASI

Kimlik oluşturucu ve kimlik belirleyicide kullanılan tasarım blokları bu bölümde ayrıntılı olarak açıklanmaktadır. Tüm tasarım bloklarının kodlamasında VHDL kullanılmıştır. Görüntü gerçek zamanlı olarak işlenmiş ve devrenin tamamı donanımsal olarak hazırlanmıştır. Tasarım blokları ALTERA Quartus II kullanıcı arayüzü kullanılarak hazırlanmıştır. ALTERA Quartus II ile ilgili detaylı bilgi Ekler Dizini altında A.1'de verilmektedir.

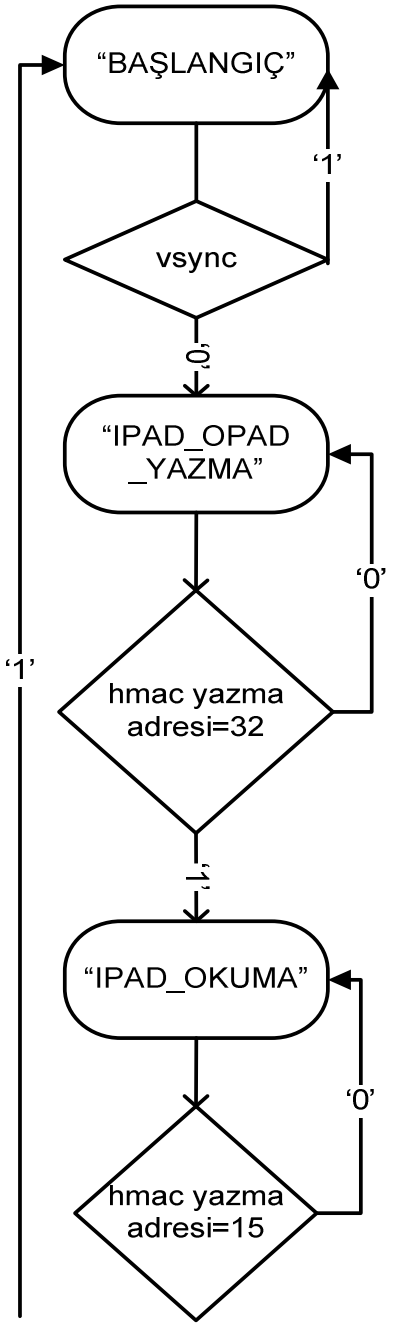
Sırasıyla GIRIS, DOLGULAMA, SHA-1, HMAC, PRBS_URETICI ve CIKIS blokları anlatılacaktır. Tablo 4-1'de FPGA içerisinde bütün blokların birbiriyle bağlantısını sağlayan ana bloğun giriş-çıkış sinyal bilgileri yer almaktadır. Giriş sinyalleri 'G', çıkış sinyalleri 'Ç' ile belirtilmektedir.

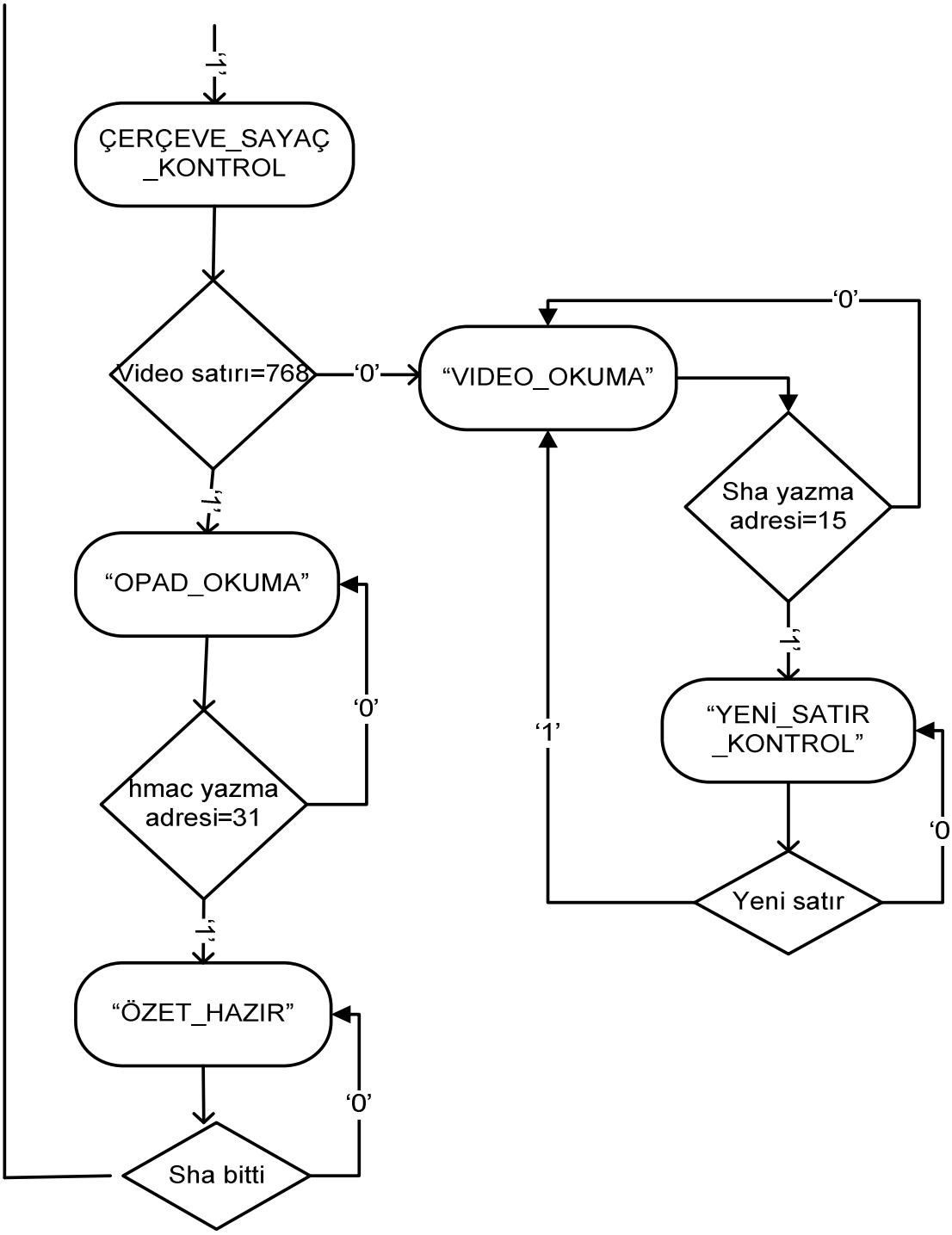
Tablo 4-1 FPGA fiziksel bağlantıları

İsim	G/ Ç	Açıklama
Pxl_clk	G	Bloğa ait çalışma frekansıdır. Bu tasarımda benek saatinin frekansına eşittir. Bütün bloklarda bu frekans kullanılmaktadır.
Rstn	G	Bütün bloklara ait sıfırlama sinyalidir. Aktif düşüktür.
Hsync_in	G	Yatay senkronizasyon sinyalidir. Giriş bloğuna girer.
Vsync_in	G	Düşey senkronizasyon sinyalidir. Giriş bloğuna girer.
De_in	G	Aktif video alanını gösteren sinyaldir. Giriş bloğuna girer.
Video_in	G	24-bit RGB video verisidir. Giriş bloğuna girer.
Hsync_out	Ç	Yatay senkronizasyon sinyalidir. Videonun gönderilebilmesi için gereklidir. Çıkış bloğundan çıkmaktadır.
Vsync_out	Ç	Düşey senkronizasyon sinyalidir. Videonun gönderilebilmesi için gereklidir. Çıkış bloğundan çıkmaktadır.
De_out	Ç	Aktif video alanını gösteren sinyaldir. Çıkış bloğundan çıkmaktadır.
Video_out	Ç	24-bit RGB video verisidir. Çıkış bloğundan çıkmaktadır.

4.1. Dolgulama Bloğu

Dolgulama bloğu özet tabanlı mesaj doğrulama işleminin tümünü kapsayan bloktur. IPAD ve OPAD'lerin ne zaman hazırlanacağı yanında, videonun SHA-1 bloğuna gönderilmesiyle ilgili kısımları organize eden bloktur. Özet tabanlı mesaj doğrulama işleminin bitişinde oluşan 160-bit'lik mesaj özeti ilgili çerçeveye aittir ve aktif videodan sonraki ilk karartma alanına koyulur.





Şekil 4.1-1 Dolgulama bloğu durum akış diyagramı

Dolgulama bloğuna ait durum akışı Şekil 4.1-1'de gösterilmiştir. Dolgulama bloğunda sistem açıldığında ya da sonradan sıfırlama işlemi gerçekleştirildiğinde "BAŞLANGIÇ" durumundan başlar. Bütün sistem çerçeve yapısında çalışması nedeniyle bu durumda yeni çerçevenin başlaması beklenir. Yeni çerçevenin başlamasıyla "IPAD_OPAD_YAZMA" durumuna gidilir ve artık çerçeveye ait hesaplamalar yapılabilir. Öncelikle bu durumda IPAD ve OPAD ile anahtarın XOR'lanması ile oluşan değerlerin belleğe yazılıp yazılmadığı sorgulanır. Yazılmamış olması durumunda "IPAD_OPAD_OKUMA" durumunda beklenir aksi takdirde "IPAD_OKUMA" durumuna gidilir. "IPAD_OKUMA" artık mesaj doğrulama işlemi başlamaktadır. Öncelikle IPAD ile XOR'lanan anahtar bilgisi SHA-1 bloğundan geçirilir. Bu işlem bittiği zaman "ÇERÇEVE_SATIR_KONTROL" durumuna gidilir ve SHA-1 bloğu hesaplamasını bitirene kadar beklenir. SHA-1 bloğu yeni bir hesaplama yapabilecek duruma geldikten sonra aktif video görüntüsü olan 760 satırın herbirinden SHA-1 bloğu oluşturarak SHA-1 işlemi uygulanır. Hesaplamalar sırasında satır sayacı sürekli kontrol edilmektedir. Satır sayacı 760'dan küçük olduğu sürece "VIDEO_OKUMA" durumuna gidilmektedir. Videodan veri toplama işlemi aktif video (DE) senkronizasyon sinyaline göre gerçekleştirilir. Kullanılan video kaynağı XGA (1024x768@60Hz) dir. Videonun görüntülenen kısmında 768 satır bulunmaktadır. Her satırda 1024 benek vardır. Her benek 24 bitten oluşmaktadır. Sistemimiz 768 satırın ilk 760 satırından her satır için bir adet 512-bit'lik SHA1 bloğu oluşmaktadır. Her satırdan verileri toplama işlemi 3 benek atlayarak gerçekleştirilmektedir. İlgili benekleri yeşil renk (8-15) bilgisi değerlendirilmektedir. Satır sayısı 760 değerini geçtikten sonra mesajdan almamız gereken bilgi tamamlanmış demektir. 760 satır kullanılmasının nedeni video ile ilgili satır depolamanın yapılmamasıdır. Aktif video alanı bittikten sonraki kalan 3 satıra eklenmek istenen bilgilerin yetiştirilebilmesi gerekmektedir. Videonun son satırına hesaplanan değerlerin yerleştirildiğini düşünürsek aslında çerçeveyi değerlendirdikten bir çerçeve sonra karar mekanizmasının sonucuyla karşılaşmaktayız. Gerçek zamanlı video hesaplaması tamamlandıktan sonra "ÖZET_HAZIR" durumuna gidilir. Bu durumda daha önceden hesaplanmış olan OPAD ile XORlanmış anahtar SHA-1 bloğuna gönderilir. Bu gönderim sırasında videoda yapılan uygulamalardan farklı olarak SHA-1 bloğunun başlangıç değeri sistem başladığındaki durumuna getirilir. Bir sonraki durum olan "YENİ_SATIR_KONTROL"de ise videonun SHA-1 bloğundan geçmesi sonucu oluşan özet dolgulama işlemi yapılarak OPAD'in özet fonksiyonundan geçmesi sonucu

