

VHDL Kullanarak FPGA ile Yüksek Kapasiteli Tam Çıkarıcı Devre Tasarımı

Kenan Baysal¹, Deniz Taşkın², Eser Sert³, Nursen Topçubaşı⁴

¹ Namık Kemal Üniversitesi, Hayrabolu Meslek Yüksekokulu, Bilgi Yönetimi Programı, Tekirdağ

² Trakya Üniversitesi, Bilgisayar Mühendisliği, Edirne

³ Trakya Üniversitesi, Edirne Teknik Bilimler Meslek Yüksekokulu, Elektronik Teknolojisi Programı, Edirne

⁴ Okan Üniversitesi, İstanbul

kbaysal@nku.edu.tr, deniztaskin@trakya.edu.tr, esersert@trakya.edu.tr, nursen@trakya.edu.tr

Özet: FPGA’ye ait donanım yapıları artan kapasiteleri sayesinde bilgisayar ve elektronik alanlarında, ihtiyaca göre uygun gömülü sistem uygulamalarında yaygın olarak kullanılmaya başlamıştır. Bilgisayar sistemlerinin artan işlem kapasitelerine cevap vermek için, işlemci tasarımları gerçekleştirilmektedir. Ancak yüksek kapasiteli paralel aritmetik işlemler için gerçekleştirilen tasarımlar, hız söz konusu olduğunda ihtiyaca cevap verememektedir. FPGA ile paralel aritmetik işlemler gerçekleştirmek için ihtiyaca özgü aritmetik üniteler donanım seviyesinde tasarlanabilir ve uygulanacak sistem üzerine kolaylıkla uygulanabilir. Bu çalışmada VHDL kullanılarak FPGA ile yüksek kapasiteli çıkarıcı devresi tasarlanmıştır.

Anahtar Sözcükler: FPGA, VHDL, Yüksek Kapasiteli Aritmetik Ünite, Tam Çıkarıcı.

High Capacity Subtractor Circuit Design with FPGA Using VHDL

Abstract: FPGA hardware structures are started to using extensively by increasing capacity, appropriate to needs of embedded system implementations on computer and electronic fields. Processor designings are performing to answer the purposes of increasing processing capacity of computer systems. But performed designings cannot able to answer the purpose when it comes to speed of high capacity parallel arithmetic processes. Arithmetic units, which are dedicated to purpose, can be design for perform parallel arithmetic processes with FPGA at hardware level and can be easily adapted to implemented system. In this study, high capacity subtractor circuit designed with FPGA, using VHDL

Keywords: FPGA, VHDL, High Capacity Arithmetic Unit, Full Subtractor.

1. Giriş

Çıkartma işlemi, bilgisayar mimarisi içerisinde, temelde toplama işleminin 2’ye göre tümleyenini alma mantığına göre gerçekleştirilen bir aritmetik işlemidir [6]. Toplama işlemi, bilgisayar aritmetiğinde kullanılan işlemlerin temelini oluşturur. Günümüz bilgisayar mimarisinde bu işlemler, özellikle yüksek kapasiteli veri işleme sırasında, oldukça uzun zaman süreçlerinde gerçekleştirilebilmektedir.

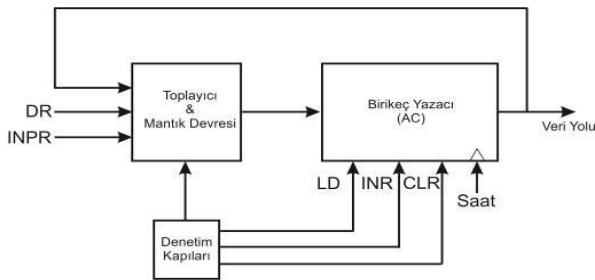
FPGA donanım yapısının yüksek kapasitesi ve kolay

programlanabilir olması sayesinde, bilgisayar işlemcilerinin sunduğu standart veri işleme kapasitesi yerine, ihtiyaçlar doğrultusunda aritmetik üniteler tasarlanabilir. Karmaşık algoritmaların, FPGA uygulamalarının paralel işlem kabiliyetleri ve gerekli donanım kaynakları sayesinde performansı artırılabilir. [2]

