

PAPER DETAILS

TITLE: Assembler Design for BZK.SAU.FPGA Micro Computer Architecture

AUTHORS: Halit Öztekin, Ali Gülbag, Feyzullah Temurtas

PAGES: 1-9

ORIGINAL PDF URL: <https://dergipark.org.tr/tr/download/article-file/313191>

Assembler Design for BZK.SAU.FPGA Micro Computer Architecture

Halit Öztekin¹, Ali Gülbağ², Feyzullah Temurtaş³

¹Bozok Üniversitesi, Bilgisayar Mühendisliği, Yozgat

²Sakarya Üniversitesi, Bilgisayar Mühendisliği, Adapazarı

³Bozok Üniversitesi, Elektrik Elektronik Mühendisliği, Yozgat

Abstract: It is necessary to constitute a memory organization and a memory map in the microcomputer architecture in order to allow operation systems and assembler to work. The BZK.SAU.FPGA microcomputer architecture assembler instruction set consists of 59 instructions and uses 6 different addressing modes. This work demonstrates how to design an assembly language from scratch on the BZK.SAU.FPGA microcomputer architecture. The assembler program uses the Brute-force Search Algorithm to convert the user source program to machine code.

Keywords: BZK.SAU.FPGA, Assembler Design

BZK.SAU.FPGA Mikro Bilgisayar Mimarisi için Assembler Tasarımı

Özet: İşletim sistemleri ve assembler çalışmalarına olanak sağlamak amacı ile mikro bilgisayar mimarisinde bir bellek organizasyonu ve bellek haritası oluşturulmak gerekmektedir. BZK.SAU.FPGA mikro bilgisayar mimarisi assembler komut kümesi 59 adet komuttan oluşmakta olup 6 farklı adresleme modunu kullanmaktadır. Bu çalışmada BZK.SAU.FPGA mikro bilgisayar mimarisi üzerine sıfırdan bir assembly dili nasıl tasarlanacağı gösterilmiştir. Assembler programı, kullanıcı kaynak programını makine koduna dönüştürürken Brute-force Search Algoritması' nı kullanılmıştır.

Anahtar Kelimeler: BZK.SAU.FPGA, Assembler Tasarımı

1 Giriş

Bilgisayar ve Elektronik Mühendisliği gibi bilim dallarında önemli bir yer teşkil eden Bilgisayar Mimarisi ve İşletim Sistemi derslerinde teorik olarak işlenen kavramların pratiğe dönüştürülmesi önemlidir [1–5]. IEEE bilgisayar topluluğu ve bilgisayar bilimleri müfredatı düzenleme birliği tarafından tavsiye edilen müfredat içeriğinde yer alan “Tüm öğrenciler pratik çalışma ile teorik bilgiyi bir bütün halinde öğrenmek zorundadırlar” ifadesi derslerin işlenmesinde pratik çalışmaların gerekliliğine vurgu yapılmaktadır [6]. Bu amaçla son yıllarda, alanda programlanabilir kapı dizileri (FPGA- Field Programmable Gate Arrays) geliştirme kartları kullanılarak teorik bilginin uygulamaya geçirilmesine dair yeni bir yaklaşım getirilmiştir [7, 8].

BZK.SAU.FPGA mikro bilgisayar mimarisi, FPGA geliştirme ortamında tasarlanmış modüller özellikle mantık kapı tabanlı bir mimarisidir. Bilgisayar mimarisinin modülerlik özelliği ile kullanıcılara sisteme müdahil olma avantajını getirerek kullanıcının sistemdeki bir bileşenin yerine kendi tasarımını ekleyerek sistemin işleyişinde herhangi bir olumsuzluğa neden olmadan tasarım yapma olanağı sağlamıştır [7,8].

