İSTANBUL TEKNİK ÜNİVERSİTESİ

ELEKTRİK-ELEKTRONİK FAKÜLTESİ

GÜVENLİ RFID SİSTEMLERDE KULLANILAN BİR MESAFE SINIRLAMA PROTOKOLÜNÜN FPGA ÜZERİNDE GERÇEKLENMESİ

BİTİRME ÖDEVİ

Hasan ÜNLÜ

040060352

Bölümü: Elektronik ve Haberleşme Mühendisliği

Programı: Elektronik Mühendisliği

Danışmanı: Yrd. Doç. Dr. Berna ÖRS YALÇIN

MAYIS 2011

ÖNSÖZ

Bitirme ödevim boyunca bana zaman ayıran ve gereken desteği veren Sayın Yrd. Doç. Dr. Berna Örs Yalçın'a, desteğini hiçbir zaman esirgemeyen aileme ve arkadaşlarıma teşekkürlerimi sunarım.

Mayıs 2011 Hasan ÜNLÜ

İÇİNDEKİLER

KISALTMALAR

FPGA	**:** Field Programmable Gate Array
HDL	**:** Hardware Description Language
ISIM	**:** Xilinx ISE Simülatör
RF	**:** Radio Frequency
RFID	**:** Radio Frequency Identification
SHA	**:** Secure Hash Algorithm
JTAG	**:** Joint Test Action Group
USB	**:** Universal Serial Bus
LED	**:** Light Emitting Diode
XOR	**:** Exclusive OR (Dar Veya)

ŞEKİL LİSTESİ

Sayfa No

ÖZET

SUMMARY

1. GİRİŞ

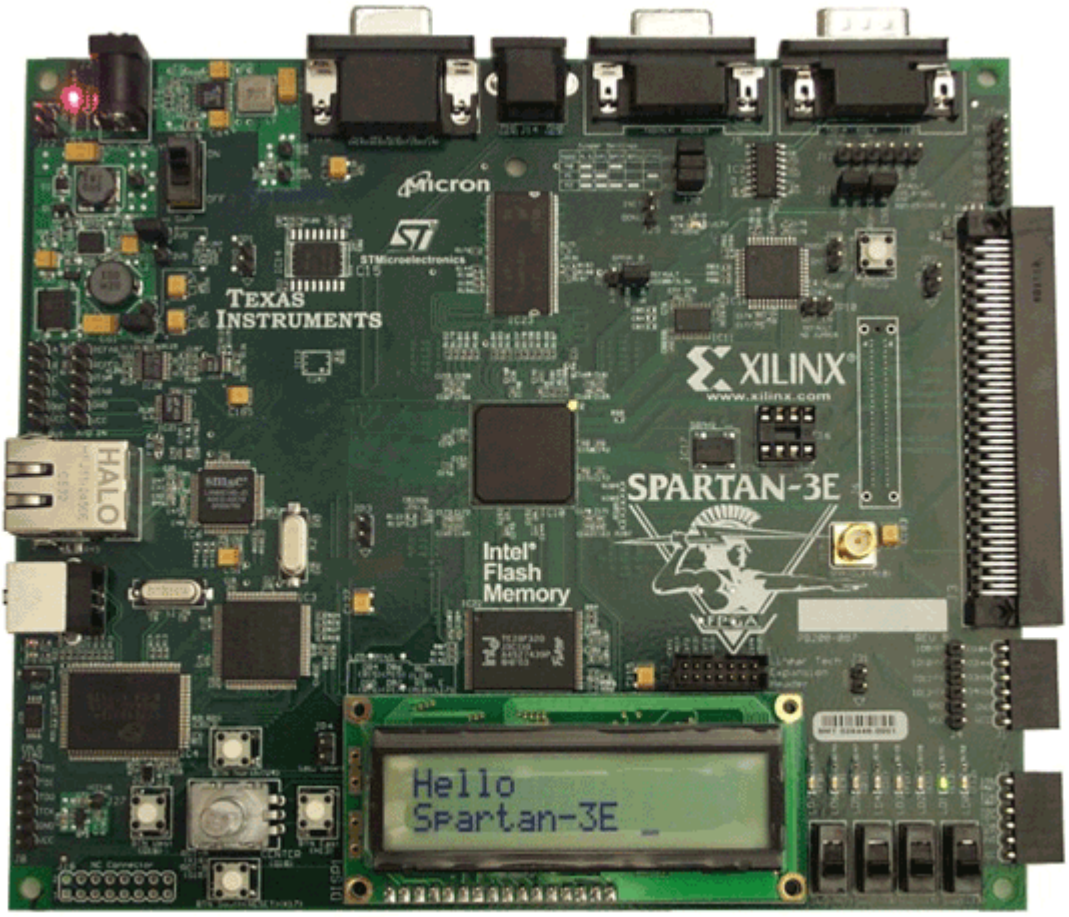
Radyo Frekansı ile Tanımlama (RFT – Radio Frequency Identification RFID) teknolojisi üretim, ürün tedarik zinciri yönetimi, döküm kontrolü gibi alanlarda yaygın olarak kullanılmaktadır ref. Düşük üretim maliyeti ve küçük boyutları sebebiyle barkod gibi geleneksel doğrulama yöntemlerinin yerini almaktadır. Barkod yöntemine göre avantajı doğrulamanın uzaktan yapılabilmesidir. Doğrulamanın uzaktan yapılmasının avantajlarına karşın, çözülmesi gereken yeni problemler ortaya çıkmaktadır. Bu problemlerden ilk akla gelenleri, paylaşılan bilginin gizli kalması, verilerin istenmeyen kişiler tarafından dinlenmesi, okuyucu ya da etiketin kopyalanması gibi sorunları ortaya çıkarmaktadır.

RFTID sistemleri daha güvenli hale getirmek için kullanılan protokollerden biri de mesafe sınırlama protokolüdür ref. Bu protokol kimlik doğrulamanın yanında etiketin okuyucuya göre konumunun belirlenen bir aralıkta olup olmadığınıda kontrol eder. Bitirme projesi kapsamında Hancke ve Kuhn'un [1] önerdiği mesafe sınırlama protokolü Sahada Programlanabilir Kapı Dizisi (SPKD – Field Programmable Gate Array – FPGA ref) içeresinde gerçeklenecektir.