oluşan değeri kullanarak videoya ait 160-bit'lik özet değerini elde etmiş oluruz. Dolgulama işlemi mesaj uzunluğunu 512-bit'e tamamlamak içindir. Bunun için mesajın sonuna '1' eklenir ve genişletilmiş yeni mesajın son 64-bit'i, esas mesajın boyut bilgisidir. Kalan bütün bitler '0' olacak şekilde mesaj uzunluğu bilgisinden önceki alana yazılır. Sistem "YENİ_SATIR_KONTROL" durumu tamamlandıktan sonra "BAŞLANGIÇ" durumuna giderek yeni çerçevenin gelmesini beklemektedir. Tablo 4.1-1'de dolgulama bloğunun sinyal isimleri ve tanımları yer almaktadır.

Tablo 4.1-1 Dolgulama Bloğu Giriş- Çıkış İşaretleri

İsim	**G/ Ç**	**Açıklama**
Clk	G	Bloğa ait çalışma frekansıdır.
Rstn	G	Sıfırlama sinyalidir. Aktif düşüktür.
Hsync_in	G	Yatay senkronizasyon sinyalidir. Giriş bloğuna girer.
Vsync_in	G	Düşey senkronizasyon sinyalidir. Giriş bloğuna girer.
De_in	G	Aktif video alanını gösteren sinyaltir. Giriş bloğuna girer.
Video_in	G	24-bit RGB video verisidir. Giriş bloğuna girer.
Sha_wren	Ç	SHA-1 bloğuna ait yazma aktif sinyalidir.
Sha_dat[31:0]	Ç	512-bit veriyi yazmak için kullanılan 32-bit SHA-1 bloğu veri sinyalidir.
Hmac_dat[31:0]	Ç	Mesajdan önce kullanılacak olan IPAD ve OPAD verisini hmac_ram'e yazmak için kullanılan 32-bit veri sinyalidir.
Sha_addr[6:0]	Ç	512-bit veriyi yazmak için kullanılan SHA-1 bloğu adres sinyalidir.
Hmac_addr[6:0]	Ç	Mesajdan önce kullanılacak olan IPAD ve OPAD verisini hmac_ram'e yazmak için kullanılan adres sinyalidir.
Blok_start	Ç	SHA-1 bloğuna 512-bit veri gönderildikten sonra çıkan sinyaltir.
Start_hmac	Ç	Yeni frame ile birlikte IPAD ve OPAD verisinin hmac belleğine yazılmasını sağlayan sinyaltir.

Hsync_out	Ç	Yatay senkronizasyon sinyalidir.
Vsync_out	Ç	Düşey senkronizasyon sinyalidir.
Video_aktif_out	Ç	Aktif video alanını gösteren sinyaltir.
Video_out	Ç	24-bit RGB video verisidir.
Finish_sha	G	Bir adet SHA-1 bloğunun sonucunun oluştuğuna dair oluşan bilgidir.
Finish_hmac	G	Belleğe yazılacak verinin başlangıç adresidir.
Frame_start	Ç	Yeni bir çerçevenin başladığını gösterir.

4.2. SHA-1 Bloğunun Yapısı

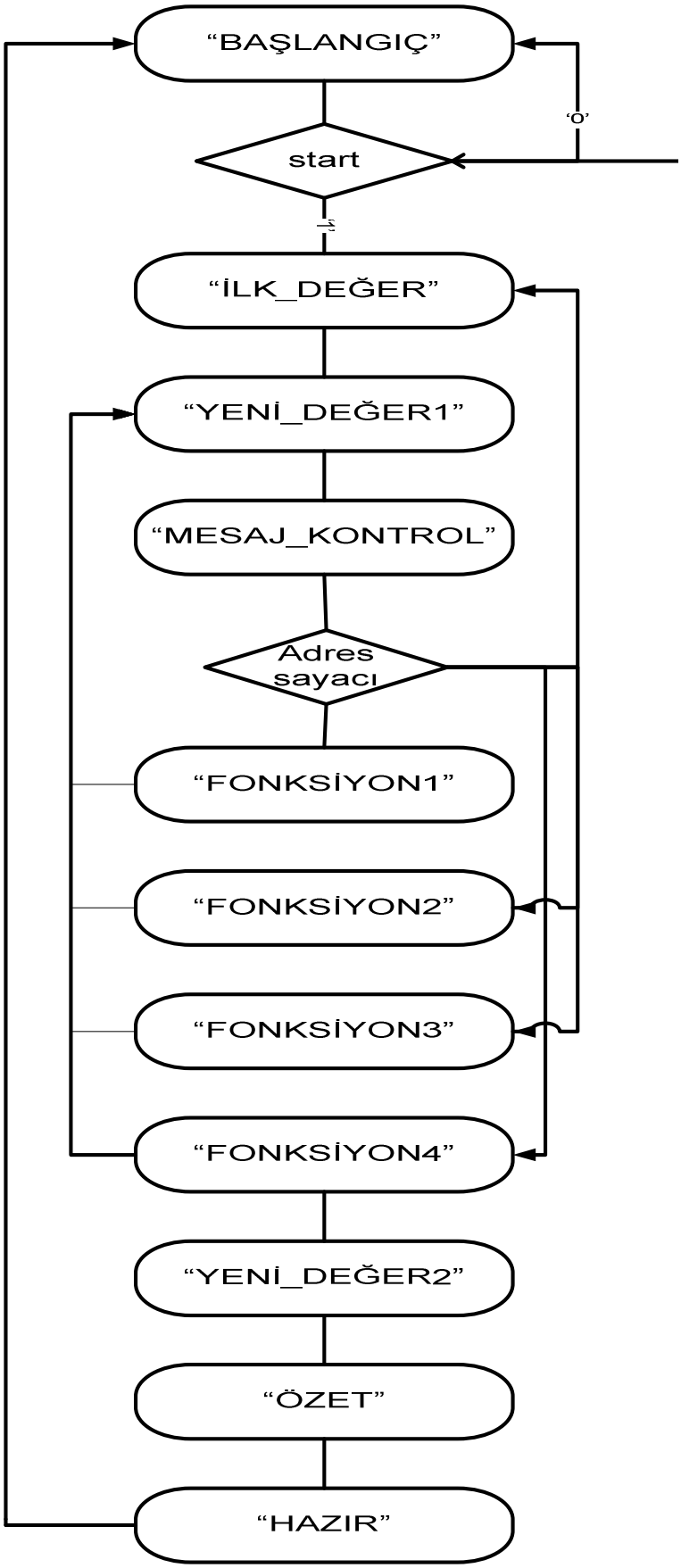
SHA-1 özet fonksiyonunun çalışma yapısı Bölüm 2.2.2'de anlatılmaktadır. Bu kısımda ise SHA-1 özet fonksiyonunun nasıl gerçeklendiğinden bahsedilecektir. SHA-1 bloğunun FPGA tasarımı gerçeklenirken [12] dikkate alınmıştır.

SHA-1 bloğuna ait 512-bit'lik verinin belleğe yazılmasından 124 saat devirinden sonra gönderilen mesajın özeti oluşmuş olur. Tablo 4.2-1'de SHA-1 bloğundan kullanılan sinyallerin özellikleri ve tanımları bulunmaktadır. SHA-1 bloğunun tek blok ve çift blok uygulamalarındaki sonuçları Şekil A.5-2 ve Şekil A.5-3'de yer almaktadır.

Tablo 4.2-1 SHA-1 Bloğu Giriş- Çıkış İşaretleri

İsim	**G/ Ç**	**Açıklama**
Clk	G	Bloğa ait çalışma frekansıdır.
Rstn	G	Sıfırlama sinyalidir. Aktif düşüktür.
Wren_in	Ç	SHA-1 bloğuna ait yazma aktif sinyalidir.
Data_in[31:0]	Ç	512-bit veriyi yazmak için kullanılan 32-bit sha1 bloğu veri sinyalidir.
Data_out[31:0]	Ç	SHA-1 bloğundan okunan 32-bit'lik veridir.

W_addr_in[6:0]	Ç	512-bit veriyi yazmak için kullanılan SHA-1 bloğu adres sinyalidir.
start	Ç	32-bit'lik verinin 80 adrese yazılmasının bittiğini gösteren bilgidir.
Digest[159:0]	Ç	Oluşturulan mesaj özetidir.
Finish	G	Bir adet SHA-1 bloğunun sonucunun oluştuğuna dair oluşan bilgidir.
Frame_start	Ç	Yeni bir çerçevenin başladığını gösterir.



Şekil 4.2-1 SHA-1 bloğu durum akış diyagramı

SHA-1 bloğu 3 farklı kısımdan oluşmaktadır. Bunlardan ilki alınan 512-bit'in 2560 bit-'e çıkarılmasını sağlar. İkinci kısım, oluşan bütün verinin özet fonksiyonunda hesaplanmasını organize eder. Üçüncü kısım ise SHA-1 bloğuna ait sonucun oluştuğunu haber verir.