İşletim sistemleri ve assembler çalışmalarına olanak sağlamak amacı ile mikro bilgisayar mimarisinde bir bellek organizasyonu ve bellek haritası oluşturulmak gerekmektedir. BZK.SAU.FPGA mikro bilgisayar mimarisi assembler komut kümesi 59 adet komuttan

* Corresponding author; Tel.: +(90) 555 8961679 , E-mail: halit.oztekin@bozok.edu.tr

oluşmakta olup 6 farklı adresleme modunu kullanmaktadır [7-9]. Bu çalışmada BZK.SAU.FPGA mikro bilgisayar mimarisi üzerine sıfırdan bir assembly dili nasıl tasarlanacağı gösterilmiştir. Assembler programı, kullanıcı kaynak programını makine koduna dönüştürürken Brute-force Search Algoritması ve FPGA geliştirme ortamı olarak Altera DE2 kullanılmıştır.

2 BZK.SAU.FPGA Mikro Bilgisayar Mimarisi

Bu çalışmada BZK.SAU.FPGA Mikro bilgisayar mimarisi kullanılmıştır. BZK.SAU.FPGA Mikro bilgisayar mimarisinde 640×480 piksel çözünürlüğüne sahip VGA tipinde bir ekran tercih edilmiştir. Mimarinin sahip olduğu bellek boyutu kısıtlı olduğundan ve ABC80 [10], Apple I ve Apple II [11] gibi sistemlerde kullanılan ekran boyutunun 40 sütun×24 satır olması nedeniyle mikro bilgisayar sisteminde de 40×24 boyutunda bir ekran kullanımına gidilmiştir. Sistemde giriş birimi olarak diğer iletişim metotlarına nazaran daha basit olan PS/2 klavye tercih edilmiştir. 102 tuşlu geleneksel bir PS/2 klavyede bulunan fonksiyon tuşları hariç bütün tuşların kullanımına imkan tanınmıştır. Tuşlara kod atanması işleminde literatürde yer alan ASCII kod tablosundan faydalanılmıştır. Mimarinin adres ve veri yolu genişliği genelde 16-bit ve sistem ana bellek biriminin 64KB kapasiteye sahiptir. Sistemin büyük bir kısmının tasarımı şematik olarak, başka bir deyişle lojik kapı seviyesinde tasarım yapılmıştır. Sistemin inşa edildiği donanım birimi olan FPGA kartlarını programlamak için kullanılan Quartus sistem geliştirme aracı, istendiği takdirde şematik tasarımları FPGA kartları programlamak için kullanılan VHDL, Verilog gibi yazılımsal dillere de dönüşümü yapabilmektedir. Sistemin genel yapısının ve projenin eğitimsel amaçlı olmasından dolayı basitliğin temel alınması açısından işlemci mimarisi tek komut-tek veri esasına dayalı SISD mimarisidir.

Bir bilgisayar mimarisinin tam olabilmesinin [12] şartlarından biri olan depolama birimi olarak bu tasarımda hem mevcut kaynaklar içinde bulunması hem de kontrolünün kolaylığı sebebiyle Flash bellek kullanılmıştır. Hem ana bellek hem de depolama biriminde bellek kelime yerleşim düzeni big-endian yapısındadır. Bir bilgisayar mimarisinin tam olup olmaması sahip olduğu komut kümesine göre şekillenmektedir [12]. Eğer komut kümesi, aritmetik ve mantıksal işlemleri yapabilen, bellek ve kaydediciler üzerinde veri akışını sağlayabilen, işlemciyi kontrol edebilen ve giriş-çıkış birimleri ile alışveriş yapabilen komutlara sahip olması durumunda bir bilgisayar mimarisi “tam” özellikli bir yapıya kavuşmuş olur. Bundan dolayı mimaride kullanılan komutlar hem tam bir bilgisayarda olması gereken temel komutlardan oluşmakta hem de komutların tasarımında basitlik temel alınarak oluşturulmuş komutlardır. Ayrıca kullanılan komutların mikro işlem adım sayıları birbirinden farklı olduğundan mikro bilgisayar mimarisi komutları CISC yapıdadır.