2. KULLANILAN DONANIM VE YAZILIMLAR

2.1. Donanımlar

Mesafe sınırlama protokolü, Digilent firmasının ürettiği Spartan 3E geliştirme kartında gerçeklenmiştir ref. Bu kart üzerinde Xilinx firmasının XC3S500E SPKD~~FPGA~~'sisı bulunmaktadır. Kart üzerinde 8 adet LED, 4 adet anahtar, 4 adet düğme, VGA portu, PS/2 portu, seri iletişim arayüzü, sayısal analog çevirici, analog sayısal çevirici ve bellek entegreleri, 50 MHz kristal bulunmaktadır [2]. Kart üzerindeki USB port ise SPKD~~FPGA~~'yı programlamak için kullanılır. Şekil 2.1'de kartın üstten görünüşü görülmektedir.



Şekil 2.1 : Spartan 3E geliştirme kartı [2].

2.2.Yazılımlar

FPGA'ları programlamak için donanım tanımlama dilleri (DTD – Hardware Description Language – HDL ref) kullanılmaktadır. Verilog HDL ref ve very high .... (VHDL ref) en çok kullanılan donanım tasarlama dilleridir. Donanım tasarlama dili ile tasarlanacak sistem mantıksal şekilde modellenir. Bu modelleme işlemi kodlar aracılığıyla olmaktadır. Bu kodlar sentezleme programlarıyla indirgenir ve devreye dönüştürülür. Devreye dönüştürme işlemi sırasında FPGA içinde hangi bağlantıların yapılacağı, hangi dilimlerin (slice) kullanılacağı belilenir. Bu tanımlamalar tek bir dosyaya dönüştürülür. Bu dosya FPGA'yı JTAG yoluyla programlar.

Projede sentezleme ortamı olarak Xilinx ISE Design Suite 12.2 ref, donanım dili olarak Verilog HDL ve benzetim programı olarak yine Xilinx firmasına ait ISIM refkullanılmıştır.

3. RFID SİSTEMLER

3.1.RFID Sistem Elemanları

RFID Sistemlerde kullanılan elemanlar etiket ve okuyucu olarak incelenebilir.

3.1.1. Etiket

Etiket içerisinde veri saklamak için bir tüm devre ve bu veriyi istenildiğinde radyo frekansı aracılığıyla okuyucuya göndermek için bir anten barındırır. Etiketler güç kullanımına göre pasif, yarı aktif ve aktif olmak üzere üçe ayrılır [3].

Pasif etiket bir güç kaynağı barındırmaz. Okuyucunun gönderdiği elektromanyetik dalgadan elde ettikleri güçle çalışırlar. Okuyucuya yanıt vermek için yine gönderilen elektromanyetik dalgadan yararlanırlar. Sınırlı güçle beslendikleri için okuma menzili ve gönderdikleri veri sınırlıdır. Aktif etiketler üzerlerinde bir güç kaynağı bulundurur. Bu sayede dışarıdan bir uyarıya gerek kalmadan iletişim kurabilir ve üzerindeki veriler üzerinde işlem yapabilir. Okuyucudan gelen zayıf işaretlerle bile çalışabilirler. Üzerlerinde mikroişlemci yaz/oku bellek ve gömülü işletim sistemleri vardır. Yarı aktif etiketler ise üzerlerinde yine aktif etiketler gibi güç kaynağı içermektedir. Ancak güç kaynağını üzerindeki tüm devreyi beslemek için kullanır. Okuyucu ile iletişim esnasında pasif etiket gibi okuyucunun gönderdiği elektromanyetik dalgadan güç alır. Haberleşmede pasif etiket gibi çalıştıklarından çalışma mesafesi düşüktür. Ancak içerisindeki güç kaynağı sayesinde pasif etikete göre veri üzerinde iş yapma yeteneği daha fazladır.

3.1.2. Okuyucu

Okuyucu etiketleri uyarır ve içeriğini okuyup bir veritabanına aktarır [4]. Uygulamaya göre taşınabilir ya da sabit olabilir. Etiketle haberleşmesi esnasında yönetici konumundadır.

3.2. RFID Güvenlik Protokolleri

RFID sistemler diğer tanımlama sistemlerinden farklı olarak yakında olmayı gerektirmediği için içerisindeki özel bilgileri kolaylıkla dışarı vermemelidirler. Özel bilgiler sadece tanımlanmış okuyuculara verilmeli ve uzun süreli bağlantılar kurulmamalıdır. Kanalın güvenliği için aktarılan bilgiler şifrelenmelidir. Bunların yanı sıra okuyucu ve etiket birbirine güvenmeli ve birini taklit etmesi zor olmalıdır. Protokoller okuyucu ve etiketin haberleşmesi ve bir takım kriptografik işlemler yapması sonucu ortaya çıkar. Bu protokollerden bazıları aşağıda tanıtılmaktadır.

3.2.1. İleti-Cevap Protokolleri

Bu protokolde okuyucu rastgele bir sayı üretir ve etikete gönderir. Etikette kendi bir rastgele sayı üretir bunu okuyucudan aldığı rastgele sayıyla birleştirir ve açık ya da kapalı olan bir K anahtarıyla şifreler. Etiket şifrelenmiş sonucu okuyucuya gönderir. Okuyucu sahip olduğu anahtarla şifreyi çözer ve etiketin rastgele ürettiği sayıyı etikete gönderir [5].

3.2.2. Özet (Hash) Fonksiyon Tabanlı Protokoller

Bu protokolde okuyucu ve etiket ortak x değerine sahiptir. Okuyucu etiketi uyarır ve etiket rastgele r sayısını oluşturur. Etiket kendi kimlik bilgisini kullanarak $(r, (ID_{etiket} \parallel H(ID_{etiket})) \oplus f_x)$ hesaplar ve okuyucuya gönderir [6]. H özet fonksiyonudur. "||" dizilerin ardı ardına konmasını, "⊕" XOR işlemini, f_x gizli x bilgisini parametre olarak kullanan sözde rastgele fonksiyondur (pseudorandom function).

3.2.3. XOR Tabanlı Protokoller

Etiket ile okuyucu arasında haberleşmede önceden belirlenmiş başlangıç anahtarı vardır. Bu anahtar her veri yollanacağında özel bir permütasyonu hesaplanıp gönderilecek veri ile XOR'lanıp gönderilir. Permütasyon işlemi her adımda bir önce hesaplanan permütasyon değerini kullanır [7].

3.2.4. Mesafe Sınırlama Protokolleri

Bu protokole ilk örnek Hancke ve Kuhn'un [1] önerdiği protokoldür. Bitirme projesi kapsamında bu protokolle gerçeklemesi yapılacaktır. Protokolün detayı Bölüm 4'te verilmektedir.