512-bit'lik mesaj bloğu RAM_SHA içerisinde yer alan 32-bit'lik 16 adrese yazılır. RAM_SHA'ya yazıldıktan sonra 16 adet 32-bit'lik veri (1) hesaplaması kullanılarak 32-bit'lik 80 adrese yazılır. Bu işlem tamamlandıktan sonra SHA-1 değerinin hesaplanması için bütün veriler hazır olur. Start sinyali '1' olduğunda hesaplama işlemine geçilir. İkinci kısım asıl uygulamanın yapıldığı durum makinasıdır. Durum makinasının çalışma yapısı Şekil 4.2-1'de gösterilmiştir. Sistemin açılış durumu "BAŞLANGIÇ"'dır.

SHA-1 hesaplamasında bir önceki mesaj bloğunun özeti bir sonraki blok için başlangıç değeridir. Sadece ilk mesaj bloğunun başlangıç özeti [4] tanımlanan onaltılık sayı sisteminde "67452301EFCDAB8998BADCFE10325476C3D2E1F0" değerine ayarlanır. Bizim sistemimizde ilk özet değerinin kullanımı, yeni çerçevenin başladığına dair gelen "frame_start" sinyalinin '1' olması gerçekleşmektedir. "start" sinyalinin '1' olmasıyla beraber "İLK_DEĞER" durumuna gidilir. Bu durumda ilk veri için yeni değerler hesaplanır ve "YENİ_DEĞER1" durumuna gidilir ve hesaplanan yeni değerler eskilerinin yerine atanır.

RAM_SHA'da kayıtlı olan verilerin adreslerine göre uygulanan fonksiyon (3) ve kullanılan sabit sayılar (2) değişmektedir. Bütün adresler için hesaplamalar tamamlandığında elde edilen 160-bit mesaja ait özettir. "Finish" sinyali SHA-1 uygulaması bittiği anda '1' olur ve yeni bir SHA-1 bloğunun (512-bit) yazılması tamamlanana kadar da '1' olarak kalır.

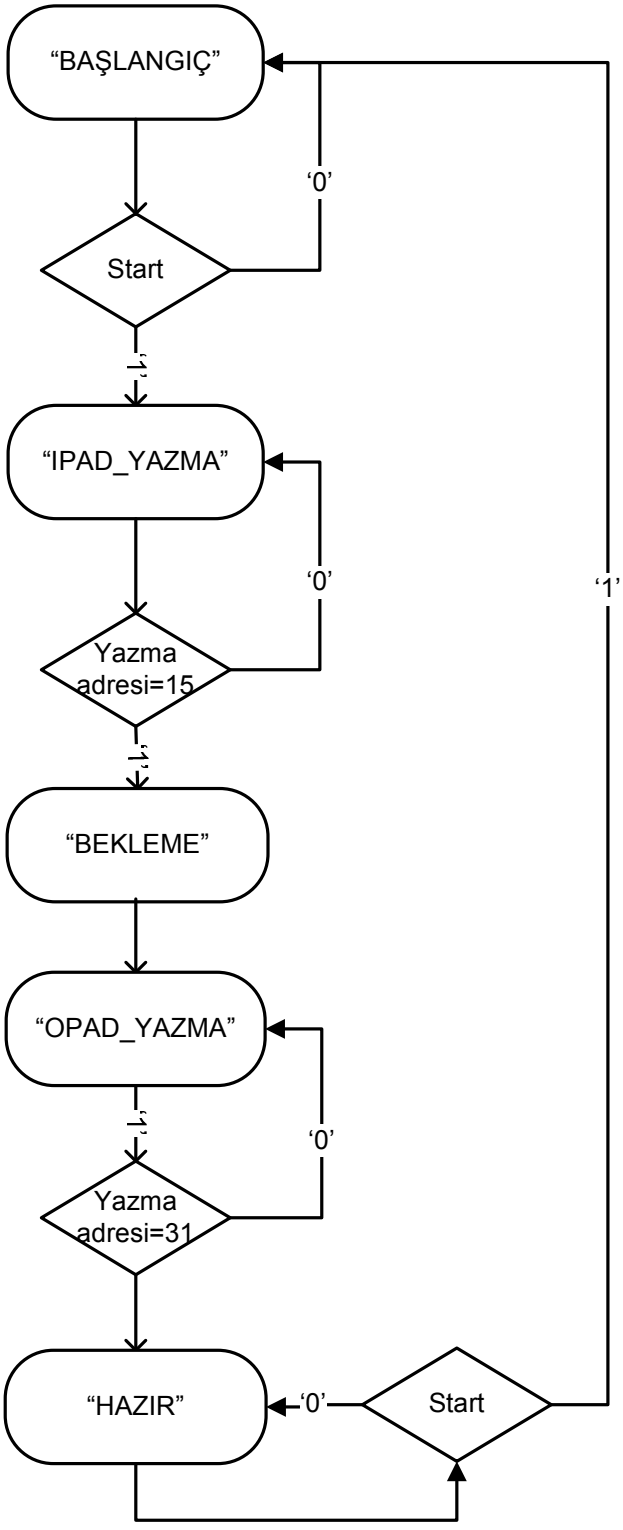
4.3. HMAC Bloğunun Yapısı

HMAC bloğunun asıl kullanım amacı; Anahtarlı Özet Fonksiyonu Tabanlı Mesaj Doğrulama Şifresini oluşturabilmek için kullanılacak anahtar uygulamalarını gerçekleştirmektir. HMAC bloğunu SHA-1 bloğunu uygulayabilmek için tasarlanan bir alt blok olarak düşünebiliriz. HMAC bloğunda özet oluşturmak için gerekli olan IPAD ve OPAD değerleri anahtarla XOR'lanarak daha sonradan kullanılmak üzere RAM_HMAC belleğine yazılır.

Blok içerisinde mesajdan önce ve sonra SHA-1 bloğundan geçirilmesi gereken "IPAD" ve "OPAD" değerlerinin hesaplanması yapılmaktadır. Bu işlem için IPAD (0x36h) peşpeşe sıralanarak 512-bit elde edilir. Aynı işlem OPAD (0x5Ch) için de tekrarlanır.

64-bit'lik anahtar bir SHA-1 mesaj uzunluğu olan 512-bit'e ulaşana kadar sonuna sıfır eklenir. Dolgulama bloğundan gelen "start" sinyali ile birlikte 512-bit'e çıkarılan anahtar ile tekrarlanarak 512-bit'e çıkarılan IPAD değeri ve OPAD değeri XOR'lanarak ilgili bellek alanına yazılır. Başlangıçta "start" sinyalinin tek atım şeklinde mantık '1' olması yeterlidir. "start" sinyalini bir kez görülmesinden sonra belleğe yazma işlemi başlamış olur. Bütün verinin yazılmasından sonra işlemin bittiğini gösteren "finish" sinyali dolgulama bloğuna gönderilir. Yazma işleminin kart açıldıktan sonra bir kez yapılması yeterlidir. Ancak istenirse belleğe yazma işlemi tekrar başlatılabilir.

Tablo 4.3-1'de HMAC bloğundan kullanılan sinyallerin özellikleri ve tanımları bulunmaktadır. Ayrıca Şekil A.5-1'de HMAC bloğuna ait bellek erişiminin simulasyon görüntüsü yer almaktadır.



Şekil 4.3-1 HMAC bloğu durum akış diyagramı

Şekil 4.3-1'de HMAC bloğuna ait akış diyagramı gözükmektedir. Bu akış bloğa ait durumların geçişlerini göstermektedir. Sistemde bulunan eşzamanlı olmayan sıfırlama sinyalinin '0' olması halinde hangi basamakta olursa olsun "BAŞLANGIÇ" durumuna dönülür. "BAŞLANGIÇ" durumunda "start" sinyali mantık '1' olmuşsa "IPAD_YAZMA" durumuna gidilir. Anahtar 64-bit uzunluğunda olduğu için iki farklı değişken farklı durumlarda atanmaktadır. "BEKLEME" durumunda IPAD ile XOR'lanmış anahtar bilgisinin tamamlanması beklenmektedir. Yazma işlemi tamamlandığında "BEKLEME" durumundan "OPAD_YAZMA" durumuna gidilmektedir. OPAD ile XOR'lanmış anahtar bilgisinin yazılmasının tamamlanması beklenmektedir. Yazma işlemi tamamlandıktan sonra "HAZIR" durumuna gidilerek "start" sinyalinin '1' olması beklenmeye başlanır. "start" sinyali iki farklı blokta değerlendirildiği için dolgulama bloğundan gelen bu işaretin bir saat değişimde olması sistemin yapısı için uygundur.

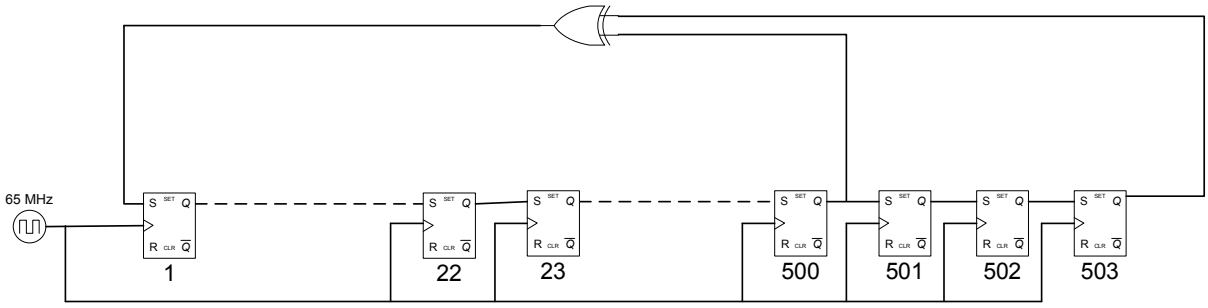
Tablo 4.3-1 HMAC Bloğu Giriş- Çıkış İşareti

İsim	**G/ Ç**	**Açıklama**
Clk	G	Bloğa ait çalışma frekansıdır. Bu tasarımda benek saatinin frekansına eşittir.
Rstn	G	Sıfırlama sinyalidir. Aktif düşüktür.
start	G	Aktifleştime sinyalidir.
IPAD[7:0]	G	0x36h
OPAD[7:0]	G	0x52h
Key[63:0]	G	Özet algoritmasında kullanılan anahtardır.
w_dat[31:0]	Ç	IPAD ve OPAD'den oluşturulan sha blokları istenildiğinde kullanılmak üzere RAM_HMAC'e kaydedilir.
w_addr[5:0]	Ç	RAM_HMAC adres yoludur.
wren	Ç	RAM_HMAC yazma aktif sinyalidir.
finish	Ç	Belleğe yazma işleminin bittiğini gösteren sinyalti.

b_addr[5:0]	G	Belleğe yazılacak verinin başlangıç adresidir.