Tasarımda mimarinin kullandığı toplam 10 adet genel ve özel amaçlı kaydedici olup, bunlardan giriş ve çıkış kaydedicileri 8 bit diğerleri ise 16 bit genişliğindedir. Hem literatürde eğitimsel amaçlı bilgisayar mimarilerinde hem de ticari mikro işlemci komut kümelerinde yaygın olarak kullanılan inherent, immediate, direct, indirect, index ve relative olmak üzere 6 adet temel adresleme modu kullanılmaktadır. Komutların mikro işlem adımlarında donanımsal kontrol yapısı kullanılmıştır. Mimarinin aritmetik ve lojik birimi sadece 16-bit uzunluğundaki tamsayılar üzerinde işlem yapabilirken negatif sayıların gösterimi için literatürde sıkça kullanılan 2’ye tümleme mantığı kullanılmaktadır.

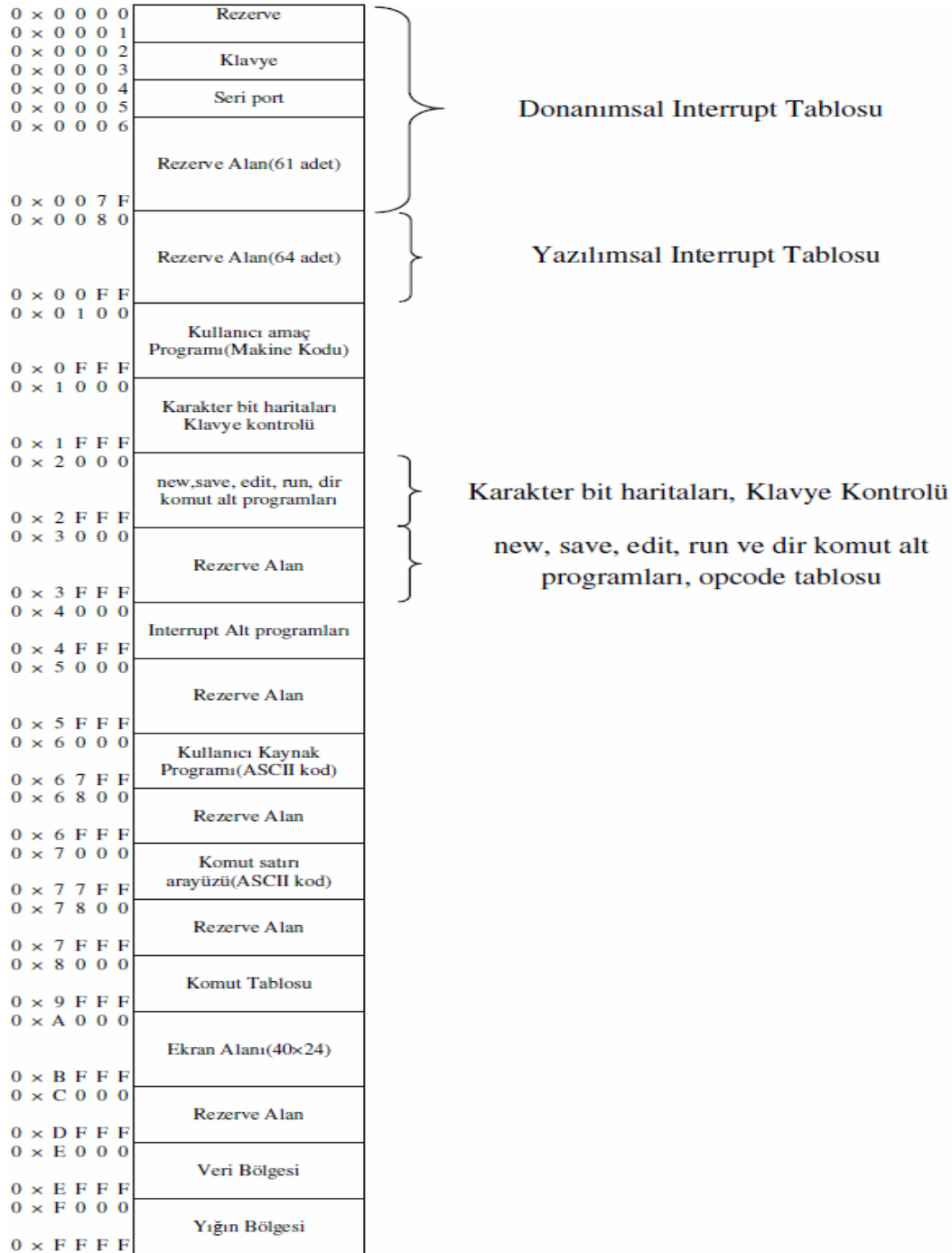
Mikro bilgisayar mimarisi üzerinde mimarinin kendi assembly komutları kullanılarak temel seviyede olan işletim sistemi komutları çalışmaktadır. Bu komutlar kullanıcının yeni bir program yazabilmesi, yazılan programı derleme ve çalıştırma, yazılan programı depolama birimine saklama ve saklı olan programın depolama biriminde çağırılarak düzenleme yapabilmesine