4. MESAFE SINIRLAMA PROTOKOLÜ

Bu bölümde Hancke ve Kuhn'un [1] önerdiği mesafe sınırlama protokolünün çalışması anlatılacaktır. Protokol bir etiket ve okuyucu arasında gerçekleşir. Protokol kimlik doğrulama yanında gönderilen bir bitin gidip gelme süresinden faydalanarak etiket ile okuyucu arası mesafeyi belirler. Okuyucunun gönderdiği bir bitin sonucunda etiketin bu biti kullanıp tekrar okuyucuya ulaşma süresi denklem 4.1 ile hesaplanır.

$$t_m = 2t_p + t_d \quad \textbf{(4.1)}$$

Denklem 4.1'de t_d bitin işlem görmesi esnasında oluşan gecikmeler toplamıdır. İçerisinde kapı gecikmeleri, modemlerin modülasyon ve demodülasyon süreleri vardır. t_p elektromanyetik dalganın okuyucudan etikete ya da etiketten okuyucuya ulaşma süresidir. Elektromanyetik dalgalar hava içerisinde ışık hızına yakın hareket eder. Bu nedenle t_p süresi mesafe ile orantılı şekilde değerler alır. Mesafe ve gecikmeler arası ilişki denklem 4.2 ile bulunur. Denklem 4.2'de c ışık hızını göstermektedir.

$$d = \frac{c(t_m - t_d)}{2} \quad \textbf{(4.2)}$$

Okuyucu V, etiket P'ye tek kullanımlık bir N_V bit dizisi gönderir. Denklem 4.3'te gösterilmiştir.

$$V \rightarrow P : N_V \quad \textbf{(4.3)}$$

Okuyucu ve etiket ortak sahip oldukları anahtar K'yi ve tek kullanımlık oluşturulan N_V bit dizisini sözde rastgele fonksiyon h'ye gönderir. Fonksiyon 2n elemanlı bir bit dizisi çevrimi oluşturur. Sonuç olarak hesaplanan bu 2n elemanlı bit dizisi R^0 ve R^1 adında eşit eleman içeren iki alt diziye ayrılır. Denklem 4.4'te gösterilmektedir.

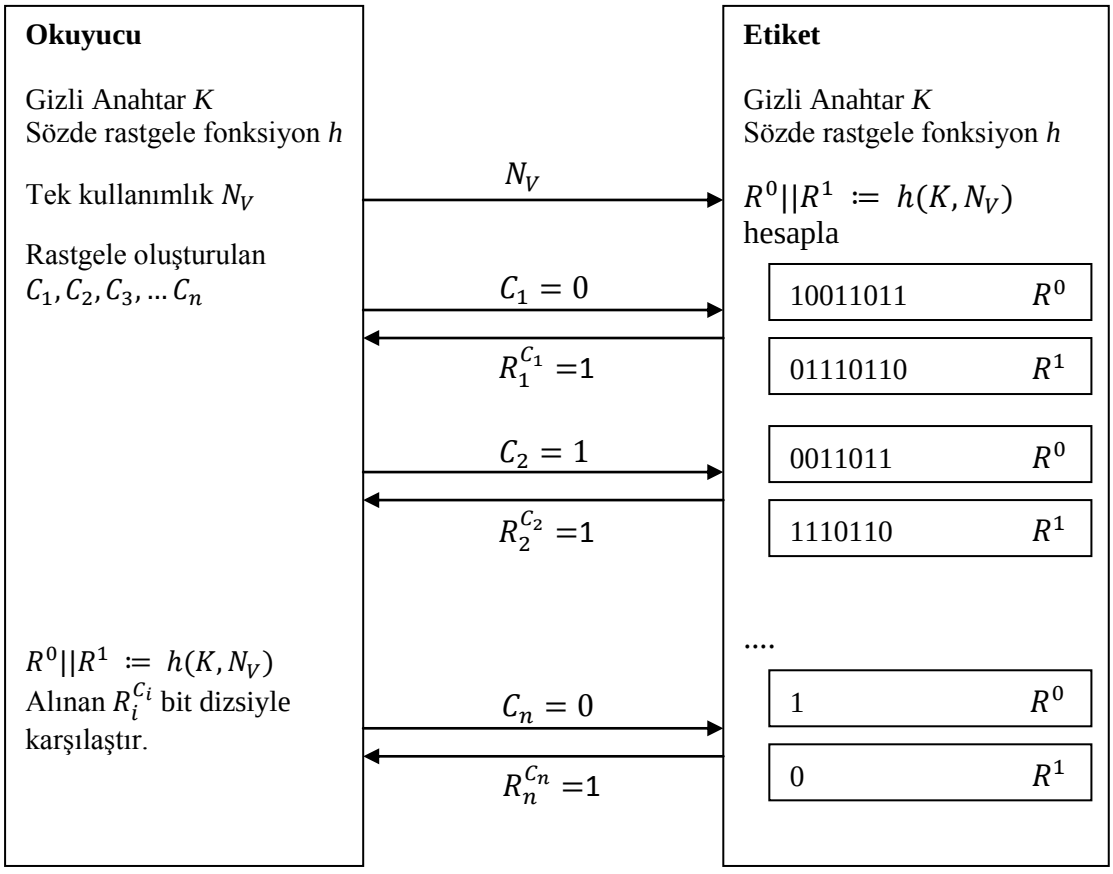
$$R_1^0 R_2^0 R_3^0 \dots R_n^0 || R_1^1 R_2^1 R_3^1 \dots R_n^1 \coloneqq h(K, N_V) \quad \textbf{(4.4)}$$

":=" atama işlemini göstermektedir. Bu işlemin ardından okuyucu n boyutlu tahmin edilemeyen rastgele oluşturulan bir C bit dizisi oluşturur. C dizisinin her elemanı sırayla etikete gönderilir. Etiket ise gelen bit değeri 1 ise R^1 bit dizisinden 0 ise R^0 bit dizisinden 1 bitlik yanıtlar gönderir. Denklem 4.5 ile okuyucunun etikete bit gönderimi, Denklem 4.6 ile etiketin okuyucuya bit gönderimi gösterilmiştir.

$$V \rightarrow P: \ C_i \in \{0,1\} \quad \forall i \ \ 1 \leq i \leq n \tag{4.5}$$

$$P \rightarrow V: \ R_i^{C_i} \in \{0,1\} \quad \forall i \ \ 1 \leq i \leq n \tag{4.6}$$

Her gönderilen C_i bitine karşı gönderilen bit belirli bir t_m zaman aralığından önce geldiyse, etiket belirli bir d mesafesinden uzakta olmayacağı belirlenmiş olur. Mesafe tespitinden sonra alınan bit dizisi ile olması gereken bit dizisi karşılaştırılır.