4.3.1. PRBS Üretecinin Yapısı

Sözde rastgele ikili dizi üreteci, ikili sayı sisteminde rastgele sayılar üretmektedir. Sözde denilmesinin sebebi, rastgele sayılar üretmesine rağmen bir süre sonra hesaplanan sayıların kendini tekrarlamasından kaynaklanmaktadır. PRBS üreteci gerçekleştirebilmek için LFSR kullanılır. LFSR girişi doğrusal bir fonksiyondan oluşan kaymalı kaydedicidir. LFSR, mantıksal kapılardan oluşan seçilen devrelerin çıkışlarının XOR kapısından geçirilerek tekrara sisteme giriş olarak verilmesiyle oluşan döngüsel bir yapıdadır [21]. PRBS bloğunun tasarım yapısı Şekil 3.3.4-1'de gösterilmiştir [22].

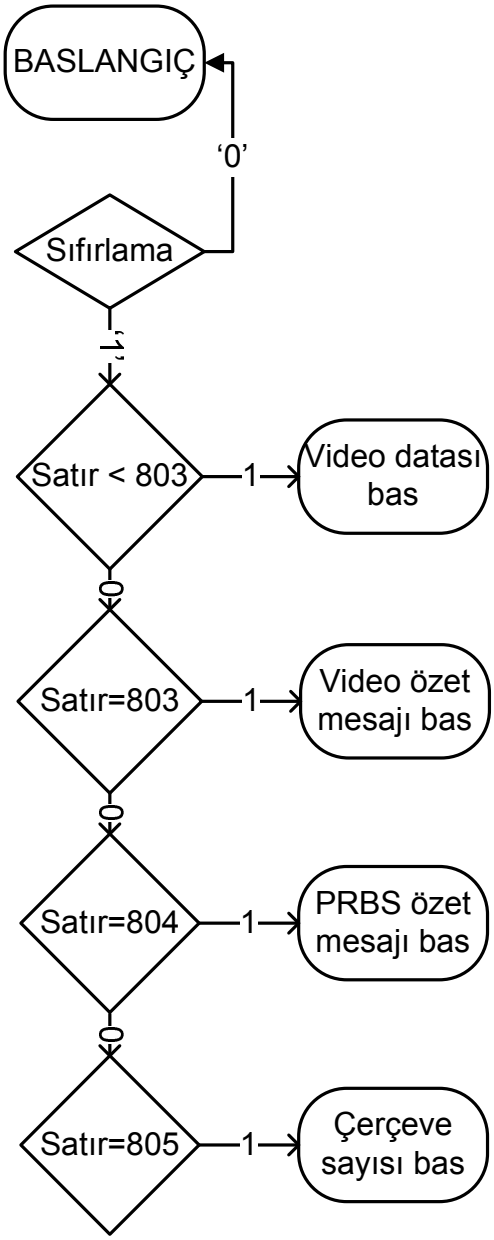


Şekil 4.3.1-1 PRBS bloğunun yapısı

Tablo 4.3.1-1 PRBS Bloğu Giriş- Çıkış İşareti

İsim	G/ Ç	Açıklama
Clk_pix	G	Bloğa ait çalışma frekansıdır. Bu tasarımda benek saatinin frekansına eşittir.
Rstn	G	Sıfırlama sinyalidir. Aktif düşüktür.
PRBS_data[502:0]	Ç	Üretilen sözde rastgele ikili sistemdeki sayıdır.

4.3.2. Çıkış Bloğunun Yapısı



Şekil 4.3.2-1 Kimlik oluşturucu çıkış bloğu durum akış diyagramı

Çıkış bloğu kimlik oluşturucu ve kimlik belirleyici kısımlarda farklılık göstermektedir. Kimlik oluşturucu kısmı gelen videonun saat ve senkronizasyon sinyallerini kullanır. Bu blokta hesaplanan mesaj özetleri video verisine eklenip kimlik oluşturucu kısıma gönderilir. XGA çözünürlüğünde 0'dan 805'e kadar tanımlanan toplam 806 satır bulunmaktadır. 806 satırın ilk 35 satırı aktif olmayan video alanını göstermektedir. 34'üncü satır dahil olmak üzere 803'üncü satıra kadar aktif video alanıdır. Kalan 3 satıra ise hesaplanan özet değerleri eklenmiştir. 803'üncü satırın aktif video alanını temsil eden beneklerden 291-297 benek aralığına video verisinden elde edilen mesaj

özeti eklenmiştir. 804'üncü satırının 291-297 benek aralığına PRBS üreteci kullanılarak elde edilen SHA-1 bloğunun mesaj özeti eklenmiştir. 805'inci satırın 291-296 benek aralığına çerçeve sayaç bilgisi, 297-309 benek aralığına hesaplanan PRBS değeri eklenmiştir.

Kimlik belirleyici kısmında bu veriler monitöre yansıtılmamaktadır. Ancak hatalı durumların oluşmasında ekrana bilgi amaçlı tam ekran X işareti basılmaktadır. Video kaynağının olmaması durumunda hata verebilmek için kimlik belirleyici kısımda kimlik oluşturucu kısımdan farklı olarak FPGA içerisinde PLL kullanılarak kart üzerinde bulunan 50MHz'lik saat sinyalinden 65 MHz üretilmektedir. Hata durumu oluşması halinde kimlik belirleyici çıkış bloğunda üretilen 65 MHz'lik saat sinyali ve senkronizasyon sinyalleriyle LCD monitöre kırmızı renkli X basılmaktadır. Şekil 3.3.5-1'da çıkış bloğunun şematik gösterimi bulunmaktadır. Tablo 3.3.5-1'de bloğa ait sinyal isimleri açıklanmıştır.

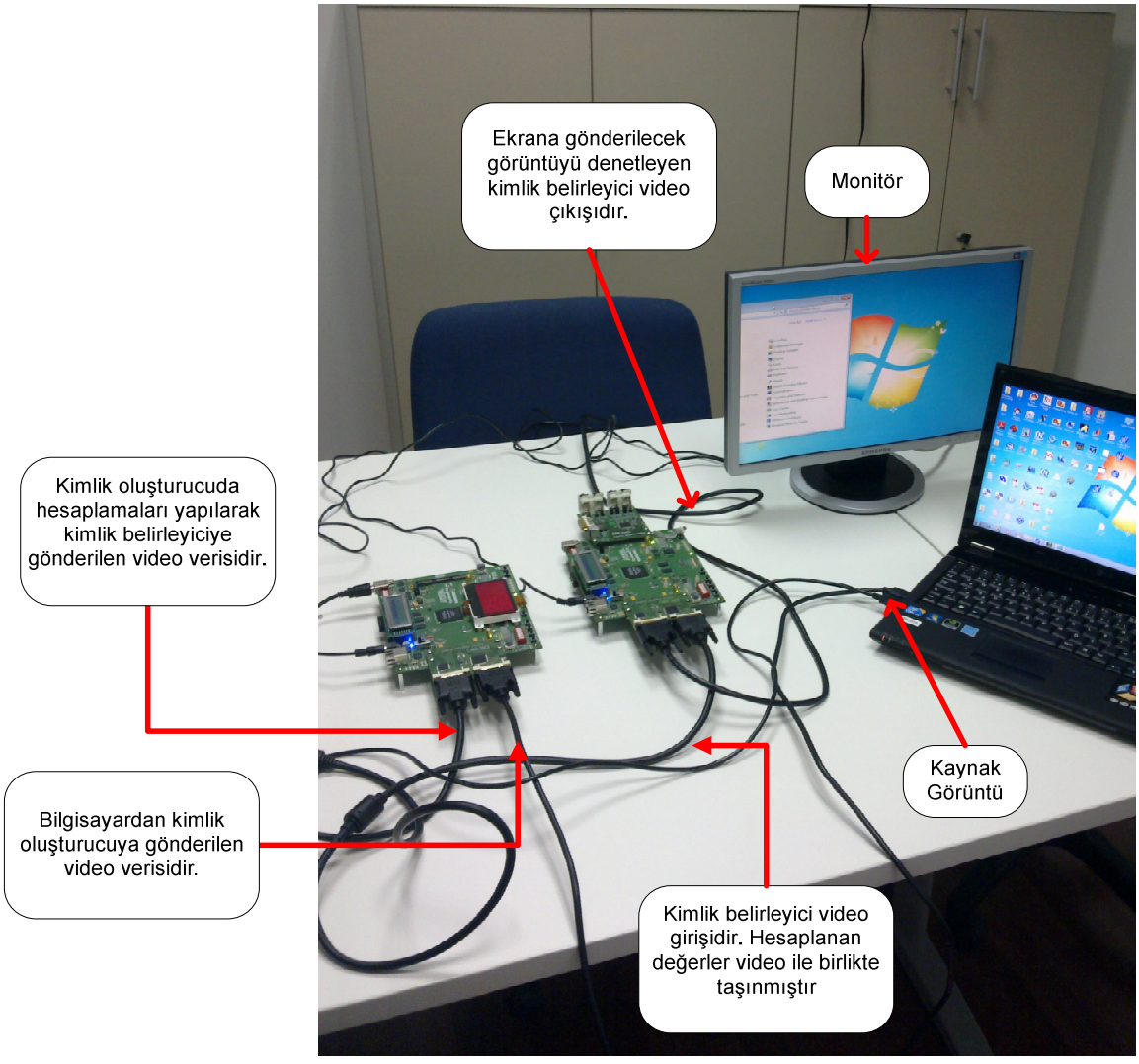
Tablo 4.3.2-1 Kimlik Oluşturucu Çıkış Bloğu Giriş- Çıkış İşareti

İsim	**G/ Ç**	**Açıklama**
Clk_pix	G	Bloğa ait çalışma frekansıdır. Bu tasarımda benek saatinin frekansına eşittir.
Rstn	G	Sıfırlama sinyalidir. Aktif düşüktür.
Hsync_in	G	Gelen video kaynağına ait yatay senkronizasyon sinyalidir.
Vsync_in	G	Gelen video kaynağına ait düşey senkronizasyon sinyalidir.
Video_in[23:0]	G	Gelen video kaynağına ait video verisidir.
Digest_video[159:0]	G	Video bilgileri kullanılarak hesaplanmış mesaj özetidir.
Digest_prbs[159:0]	G	PRBS üreteci ve çerçeve blok sayısından elde edilerek oluşturulan 1 adet SHA-1 bloğunun mesaj özetidir.
PRBS[287:0]	G	503-bit PRBS üretecinin en yüksek 288-bit'inin değeridir.
Frame_counter[127:0]	Ç	Her yeni çerçeve ile değeri bir arttan çerçeve sayaç bilgisidir.