Gerçekleştirilen bu çalışmada FPGA donanımını programlamak için VHDL kullanılmıştır. VHDL (Very High Speed Integrated Circuit - Hardware Description Language), yüksek hızlı tümleşik donanım tanımlama dilidir. Buna paralel olarak günümüzde yaygın biçimde kullanılan diğer donanım tanımlama dili olarak Verilog HDL sayılabilir. Bu çalışmada kullanılan VHSIC HDL kod yapısı ADA programlama dilini baz alan Pascal sınıfında bir dil olarak sayılabilir. Verilog HDL ise daha ziyade C programlama diline yakın bir yapıdadır. [3]

2. Bilgisayar Aritmetiğinde Çıkartma İşlemi

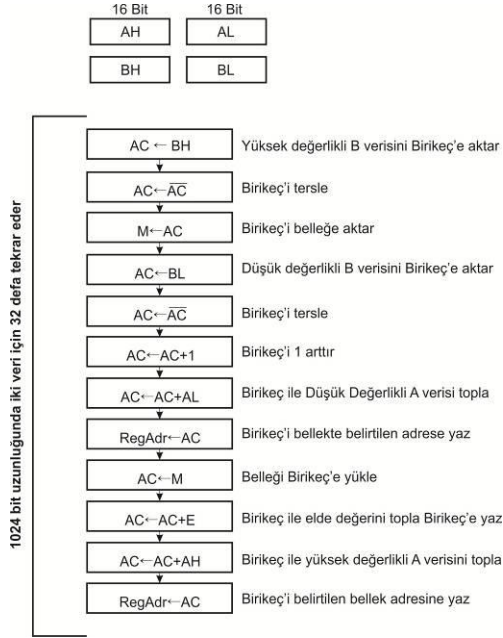
Standart bir bilgisayar işlemcisinde, Birikeç Yazacı (Akümülatör-AC), Toplayıcı ve Mantık devresi ile Denetim kapılarından oluşan bir yapı mevcuttur. İşlenecek veri kaydedicilerden veri yolu ile buraya aktarılarak işlem süreci gerçekleştirilir. [1]



Şekil 16. Birikeç Yapısı [1]

32 bit bir işlemciye sahip olan standart bir bilgisayar, yazılımsal olarak birçok adımda çıkarma işlemini gerçekleştirir.

Şekil 2’de görüldüğü gibi işlemcinin veri kapasitesi ile aynı veri boyutu büyüklüğüne sahip iki değer dahi çıkarma işlemi birçok adımda gerçekleştirilebilmektedir. Şifreleme, şifre çözme veya grafik işleme konularında günümüzde kullanılan veri boyutları 2048 bit ve üzerine ulaşmıştır. Aynı veri işleme kapasitesine sahip bir işlemci, yazılımsal olarak gerçekleştirilen bu çıkartma işlemini

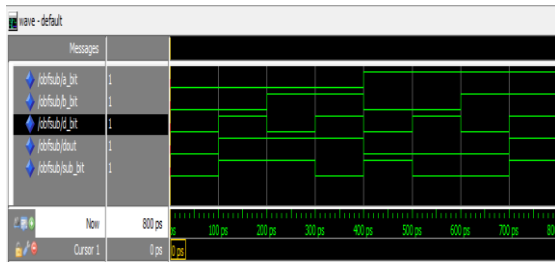


Şekil 2. 32 bitlik iki veriye çıkarma işlemi uygulama algoritması [1]

1024 bit uzunluğundaki iki veri için, verileri paketlere bölerek işleyebilmesi için 32 defa tekrar etmesi gerekir.