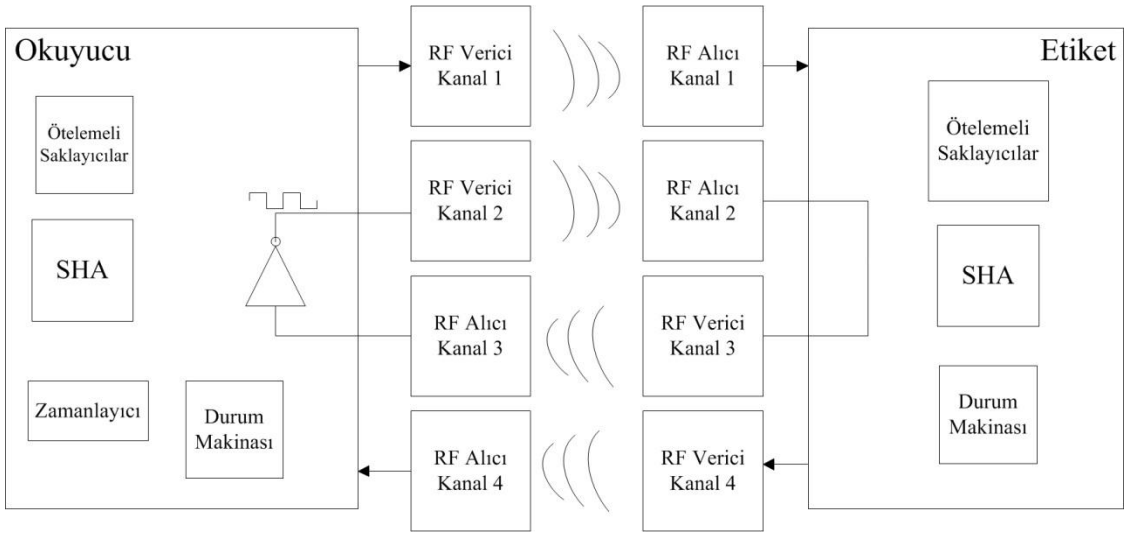


Şekil 4.1 : Protokolün çalışması özeti [1].

5. PROTOKOLÜN GERÇEKLENMESİ

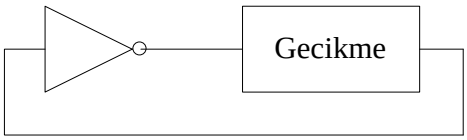
5.1. Devre Tasarımı

3. bölümde anlatılan protokol gerçekleme aşamasında okuyucu ve etiketin bazı koşulları sağlaması gereklidir. Bunlardan en önemlisi okuyucunun göndereceği bit karşılığında etiketin çok hızlı yanıt veren bir asenkron devre olması gereklidir [1]. Eğer senkron devre olursa okuyucunun göndereceği bitin, etiketin senkron saat işaretini hep beklemek durumunda kalacak aradaki mesafe bilgisi hep sabit algılanacaktır. Anlatılan protokolde arada iki kanal bulunmaktadır. Gerçekleme esnasında iki kanal bulunursa, devre de asenkron tasarlanacağı için art arda gelen 00...0 bit dizileri ve 11...1 dizileri asla fark edilemez. Bu durumu çözmek için dört kanal kullanan bir sistem önerildi. h fonksiyonu gerçeklemek için SHA blokları kullanıldı. Şekil 5.1'de önerilen sistemin modeli görülmektedir.



Şekil 5.1 : Protokol gerçeklemesi için önerilen devre.

Şekil 5.1'deki devrede Kanal 2 ile Kanal 3 arasındaki gecikmeden faydalanarak frekansı mesafe ile değişen bir ring osilator [3] kurulmuştur. Basit ring osilator yapısı Şekil 5.2'de görülmektedir.



Şekil 5.2 : Ring Osilator [8].

Ring osilatorün frekansı mesafe ile ters orantılıdır. Ring osilatorün frekansı denklem 5.1 ile hesaplanır.

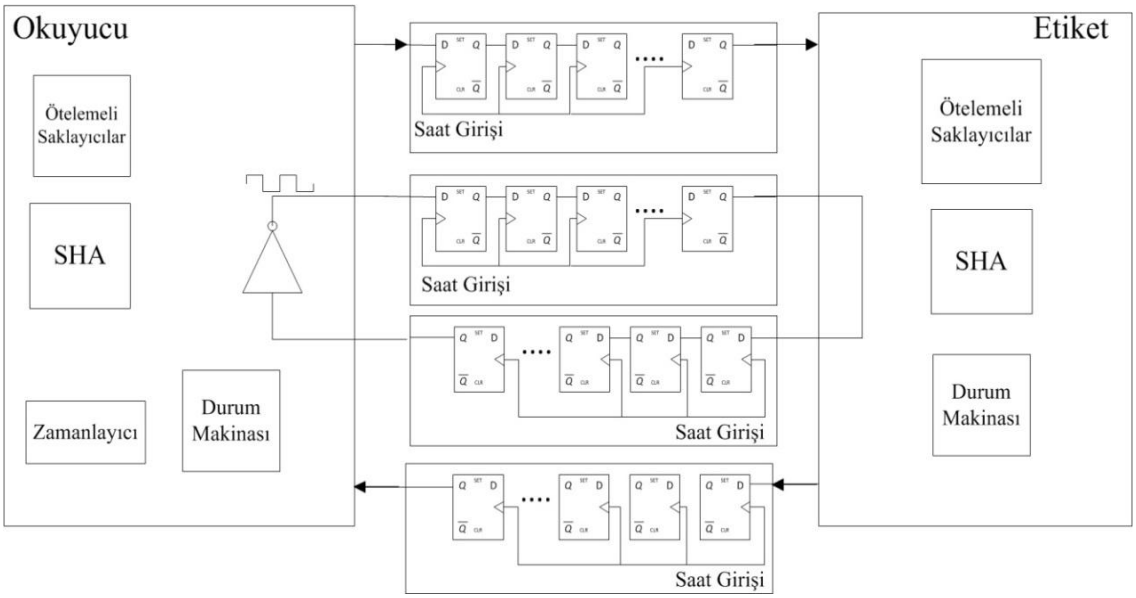
$$f_{osc} = \frac{1}{2(2t_p + t_d + t_{gd})} \quad \textbf{(5.1)}$$

t_p ve t_d süreleri bölüm 3'te anlatılmıştır. Buna ek olarak t_{gd} ise DEĞİL kapısının gecikmesidir. Kurulan osilator etiket ve okuyucunun ötelemeli saklayıcılarının saat işareti olarak verildi. Böylece bitlerin gönderilme süreleri arası ilişki mesafe ile orantılı şekilde değişmesi sağlandı. Mesafe tespiti içinse n sayıda bit aktarımı sonucu geçen süre bilgisinden faydalanıldı.