Cnt_line[11:0]	Ç	Video kaynağının satır sayısını gösteren sayaç verisidir. Her yeni çerçeveyle birlikte sıfırlanır.
Cnt_pix[11:0]	Ç	Video kaynağında her satırın benek sayısını hesaplayan sayaçtır. Her yeni gelen düşey senkronizasyon sinyaliyle sıfırlanır.
De_out	G	Videonun ekranda gösterilebilmesi için oluşturulan aktif video senkronizasyon sinyalidir.
Hsync_out	G	Gönderilecek videoya ait yatay senkronizasyon sinyalidir.
Vsync_out	G	Gönderilecek videoya ait düşey senkronizasyon sinyalidir.
Video_out[23:0]	G	Gönderilecek videoya ait video verisidir.

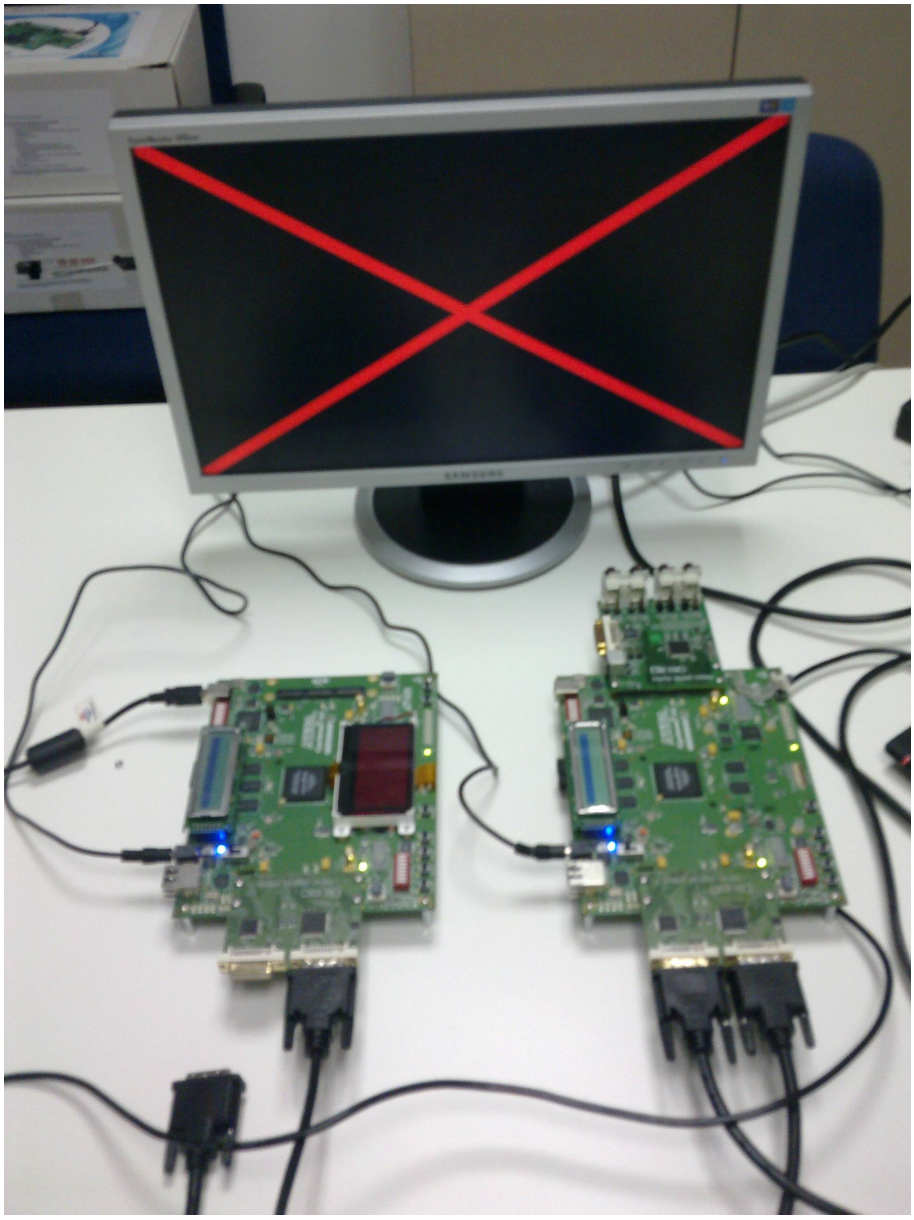
5. GERÇEKLEŞTİRİLEN TASARIM

Motivasyon senaryolarını gerçekleştirebilmek için Ekler kısmında yer alan ALTERA ve BITEC kartları kullanılmıştır. Gösterim için HDMI fiziksel bağlayıcısına sahip olan bir bilgisayar kullanılmakta ve video girişi bu şekilde sağlanmaktadır. HDMI'dan gelen görüntünün ALTERA kartına giriş olarak verilebilmesi için DVI standardına çevrilmesi gerekmektedir. Bu nedenle HDMI girişi DVI'a standartına dönüştüren çevirici bir kablo kullanılmaktadır. Bilgisayardan alınan görüntü kimlik oluşturucu kartına gelmektedir. Kimlik oluşturucunun DVI şifreleyicisinden çıkan video verileri kimlik belirleyici kısmına gönderilmektedir. Kimlik belirleyicinin DVI şifreleyicisinden çıkan video verileri ekrana gönderilmektedir. Bu yapı video doğrulama düzeneği olarak adlandırılacaktır. Video doğrulama düzeneğinin genel görüntüsü Şekil 4-1'deki gibidir.



Şekil 5-1 Video doğrulama düzeneği

Anahtar değerinin tek taraflı olarak değişmesi, video iletim hattının kopması, farklı ya da hesaplama işlemi yapılmayan video kaynağının basılması durumunda ekrana kırmızı X basılmaktadır. İletim hattının kopması durumu Şekil 5-2'de gösterilmiştir.



Şekil 5-2 Video doğrulama düzeneğinde hata durumunun oluşması

Gerçeklenen tasarımın video özellikleri, işlem süreleri ile ilgili bilgi Tablo 5-1'de yer almaktadır. Tasarımı gerçekleyebilmek için kullanılan bit, eleman, fonksiyon ve yazmaç sayısı Tablo 5-2'de verilmektedir.

Tablo 5-1 Gerçeklenen Tasarımın Özellikleri

Video girişleri ve çıkışları:	1024x768 DVI
Veri taşıma kanalı:	Benek'te renk bileşenleri; R, G, B
Benekte kullanılan bit sayısı:	d
Satırda kullanılan bit sayısı:	512
Satırda kullanılan benek sayısı:	512 / (d)
Kullanılan video satır sayısı:	760
Gizli anahtar bit uzunluğu:	64
Eksik video çerçeve algılayıcı:	Var
Çalışma saat periyodu (Pç):	15,38 nsn
Çıktı hızı 512 / (124 * Pç):	413Mbit/sn
İşlem süreleri:	
SHA-1 (124*Pç):	1,9 µsn
hmac1(3*124*Pç):	5,72 µsn
hmac2(763*124*Pç):	1,45 msn

Tablo 5-2 Gerçeklenen Tasarımın Kullanım İhtiyaçları

Kimlik oluşturucu	
Mantık Eleman Sayısı	1,795
Bileşimli Fonksiyon Sayısı	1,646
Toplam Yazmaç Sayısı	955
Toplam Bellek Bit Sayısı	6,144
Kimlik Belirleyici	
Mantık Eleman Sayısı	2,215
Bileşimli Fonksiyon Sayısı	1,959
Toplam Yazmaç Sayısı	1,234
Toplam Bellek Bit Sayısı	6,144

6. SONUÇ

Bu çalışmayla birlikte güvenlik sistemlerinde tehlike oluşturan durumlar farkedilebilmektedir. Yüksek çözünürlüklü videoda anahtarın değişmesi, bağlantı hattının kopması, görüntüde oluşan donmalar ve görüntü verisinde yapılan değişiklikler algılanıp anında sistem kullanıcısı uyarılabilmektedir.

Kurulu bir güvenlik sisteminin kamera hatlarına kolayca bağlanarak sisteme entegre edilebilen ve video aslının doğrulanması ile birlikte aynı zamanda görüntü bütünlüğü işlevlerini sağlayarak sistem güvenirliğini arttıran bir sistem tasarım çalışması gerçeklenmiştir. Sözedilen işlevler kart üzerindeki FPGA içerisinde, gizli anahtar kullanımına olanak sağlayacak şekilde, standart SHA-1 özet alma algoritmasını çalıştıran donanım tasarımı ile sağlanmıştır. Birden fazla kaynak görüntü kullanımı için her kanal için farklı bir gizli anahtar kullanıp kameraya bir kimlik tanınmış olur.

Sistem çalışma yapısı çeşitli çözünürlükleri desteklemektedir. Ayrıca videonun yapısına göre farklılıklar gösterebilmekte ama kolaylıkla uyarlanabilmektedir. Video aslının doğrulanmasında video çözünürlüğünün doğrudan etkisi olmamakta, ancak video benek frekansı belirleyici olmaktadır. Sistemin çalışma saati olarak benek saati kullanılmaktadır. Kullanılan FPGA'in özelliğine bağlı olarak sistem saat hızı değişmektedir. Gösterim için gerçekleştirilen tasarımda SHA-1 bloklarının çıkabileceği en yüksek saat frekansı 125 MHz'dir. Bu değer yüksek çözünürlükte video sinyallerini kapsayabilecek bir değerdir [15]. Video veri bütünlüğünü kontrol etmek amacıyla videonun çeşitli yerlerinden örnekler alınmaktadır. Bu durum veri bütünlüğünü tam kapsamamış olmakla beraber video içerisinde oluşabilecek değişikliklerin farkedilebilmesi için yeterli olabilmektedir. Seçilebilir d değeri görüntü bütünlüğünde kullanılabilir renk sayısını belirlemektedir. Görünen satırların her birinde toplam 512-bit kullanılması ve d değeri, çözünürlüğe bağlı olarak benek kullanım oranını ortaya koymaktadır. Tablo 6-1'de bazı video standartlarına göre çözünürlük (Ç), benek frekansı (F), ve çeşitli d değerleri için satırda benek kullanım (kapsama) oranları (K) verilmiştir.

$$\text{Kapsama oranı} = \frac{\text{1adet SHA-1 bloğuna ait bit sayısı}(512)}{\text{Çözünürlükteki benek sayısı}} \times \frac{\text{Çözünürlükteki satır sayısı-8}}{\text{Çözünürlükteki satır sayısı}} \times 100$$

Tablo 6-1 Uygulanabilen video standartları ve satırda benek kullanım oranları

Video	**Ç**	**F**	**K d=1**	**K d=2**	**K d=4**	**K d=8**
PAL	720 / 625	13.5	%70	%35	%17.5	%8.5
NTSC	720 / 525	13.5	%58	%29	%14.5	%7
VGA	640 / 480	25	%78	%39	%19	%9
SVGA	800 / 600	40	%63	%31	%15	%7
XGA	1024 / 768	65	%49	%24	%12	%6
1080p25	1920 / 1080	74.25	%26	%13	%6	%3

Gelecekte, geliştirmeye yönelik yapılacak çalışmalarda, satırda kullanılan bit sayısını arttırmak, kimlik belirleyici işlemcisine hata düzeltici eklemek, eş zamanlı değişen güvenlik anahtarı eklemek, güvenlik kamerasının algılayıcısına doğrudan arayüz oluşturmak hedeflenmelidir.