olanak sağlayan temel komutlardı. Depolama biriminde saklı olan dosyaların yerleşim düzeni temel dosya sistemi olan ve diğer dosya sistemlerine temel teşkil eden FAT-16 dosya sistemi baz alınarak oluşturulmuştur.

Bu mimarinin detayları için 7 ve 8 nolu referanslara bakılabilir.

3 Bellek Organizasyonu ve Bellek Haritası

İşletim sistemleri ve assembler çalışmalarına olanak sağlamak amacı ile mikro bilgisayar mimarisinde bir bellek organizasyonu ve bellek haritası oluşturulmuştur (Şekil 1).



Şekil-1 BZK&SAUOS bellek organizasyonu[7]

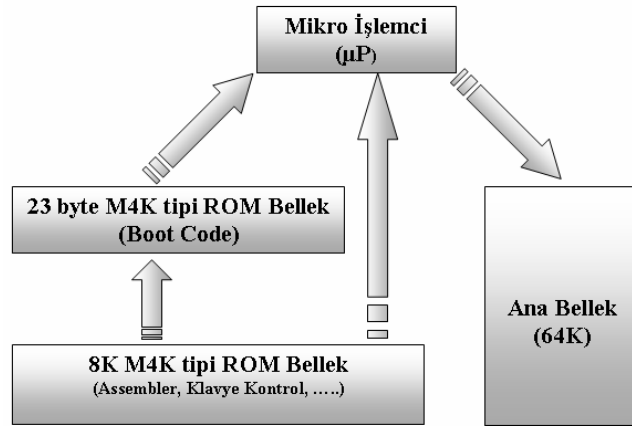
Belleğin ilk 256 byte lık bölümü donanımsal ve yazılımsal kesme tablolarına ayrılmıştır. Her bir kesme için 2 byte yer tahsis edilmiş olup rezerve kesmelerle beraber toplam 128 adet kesme desteklenmektedir. Kullanıcı kaynak programını (ASCII kod) makine koduna dönüştürüp, kullanıcı programı (makine kodu) numaralı alana aktarılmasını sağlayan Assambler programı için bellekte 1k lık yer ayrılmıştır.

4 Boot işlemi

BZK.SAU.FPGA Mikro bilgisayar birimi ilk çalıştığında, 23 byte boyutundaki M4K tipi bellekte bulunan boot kodu işleterek şu an itibariyle 64k lık bellek birime şu verileri yerleştirir.

1. Assembler programı
2. Klavye Kontrol programı
3. Komut Tablosu
4. Karakter bit haritaları
5. Kesme Tabloları

Bu verileri yerleştirdikten program akışı ana programa devredilir. Boot işlem akışı Şekil 2’de verilmiştir. Aşağıdaki şekilde yer alan M4K tipi bellek, FPGA kartının üzerinde bulunan bir tür bellek çeşididir. Bu bellek RAM veya ROM olarak kullanılabilir. FPGA kartının üreticisi tarafından geliştirilen ara yüz sayesinde bu tür belleklere.hex uzantılı başlangıç dosyası yüklenebilir. FPGA kartının bu özelliği kullanılarak mikroişlemcinin başlangıçta ana belleğe yerleştireceği sabit verileri ve boot kodunu .hex uzantılı iki adet dosya oluşturarak, sırasıyla 8k ve 23 byte kapasitesindeki belleklere yüklenmiştir.