Önerilen devrenin gerçeklemesinde bazı problemler ortaya çıkmıştır. En önemli problem, RF modemlerin modülasyon sürelerinin sabit olmaması. Bu sürenin sabit olmamasından dolayı osilator frekansı değişkenlik gösterir ve mesafe yanlış algılanır. Bir diğer problem aradaki mesafenin kısa olmasından dolayı osilator frekansını gigahertz mertebelerine ulaşmasıdır. Kullandığımız FPGA gigahertz mertebesinde yanıt veremediği için bu sistem gerçeklenememektedir. Protokolün çalışmasını göstermek amacıyla RF modemler çıkartılıp yerlerine D tipi flip-floplardan oluşan ötelemeli saklayıcı blokları konulup sistem FPGA içerisinde gerçeklendi. Şekil 5.3'te alternatif sistem görülmektedir.

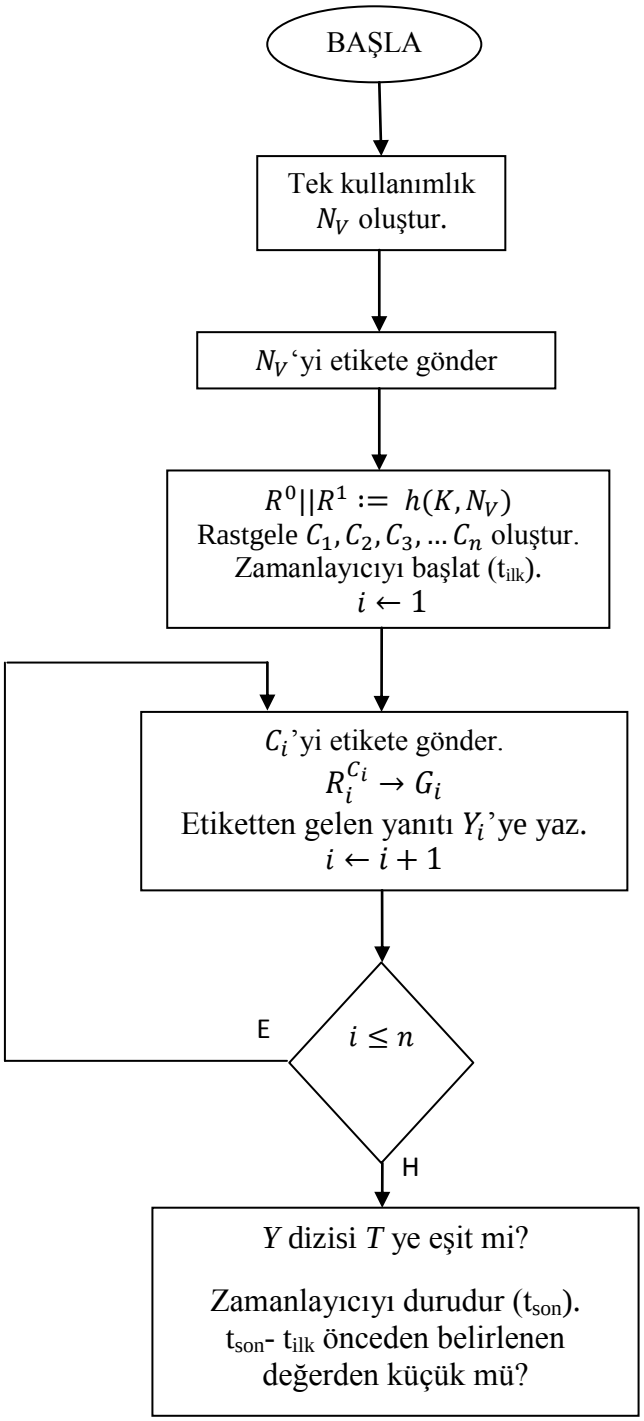
Gerçeklenen devrede okuyucunun akış şeması Şekil 5.4'te, etiketin akış şeması Şekil 5.5'te gösterilmektedir. Bu akış şemaları esas alınarak devre Verilog HDL dilinde yazıldı. Sentezleme sonucunda FPGA içerisine gömüldü. FPGA üzerinde çalışmasını test etmek için LED'lerden faydalanıldı. Ancak nasıl çalıştığının anlaşılması için devrenin davranışsal benzetimi yapıldı. Devre ortamında gecikmeler bir birine bağlı D tipi flip-floplarla yani öteleyici saklayıcılarla

gerçeklenmişti. Aradaki mesafe değişikliği hattı modellemek için konulan bu D tipi flip-flop sayılarının değişmesiyle belirlendi.