KAYNAKLAR DİZİNİ

[1] William Stallings, Cryptography and Network Security Principles and Practices, Fourth Edition, pp. 319-331, 2005

[2] FIPS PUB 198, Issued 6 March, 2002

[3] Tapan Desai, Roar Lien, Project Specification for VHDL implementation of HMAC using SHA-1

[4] FIPS PUB 180-2, 1 August, 2002

[5] Kimmo Järvinen, Design and Implementation of a SHA-1 Hash Module on FPGAs, 25 November, 2004

[6] D. Eastlake, US Secure Hash Algorithm 1 (SHA1), Eylül 2001

[7] Milind M. Parelkar, Authenticated Encryption in Hardware, A Thesis Submitted to the Graduate Faculty of George Mason University, September 2005

[8] Nicholas D. Beser, Thomas E. Duerr, Gregory P. Staisiunas, Authentication of Digital Video Evidence, The John Hopkins University Aplied Physics Laboratory, pp. 1-8, 17 September, 2003

[9] Keith Jack, Video Demystified A Handbook for the Digital Engineer 4th Edition, Newnes, pp. 6-11, 172-177, 265-293, 2005

[10] Siu F. Yeung, John C. S., David K. Y., A Case for a MultiKey Secure Video Proxy: Theory, Design, and Implementation, National Science Foundation, CCR-9875742

[11] Murat Aşkar, Tuğba Şıltu Çelebi, Design and FPGA Implementation of Hash Processor, ISCTurkey, 13 December, 2007

[12] CERG at George Mason University, Hardware Interface of a Secure Hash Algorithm (SHA), 17 October, 2009

[13] Introduction to the Quartus II Software version 10.0 Handbook, ALTERA, pp. 11-15, 2010

[14] Mentor Graphics Modelsim and QuestaSim Support, Quartus II versiyon 11 Handbook, ALTERA, Volume 3, pp. 1-10, May 2011

[15] EBU UER High Definition (HD) Image Formats for Television Production, EBU TECH 3299, Geneva, January, 2010.

[16] R. Ward, T. Molteno, Table of Linear Feedback Shift Registers, Dept. of Physics, University of Otago, New Zealand, 26 October, 2007

[17] Stefan Thiemert, Hichem Sahbi, Martin Steinebach, Using entropy for image and video authentication watermarks, Security, Steganography, and Watermarking of Multimedia Contents VIII, 2006

[18] Dimitrios Skraparlis, Design of an Efficient Authentication Method for Modern Image and Video, 9 March, 2003

[19] Chih-Hsuan Tzeng and Wen-Hsiang, A New Technique For Authentication Of Image/Video For Multimedia Applications, Tsai Department of Computer and Information Science, National Chiao Tung University, 2001

[20] Pradeep K. Atrey, Wei-Qi Yan, Ee-Chien Chang, Mohan S., A Hierarchical Signature Scheme for Robust Video Authentication using Secret Sharing, Kankanhalli School of Computing, National University of Singapore, Proceedings of the 10th International Multimedia Modelling Conference, IEEE, 2004

[21] Sandeep Mukherjee, Ruchir Pandey, Design And Implementation Of PRBS Generator Using VHDL, Department of Electronics & Communication Engineering, National Institute of Technology, Rourkela, 2007

[22] Roy Ward, Tim Molteno, Table of Linear Feedback Shift Registers, 26 October, 2007

[23] Kesavan Gopal, Dr. M. Madhavi Latha, Watermarking of Digital Video Stream for Source Authentication, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 4, No 1, July 2010

[24] Raymond B. Wolfgang, Christine I. Podilchuk, Perceptual Watermarks for Digital, Images and Video, IEEE, July 1999

[25] Pradeep K. Atrey · Wei-Qi Yan · Mohan S. Kankanhalli, Scalable Signature Scheme For Video Authentication, LLC 2006

[26] Qibin Sun, Dajun He, Qi Tian, A Secure and Robust Authentication Scheme for Video Transcoding, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16, No. 10, October 2006

[27] Digital Visual Interface DVI, Revision 1.0, 2 April, 1999

EKLER DİZİNİ

EK.A Kullanılan Yazılım Araçları

Bu kısımda tasarım aşamasında kullanılan araçlar açıklanmıştır. Tasarımın yapıldığı ve VHDL kodlarının hazırlandığı araç ALTERA Quartus II yazılımıdır. Yazılan kodların fonksiyonel çalışmasını kontrol etmek amacıyla kullanılan araç Modelsim'in ALTERA tasarım çekirdeklerini destekleyen versiyonudur.

A.1. ALTERA Quartus II

Altera Quartus II tasarım yazılımı, ALTERA FPGA ve CPLD'lerinin VHDL ya da verilog kodlarının yazımı için kullanıcı arayüzü sağlar. Ayrıca kullanılan ALTERA ürünlerinin araç yardımıyla tanıtılmasıyla beraber, aygıta özel mantık yerleşimi yapmaktadır. Quartus II yazılımı, Quartus II kullanıcı grafik arayüz ve komuta zinciri arayüzünün bütün adımlarının tasarım akışında kullanılmasını sağlar. Bu arayüzlerden bir tanesi seçilip bütün tasarım akışında kullanılabilir ya da farklı evrelerde farklı opsiyonlar kullanılabilir.

Aşağıdaki adımlar Quartus II kullanıcı grafik arayüzüne ait temel tasarım akışını tanımlamaktadir [13].

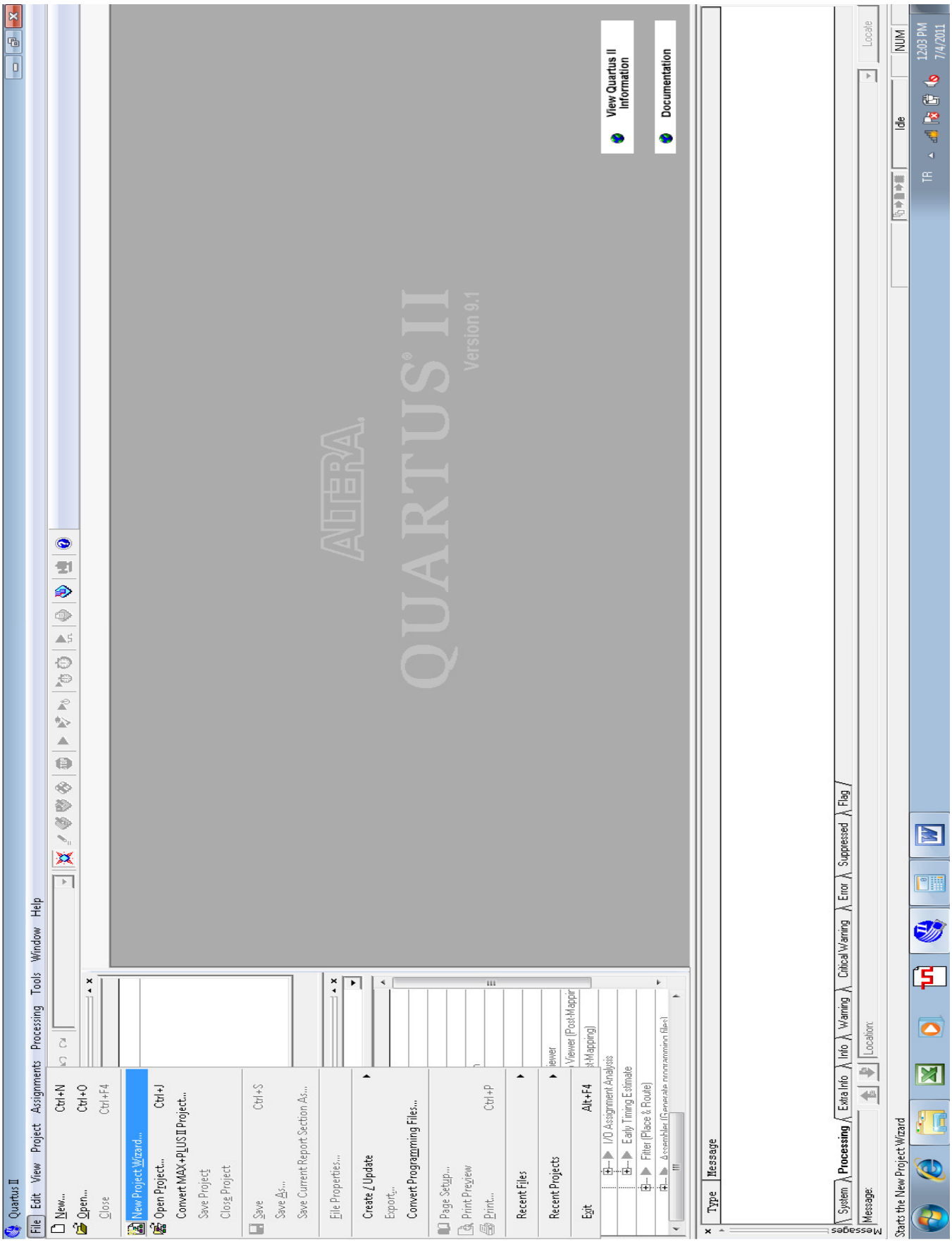
1. Dosya dizininin altında **New Project Wizard** seçeneğini seçerek yeni proje oluşturup özel bir aygıtı ve ya aygıt ailesini belirleyin. Şekil EK 7-5'e bakınız.
2. Verilog HDL, VHDL ya da AHDL (Altera Hardware Description Language) oluşturmak için Text Editör kullanın.
3. Tasarım dosyalarından blok diyagramda kullanabilmek için sembol oluşturulabilir ya da şematik dosyası hazırlanabilir. Bunun için Blok Editör kullanılmaktadır.
4. Quartus II' de tanımlı olan megafunctionsları ve IP fonksiyonları kendi tasarım yapımıza uygun hale getirebilmek için **MegaWizard® Plug-In Manager** kullanılır. Sistem seviyesinde tasarımlar oluşturmak için SOPC Builder ve DSP Builder kullanılır.
5. Assignment Editor, tasarıma özel ilklendirme kısıtlamaları için kullanılır. Ayrıca diğer ayarlamalar için Pin Planner, the Settings dialog kutusu, the Device

dialog kutusu, the Chip Planner, the Design Partitions penceresi ya da the Design Partition Planner kullanılır.