Şekil-2 Boot işlem akışı

Alt program çalışmasını bitirdikten sonra ekranda imleci belirli aralıklarla aktif/pasif yapan alt programa dallanır. Bu alt programın da assembly komutları ise [7] nolu referansın EK-C’sinde bulunmaktadır.

5 Assembler programı

BZK.SAU.FPGA Mikro bilgisayar biriminin komut kümesi 59 adet komuttan oluşmakta olup 6 farklı adresleme modunu kullanmaktadır. Her bir komutun makine kodu 16 bittir. Doğal adresleme moduna sahip komutlar bellekte 2 word yer kaplarken, doğal adresleme moduna sahip

komut bellekte 1 word yer kaplamaktadır. Mikro bilgisayar birimi için oluşturulan assembly dilinde kullanılan adresleme mod isimleri ve bunlara karşılık gelen semboller aşağıdaki Tablo 1’de verilmiştir.

Tablo 1. Mikro bilgisayar biriminde kullanılan adresleme mod ve sembolleri

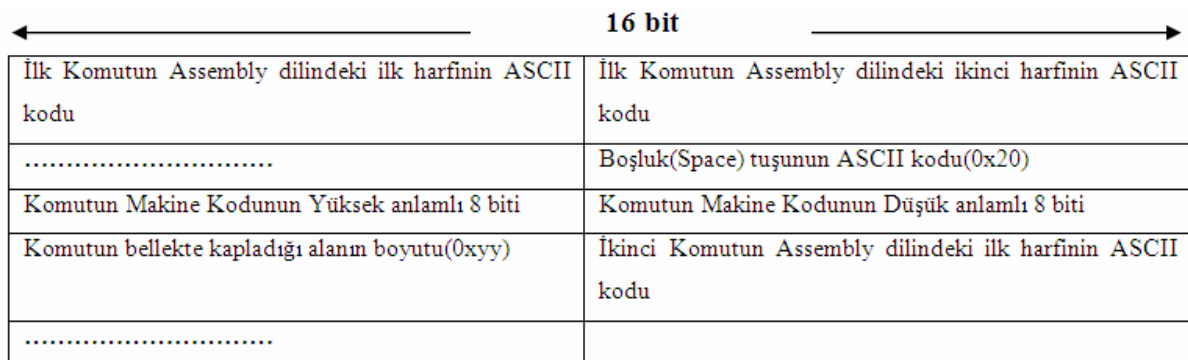
Adresleme Modu Tipi	Sembol
İvedi adresleme modu	#
Direkt adresleme modu	\$
Dolaylı adresleme modu	@
Indis adresleme modu	%
Göreceli adresleme modu	*
Doğal adresleme modu	

Kullanıcı 24 satır 40 satırdan oluşan ekranda aşağıdaki kurallara bağlı olarak assembly komutları kullanılarak program yazabilir. Assembly dilindeki komut ismini yazdıktan sonra bir boşluk bırakmalıdır. Doğal adresleme modunu kullanan bir komut kullanılacaksa boşluk bırakılmayıp, “;” karakteri ile komut sonlandırılmalıdır. Boşluk karakterinden sonra hangi adresleme modu kullanılacaksa, Tablo 1’de yer alan adresleme modu sembollerinden biri kullanılmalıdır. Bu sembolü takiben kullanılacak olan kullanılacak veri hexadecimal formatta olmalı ve sonuna “h” karakteri eklenmelidir. Komut yazımı “;” karakteri ile bitirmelidir. “;” karakterinden sonra komut hakkında açıklama yapabilir. Her bir yeni komut yeni bir satırda girilmelidir. Bu yüzden her bir yeni satıra “Enter” tuşu ile geçilmelidir. Program sonunda “-r” direktifi kullanılarak, yazılan programın makine koduna dönüşümü sağlanmalıdır. Örnek assembly programı Tablo 2’de verilmiştir.