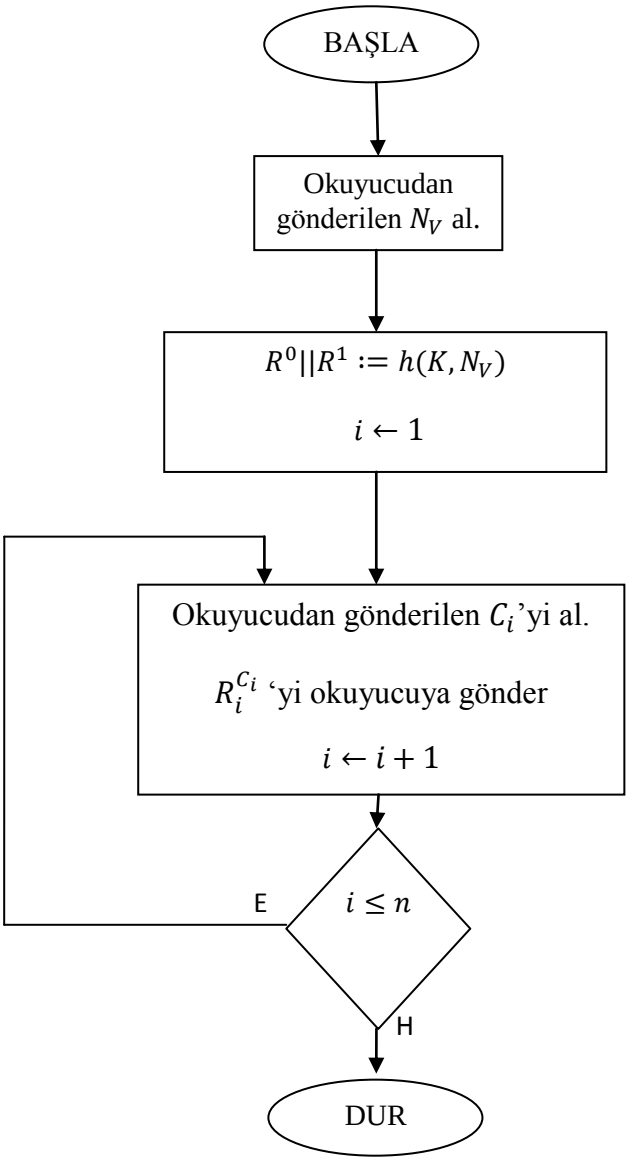


Şekil 5.3 : Protokol Gerçeklemesi İçin Önerilen Devre.

Şekil 5.4'te bütün öteleme işlemleri etiket ile okuyucu arasında oluşturulan osilatör ile tetiklenmektedir. Okuyucu rastgele oluşturulan C bit dizisi etikete gönderilmeye başlandığında, okuyucu kendi hesapladığı R^0 ve R^1 dizilerini kullanarak etiketten gerçekten alması gereken bitleri G saklayıcısına yazmaktadır. Etiketten alınan bitler Y ise saklayıcısına yazılır. Etiket ve okuyucu kaç bit aktaracaklarını önceden bilmektedir. Aktarım tamamlandığında zamanlayıcı durdurulur. Önceden belirlenmiş bir süre ile zamanlayıcının değeri karşılaştırılır. Ardından olması gereken bit dizisi ile alınan bit dizisi karşılaştırılır ve protokol tamamlanır. Zamanlayıcının bit aktarımından önce başlatılmasının en önemli sebebi, bir bit transferi arasından geçen süre çok küçük olmasından dolayı bunu ölçmesi gereken zamanlayıcının frekansı çok yüksek olmasıdır. Ancak bit paketinin toplam süresine bakılırsa daha düşük bir frekans kullanarak aynı süre ölçülmüş olur.



Şekil 5.4 : Okuyucu akış şeması.

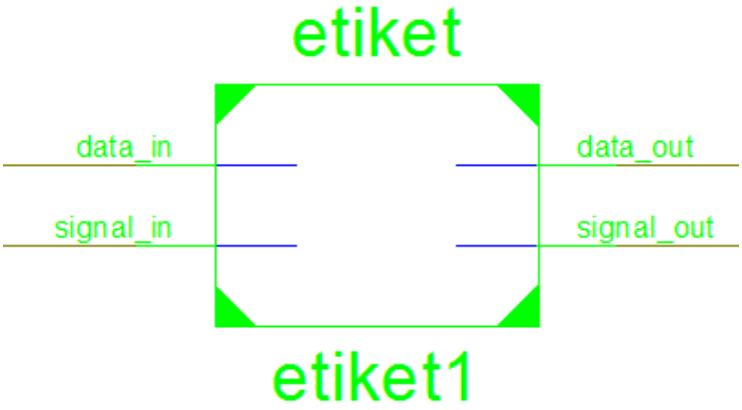


Şekil 5.5 : Etiket akış şeması.

Etikette okuyucu ile benzer şekilde tüm öteleme işlemlerini ikisi arasında oluşturulan osilatör ile yapmaktadır. Etiket okuyucunun gerçek okuyucu mu yoksa sahte okuyucu mu olduğuyla ilgili bir doğrulama yapmamaktadır.

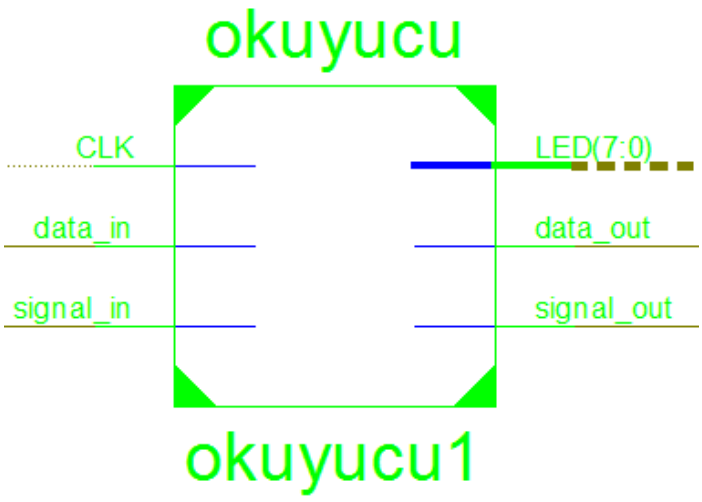
5.2. Tasarlanan Devrenin Test Edilmesi

Devrenin sentezi sonucunda etiket devresi için elde Şekil 5.6'daki blok elde edilmiştir. Devre dört kanaldan oluşmaktadır; signal_in ve signal_out osilatör oluşturmak için, data_in ve data_out bit aktarımı için kullanılacaktır.



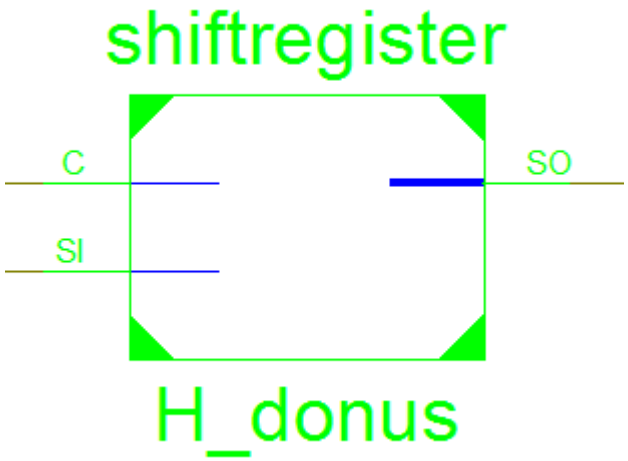
Şekil 5.6 : Etiket Devresi Bloğu.