3. Yüksek Kapasiteli Çıkartma Devresi Tasarımı

Yüksek kapasiteli verileri kısa sürede işleyebilmek, şekil 2’de ki örnekte de görüldüğü gibi, mevcut

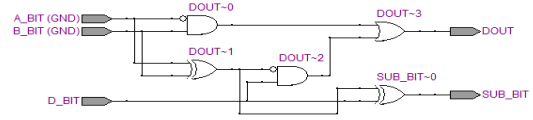


standart sistemler ile mümkün değildir. FPGA kullanılarak bu sorunu aşmak mümkündür. FPGA’nın yapısı sayesinde, istenilen veri boyutuna göre tam çıkarıcı tasarlanabilir.

3.1 VHDL ile Bir Bit Tam Çıkarıcı Devresi

Yüksek kapasitede bir çıkarıcı devresi tasarlamak için öncelikle bir bit boyutunda bir tam çıkarıcı devresi tasarlamak gerekmektedir. Bir sonraki aşamada, tasarlanan bu bir bitlik tam çıkarıcı devresini genişleterek, istenilen boyutta bir çıkarıcı devresi elde edilebilir.

Şekil 3. Bir Bit Tam Çıkarıcı devresinin ModelSim



benzetim sonuçları

Şekil 4. Bir Bit Tam Çıkarıcı Devresi RTL görünümü

Şekil 4’te tasarlanan bir bit tam çıkarıcı devresinin Quartus programındaki RTL devre görüntüsü görülmektedir.

Şekil 5’te bir bit tam çıkarıcı devresinin VHDL kodları görülmektedir. Bu kod yapısı içerisinde, “entity” kısmının altında A_BIT ve B_BIT olarak devrenin veri girişleri tanımlanmıştır. D_BIT olarak önceki devreden borç girişi, DOUT olarak bir sonraki devreye devreden borç çıkışı tanımlanmıştır. SUB_BIT olarak devrenin işlem sonucu çıkışı tanımlanmıştır. “Process” yapısı altında devrenin girişlerinin işlem davranışları tanımlanmış ve çıkışlara aktarılmıştır.

Şekil 3’te bir bit tam çıkarıcı devresinin ModelSim programında benzetim sonuçları görülmektedir.

3.2 VHDL ile “n” Bit Çıkarıcı Devresi

İstenilen veri boyutunda bir çıkarıcı devresi oluşturulabilmesi için, bir önceki kısımda tasarlanan bir bit çıkarıcı devresi, VHDL kod içerisine, şekil 7’de görüldüğü gibi, bileşen olarak aktarılıp “n” defa tekrar edilir.

VHDL kod yapısı içerisinde, devre giriş ve çıkışlarının tanımlandığı entity kısmının altında “SZ” değişken olarak tanımlanır.

Şekil 8’de verilen VHDL kodlarının içinde “SZ” değişken değeri 63 olarak verilmiştir. Bu değer ile tam çıkarıcı devresi 64 bit olarak ayarlanmıştır. Bu değer, kullanılacak olan FPGA kapasitesine bağlı olarak istenilen seviyeye ayarlanarak yüksek kapasiteli çıkarıcı devresinin kapasitesi artırılabilir.

[2] Munoz D. M., Sanchez D. F., Llanos C. H., Rincon M., “Tradeoff FPGA Design of a Floating-point Library for Arithmetic Operations”, *Journal Integrated Circuits and Systems*, v.5/n.1:43(2010)

[3] Smith D. J., “VHDL & Verilog Compared & Contrasted – Plus Modeled Example Written in VHDL, Verilog and C”, *33rd Design and Automation Conference* (1996)

[4] Baumann C., “Field Programmable Gate Array (FPGA)”, *Embedded System Architecture*, University of Innsbruck (2010)

[5] Diamond R., Racaniere S., Pell O., “Accelerating large-scale HPC Applications using FPGA’s”, *20th IEEE Symposium on Computer Arithmetic*, (2010)

[6] Kulisch U., “Computer Arithmetic and Validty, Theory, Implementation and Applications”, *De Gruyter Publishing*, 201 (2008)

[7] Parandeh-Afshar H., Verma A. K., Brisk P., Jenne P., “Improving FPGA Performance for Carry-Save Arithmetic”, *IEEE Transactions on very Large Scale Integration (VLSI) Systems*, v.18/n.4, 578 (2010)