6. (Opsiyonlu) Fitting işleminden önce kestirilen zamanlama bilgileri belirtilebilir.
7. “Analysis & Synthesis” adımı ile tasarım sentezlenir.
8. (Opsiyonlu) Eğer tasarım parçalara ayrılmıssa ve bütün bir derleme yapılmayacaksa, partion merge kullanılarak parçalar birleştirilir.
9. (Opsiyonlu) Tasarımın fonksiyonel simulasyon net listesini oluşturmak için ve fonksiyonel simulasyonunu yapmak için “EDA simulation tool” kullanılır.
10. Tasarımı fit edebilmek için “ Place and route” adımı gerçeklenir.
11. Güç kestirimi ve analizi yapabilmek için “PowerPlay Power Analyzer” kullanılır.
12. Tasarımın zamanlama simulasyonunu yapabilmek için “EDA simulation tool” kullanılır.
13. Tasarımın zamanlama analizini yapabilmek için “TimeQuest Timing Analyzer” kullanılır.
14. (Opsiyonlu) Tasarıma ait zamanlama problemlerini çözmek için the “Chip Planner”, “LogicLock™” regions” ve “Assignment Editor” kullanılır.
15. “Assembler” adımı kullanılarak tasarımdan programlama dosyası oluşturulur. Ve seçilen aygıt “Altera Programmer” kullanılarak programlanır.
16. (Opsiyonlu) Tasarımı debug yapabilmek için “SignalTap® II Logic Analyzer”, “

external logic analyzer”, SignalProbe” özelliği ya da “Chip Planner” kullanılabilir.

17. (Opsiyonlu) Teknik değişiklikler “Chip Planner”, “Resource Property Editor” ve “Change Manager” seçimleriyle gerçekleştirilebilir.



Şekil A.1-1 ALTERA Quartus II Kullanıcı Arayüzü

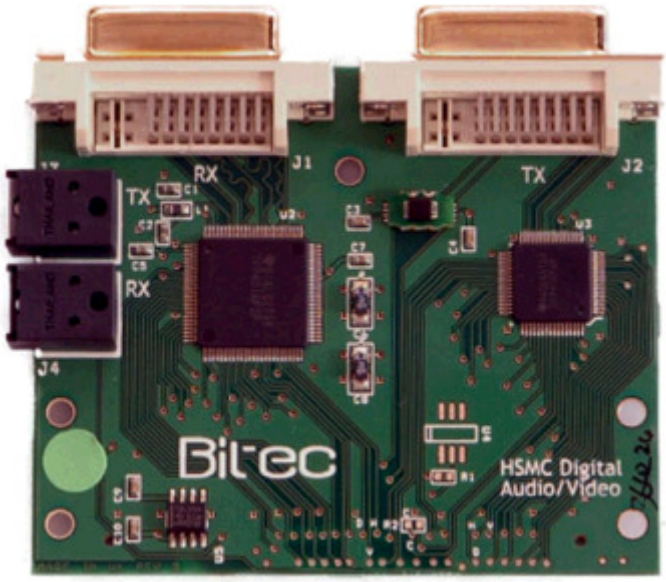
A.2. Modelsim ALTERA

Altera, Modelsim- Altera yazılımı ile tasarımların simulasyonunun basitleştirilmesini hedeflemiştir. Bu nedenle bu yazılımda Altera'nın simulasyon kütüphanesi bulunmaktadır. Böylelikle Altera aygıtları için önceden sentezlenmiş Altera simulasyon kütüphanesi kullanılmaktadır [14].

Fonksiyonel simulasyon kodun söz dizimi hatalarını ve tasarımın fonksiyonel olarak düzgün çalıştığını gösterir. "ModelSim - Altera Waveform Editor" kullanılarak tasarım simulasyonu için basitçe stimülüs oluşturulabilir.

A.3. DEMO

Video güvenlik kartının FPGA tasarımını gerçeklemek ve sistemin çalışabilir olduğunu göstermek için hazırlanan demoda Altera EP3C120F780 geliştirme kartı ve Bitec HSMC DVI kartı kullanılmıştır. HSMC DVI kartı üzerinde DVI alıcı ve DVI verici bulunmaktadır. DVI alıcı olarak kullanılan entegre Texas Instruments tarafından geliştirilen TFP401A' dır. Differensiyal olarak gelen DVI video inputu bu entegreden geçtikten sonra 24-bit RGB verisi olarak videoyu yaşımaktadır. Aynı kart üzerinde bulunan ve yine Texas Instruments tarafından geliştirilen TPF410 entegresi doğru senkronizasyon zamanlamaları ve benek saatiyle gelen video görüntüsünü DVI standardına çevirmektedir.



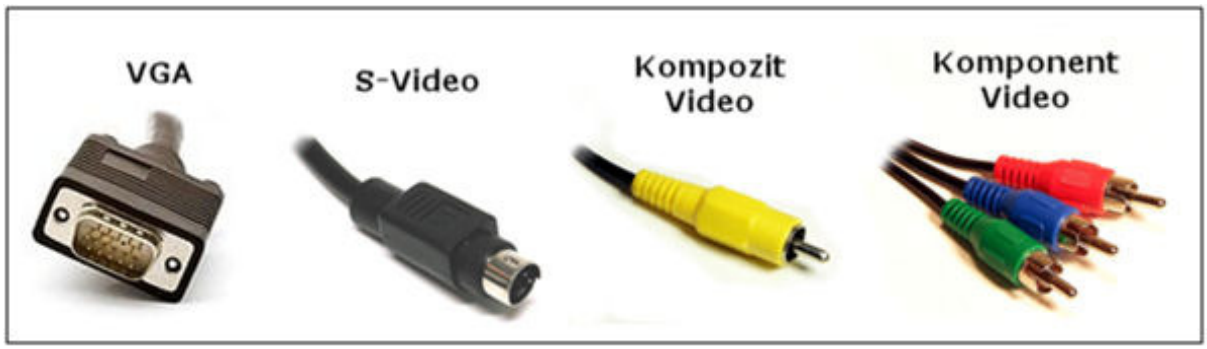
Şekil A.3-1 BITEC HSMC DVI Kartı

HSMC DVI kartından çıkan sinyaller Altera EP3C120F780 geliştirme kartına bağlanmaktadır. EP3C120F780 Altera tarafından geliştirilen Cyclone III serisi FPGA'dır.

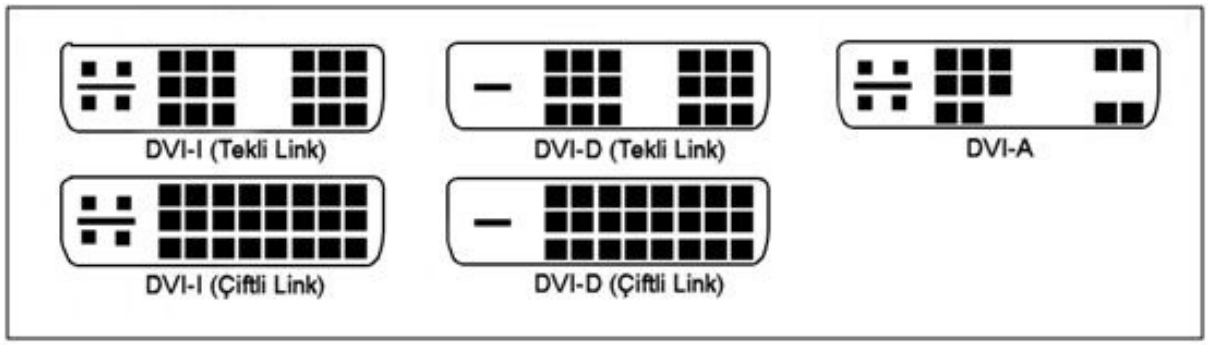
Şekil A.3-2 ALTERA EP3C120F780 Geliştirme Kartı

EP3C120F780 özellikleri	Miktarı
Mantıksal eleman sayısı	119,088
Bellek boyutu	3,888
Çarpan miktarı	288
PLL sayısı	4
Çevresel saat sayısı	20