Şekil 5.7'de okuyucunun devre bloğu görülmektedir. Okuyucuda etikette kullanılan dört kanalın dışında CLK saat girişi ve LED[7:0] çıkışı vardır. CLK saat girişi okuyucu içerisinde bit aktarım süresini ölçen zamanlayıcı için kullanılmaktadır. LED[7:0] ise devrenin çalışması esnasında test amacıyla kullanılmıştır.



Şekil 5.7 : Okuyucu Devresi Bloğu.

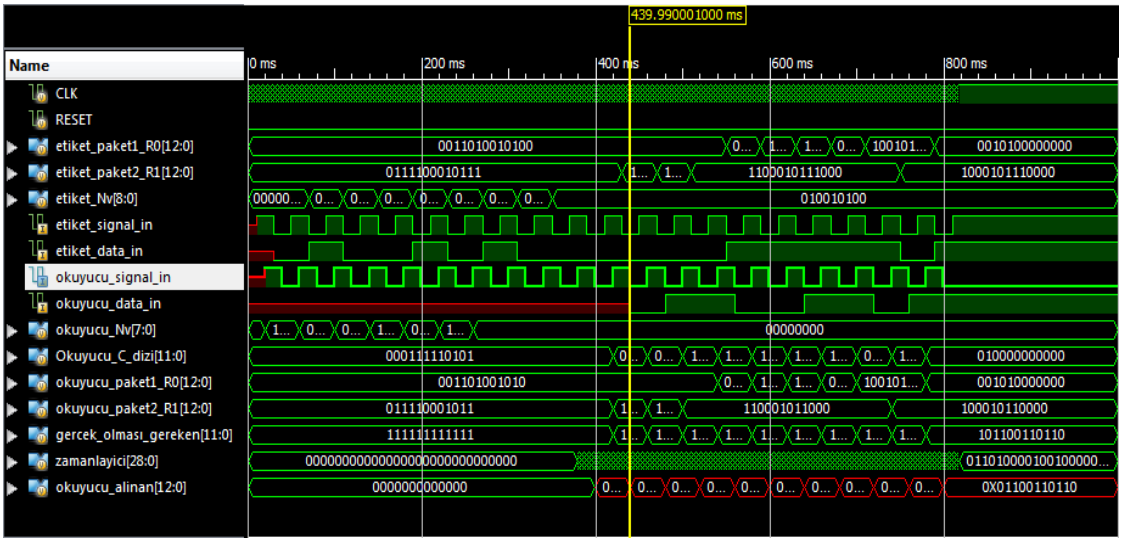
Şekil 5.8'de hattı modellemek için kullanılan ötelemeli saklayıcı görülmektedir. C saat girişi SI bir bitlik giriş ve SO bir bitlik çıkıştır. Bu bloktan dört kanalı da modellemek içinde toplam dört adet kullanılmıştır.



Şekil 5.8 : Hat Modellemede Kullanılan Ötelemeli Saklayıcı Bloğu.

Benzetim sonuçları Şekil 4.8'de görülmektedir. Etiket ile okuyucu arasında signal_in ve signal_out pinleri osilasyonu başlatmıştır. İlk olarak Okuyucunun N_V bit dizisini gönderdiği görülmektedir. Gönderilen bitleri etiket etiket_signal_in girişinin düşen kenarında okumaktadır. Gönderim esnasında okuyucu saklayıcıyı sürekli ötelediği için bittiğinde okuyucu_Nv sıfır ile dolmaktadır. Etiket N_V'yi aldıktan sonra okuyucu C bit dizisini göndermeye başlamaktadır. Aynı zamanda C bit dizisinin gönderimi başladığı anda zamanlayıcı saymaya başladığı görülmektedir. paket_R0 ve paket R1 ise R^0 ve R^1 saklayıcılarını göstermektedir. Her gönderim esnasında hangi paketten bit gönderildiyse ilgili bit paketi bir bit ötelenir. Benzetim sonucunda 400 ms'den sonra C bit dizisinin aktarımı başlar ve okuyucu etiketten ilk yanıt biti 440 ms'de (sarı ile işaretli kısım) alınır. Okuyucu da bitlerin alımını saat işaretinin düşen kenarında gerçekleştirmektedir. Aynı zamanda okuyucu gercek_olmasi_gereken saklayıcısında olması gereken sonucu C dizisini gönderimi esnasında hesaplar ve bit aktarımı sonunda etiketten aldığı bit dizisi ile karşılaştırır. Şekil 5.8'de görüldüğü gibi benzetimin sonunda gercek_olması_gereken saklayıcısı ile okuyucu_alınan saklayıcısının son 10 bit dizilimleri aynıdır. Fazlalık bitler durum geçişindeki beklemelerden dolayı oluşmuştur. Zamanlayıcı ise aktarım

tamamlandığında kendinin durdurmuştur. Zamanlayıcı 50MHz ile çalıştığı için sayacının boyu uzundur. Mesafeyi bulmak için zamanlayıcının anlamlı 6 biti ile belirlenen eşik değeri karşılaştırılır. Eğer mesafe değişiminde daha hassas olmak istenirse bu bit sayısı artırılabilir.



Şekil 5.9 : Benzetim Sonuçları.

6. SONUÇLAR

Devre gerçeklemesi sonucunda protokolün nasıl gerçeklenebileceğiyle ilgili bir fikir verilmiştir. RF modemler ve hattın, ötelemeli saklayıcı olarak kullanılarak modellenmesi aradaki mesafenin flip-flop sayısı ile orantılı olduğunu göstermektedir. Bu durumda mesafe ölçüm hassasiyeti fark edilebilecek en küçük flip-flop sayısıdır. En az sayıda flip-flop sezme kabiliyeti ötelemeli saklayıcının frekansı ve zamanlayıcının frekansı ile ilgilidir. Ötelemeli saklayıcının frekansı zamanlayıcının frekansından çok küçükse bütün flip-floplar zamanlayıcı tarafından fark edilir. Gerçeklenen devrede hat frekansı 10 KHz, zamanlayıcı frekansı 50 MHz'dir.

Gerçeklenen protokolde etiket mesafe bilgisini yanlış gösterip okuyucuyu aldatabilir. Bunu kendi içerisinde bulunan osilatorü sinyal çıkışlarına vererek sağlar. Mesafesini değiştirmek için içerisinde bulunan osilatorün frekasını ayarlar. Okuyucu toplam süreye baktığı için kolaylıkla aldanacaktır. Etiket mesafe olarak okuyucu altabilir ancak onaylanması için doğru bilgeride tahmin etmesi gereklidir. Bunu yapmak için sürekli bit tahmininde bulunur her biri biti doğru tahmin etmesi ½ olasılığa sahiptir. Bit paketi kıyaslandığı için bu olasılık $(1/2)^n$ olur. Bit paketi uzunluğu artırıldıkça bu olasılık düşer. Gerçeklenen protokolde etiket okuyucuyunun doğruluğu hakkında bir test yapmamaktadır.

Okuyucu bazı durumlarda etiketin gerçek etiket olmasına rağmen yanlış etiket gibi algılayabilir. Bu duruma örnek etiket doğru mesafede olur, ancak bit aktarımı esnasında gürültüden dolayı bit yanlış aktarılabilir. Diğer bir durum ise okuyucu etiket ile haberleşme esnasında etiketin ani konum değişiklikleri arada oluşan osilatörün periodunu değiştirebilir. Bu durum toplam sürede değişikliğe sebep olur ve konum olarak yanlış algılanabilir.

KAYNAKLAR

[1] **Hanche G., Knuh M.,** An RFID Distance Bounding Protocol, Proceeding of SECURE COMM'05, pages 67-73, IEEE Computer Soceity, 5-9 September 2005.

[2] **UG230, 2006,** Spartan 3E Starter Kit Board User Guide

[3] **K. Finkenzeller**, "RFID Handbook", Carl Hanser Verlag, Münih, 2003.

[4] **D. E. Brown,** RFID Implementation, McGraw-Hill, New York, 2007.

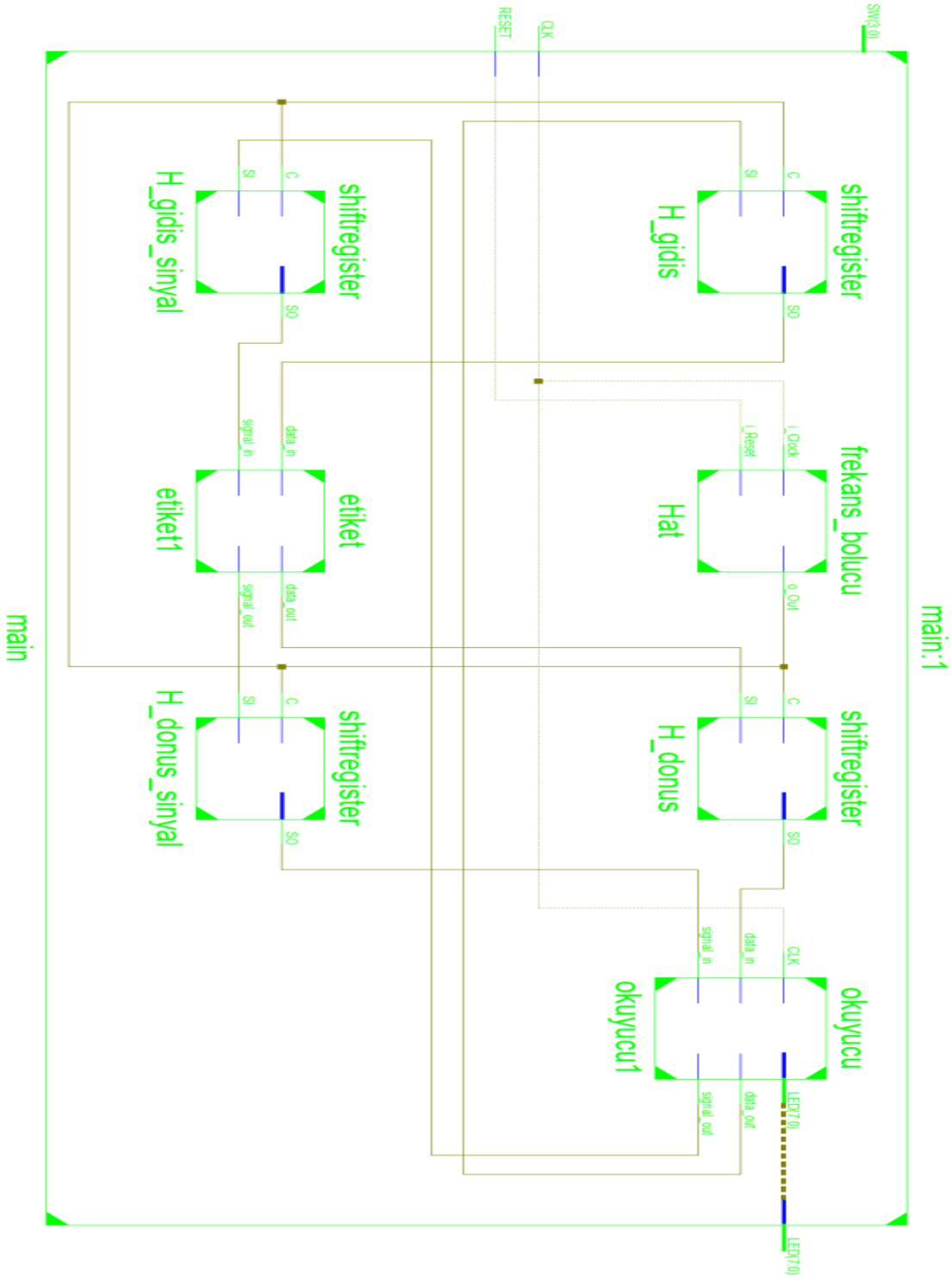
[5] **M. Feldhofer, S. Dominikus, and J. Wolkerstorfer,** "Strong Authentication for RFID Systems Using the AES Algorithm", in Proceedings of 6th International Workshop on Cryptographic Hardware and Embedded Systems (CHES), Cambridge, MA, USA, August 11-13, 2004, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, volume 3156/2004, pages 85-140.

[6] **S. A. Weis,** "Security and Privacy in Radio-Frequency Identification Devices", Master Thesis, MIT, Massachusetts, 2003.

[7] **S. Stadlober,** "An Evaluation of Security Threats and Countermeasures in Distributed RFID Infrastructures", Master Thesis, Institut f¨ur Informationssysteme and Computermedien Technische Universitat Graz, July 2005.

[8] **T. H. Lee,** The Design of CMOS Radio-Frequency Integrated Circuits. Cambridge, U.K.: Cambridge Univ. Press, 1998

EKLER



Şekil A.1 : Devrenin Sentez Sonucu Şeması.