A.4. VİDEO FİZİKSEL BAĞLAYICILAR



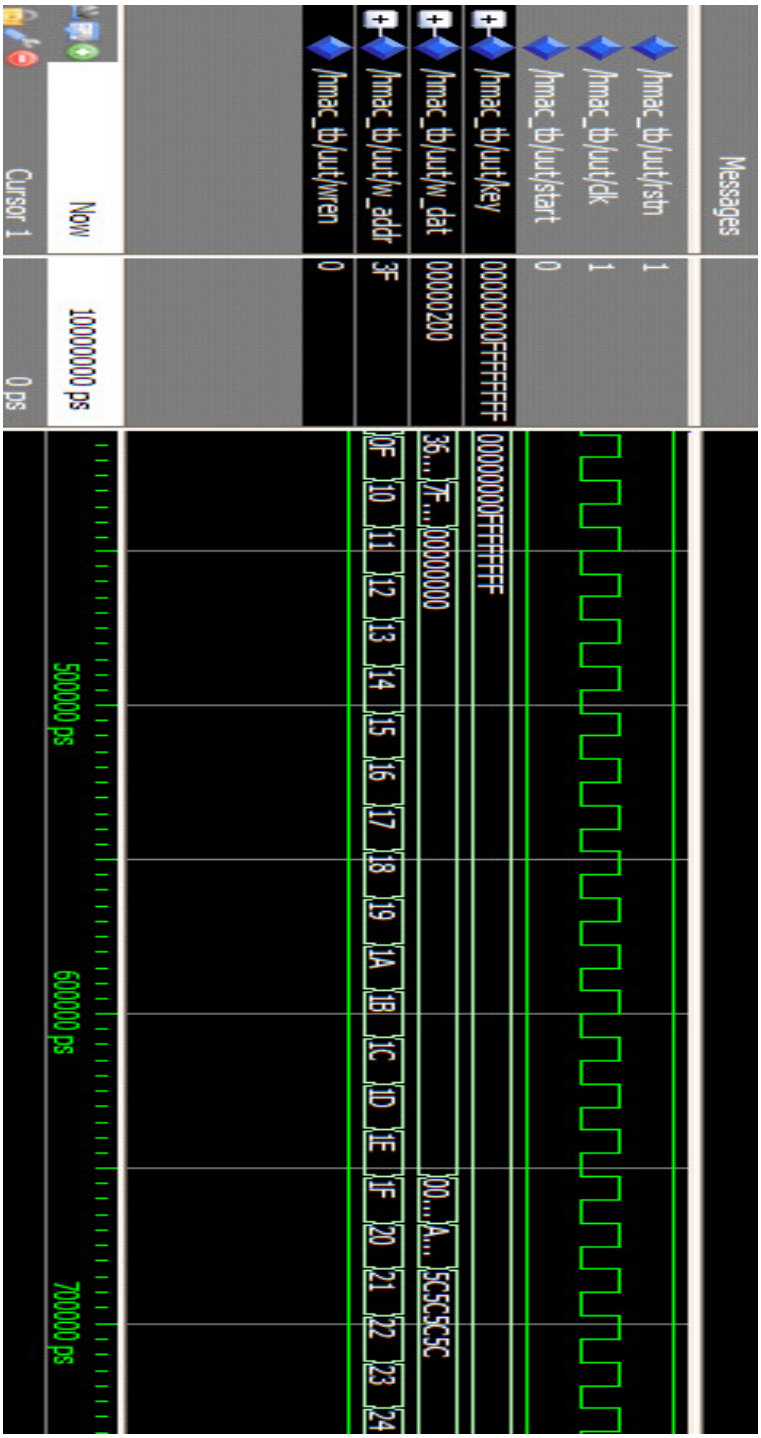
Şekil A.4-1 Analog video bağlayıcıları



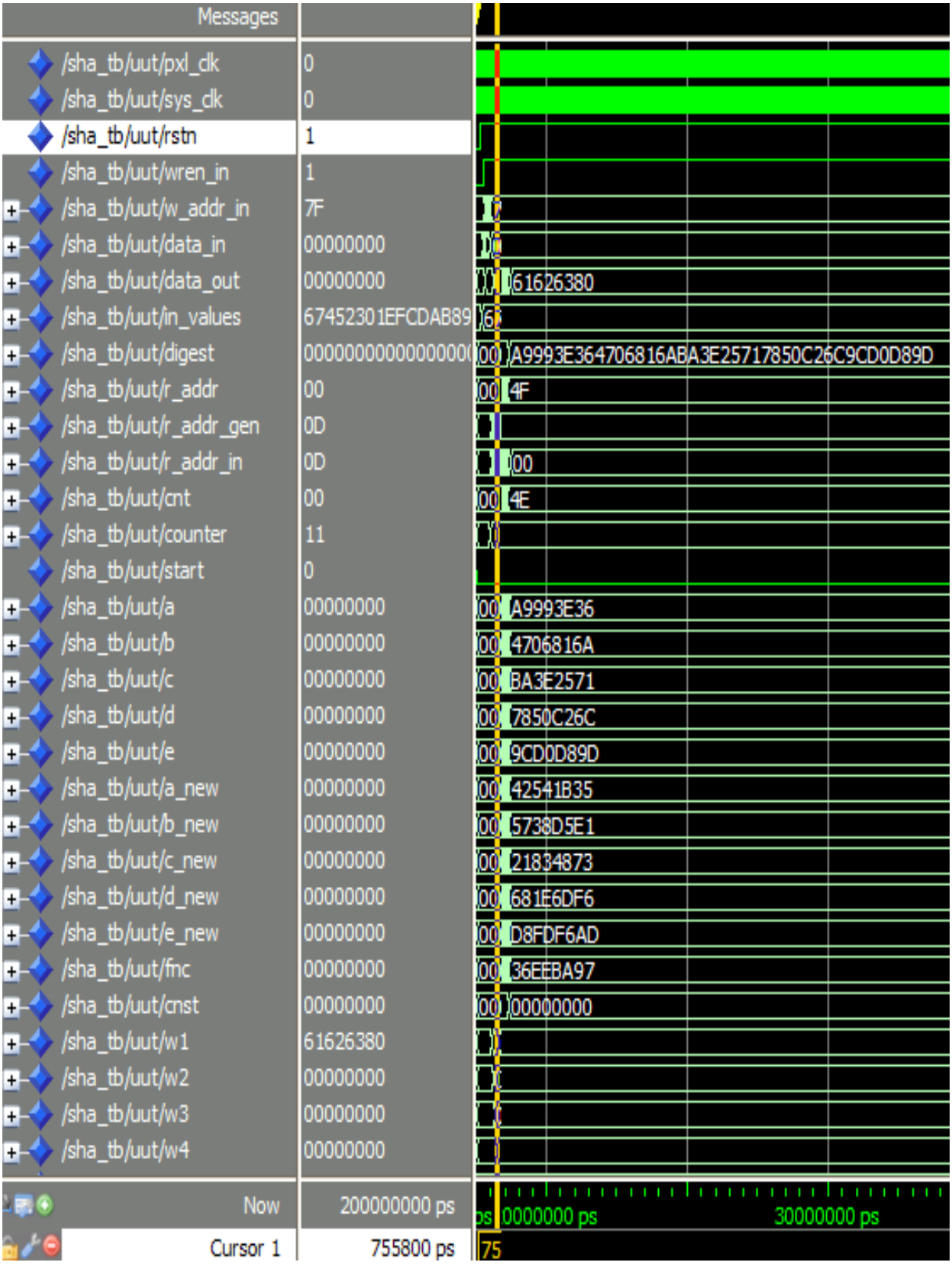
Şekil A.4-2 DVI bağlayıcı tipleri

Şekil A.4-3 HDMI konnektörü

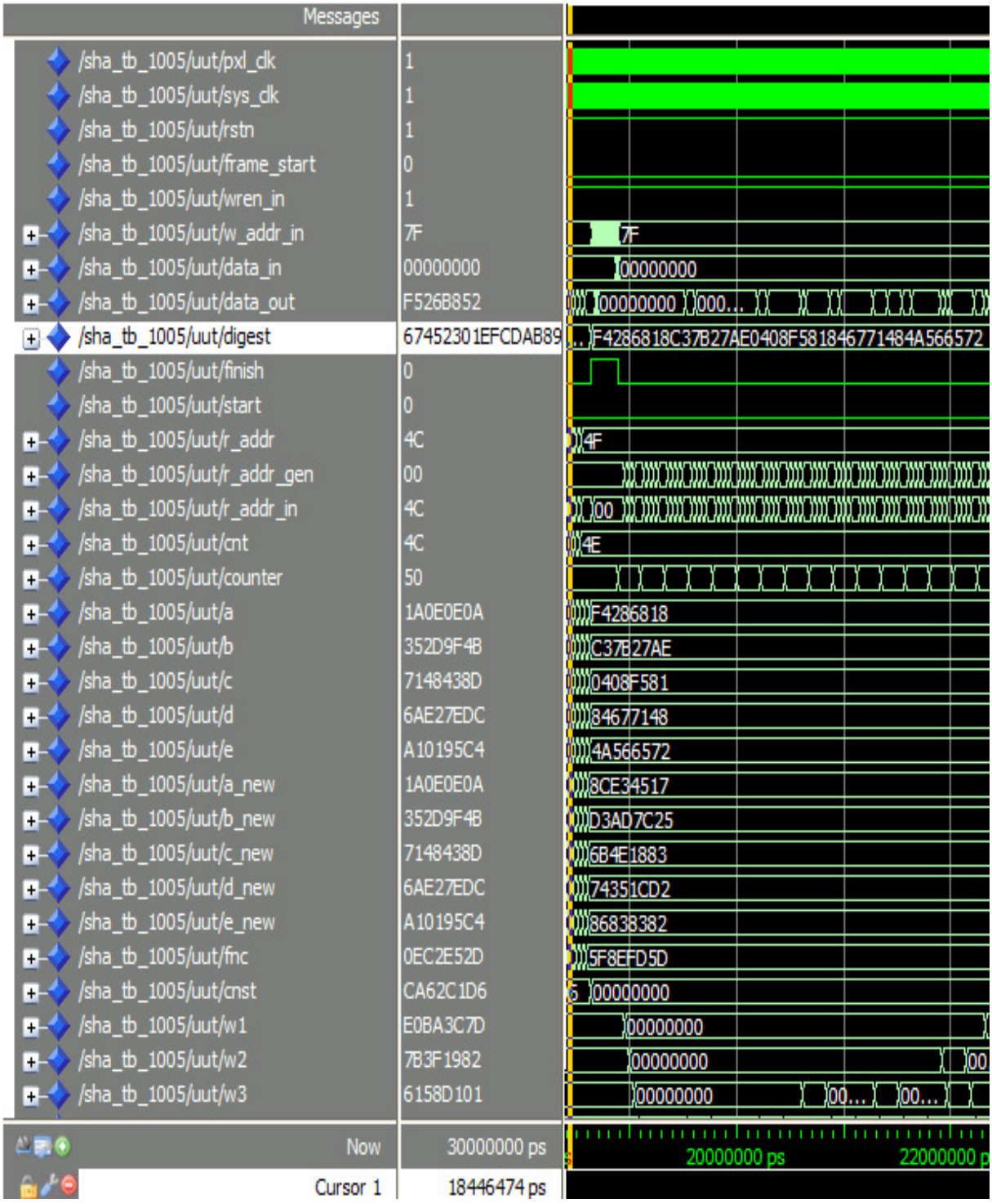
A.5. TASARIM BLOKLARI SİMÜLASYON ÇIKTILARI



Şekil A.5-1 HMAC bloğu bellek erişimi



Şekil A.5-2 Tek blok SHA-1 algoritmasının simulasyon sonucu



Şekil A.5-3 İkili SHA-1 algoritmasının simulasyon sonucu

ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı	:	Özlem Uyanık
Doğum Yeri	:	Ordu
Doğum Yılı	:	15.04.1985
Medeni Hali	:	Bekar

Eğitim ve Akademik Durumu

• İlkokul	:	Ordu Altınfındık İlköğretim Okulu	1991-1996
• Ortaokul	:	Ordu Anadolu Lisesi	1996-2000
• Lise	:	Ordu Fen Lisesi	2000-2003
• Lisans	:	Eskişehir Osmangazi Üniversitesi Elektrik ve Elektronik Mühendisliği Bölümü, ESKİŞEHİR	2003-2008
• Yüksek Lisans	:	Hacettepe Üniversitesi Elektrik ve Elektronik Mühendisliği Bölümü, ANKARA	2008- ...
• Yabancı Dil	:	İngilizce, Almanca, Fransızca	

İş Tecrübesi

• Karel A.Ş	:	2008 - …	Tasarım Mühendisi