Tablo 2. Örnek assembly programı

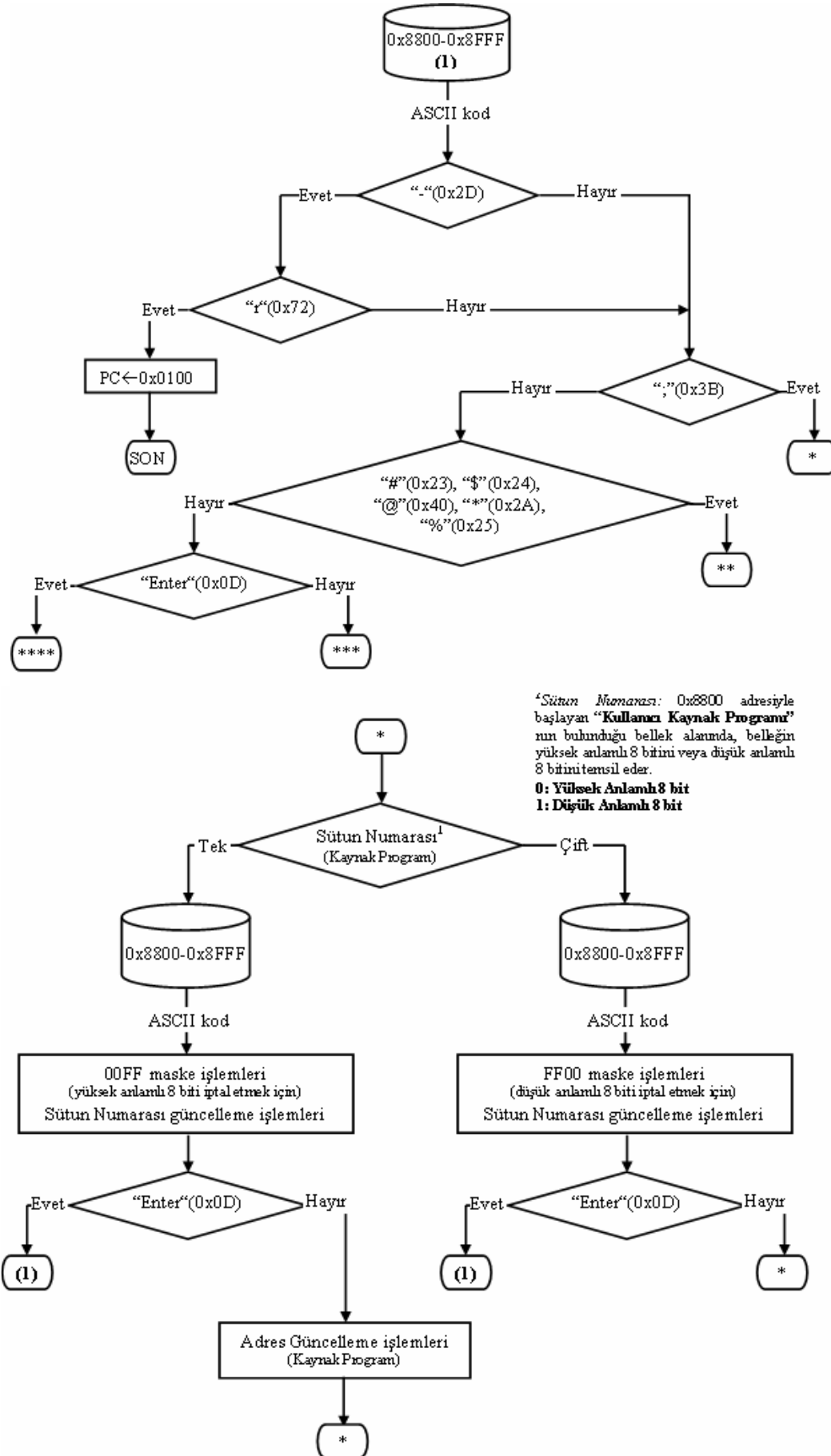
[illegible]

Her bir komut, belleğin komut tablosu alanında Şekil 3' deki düzende yerleştirilmiştir.

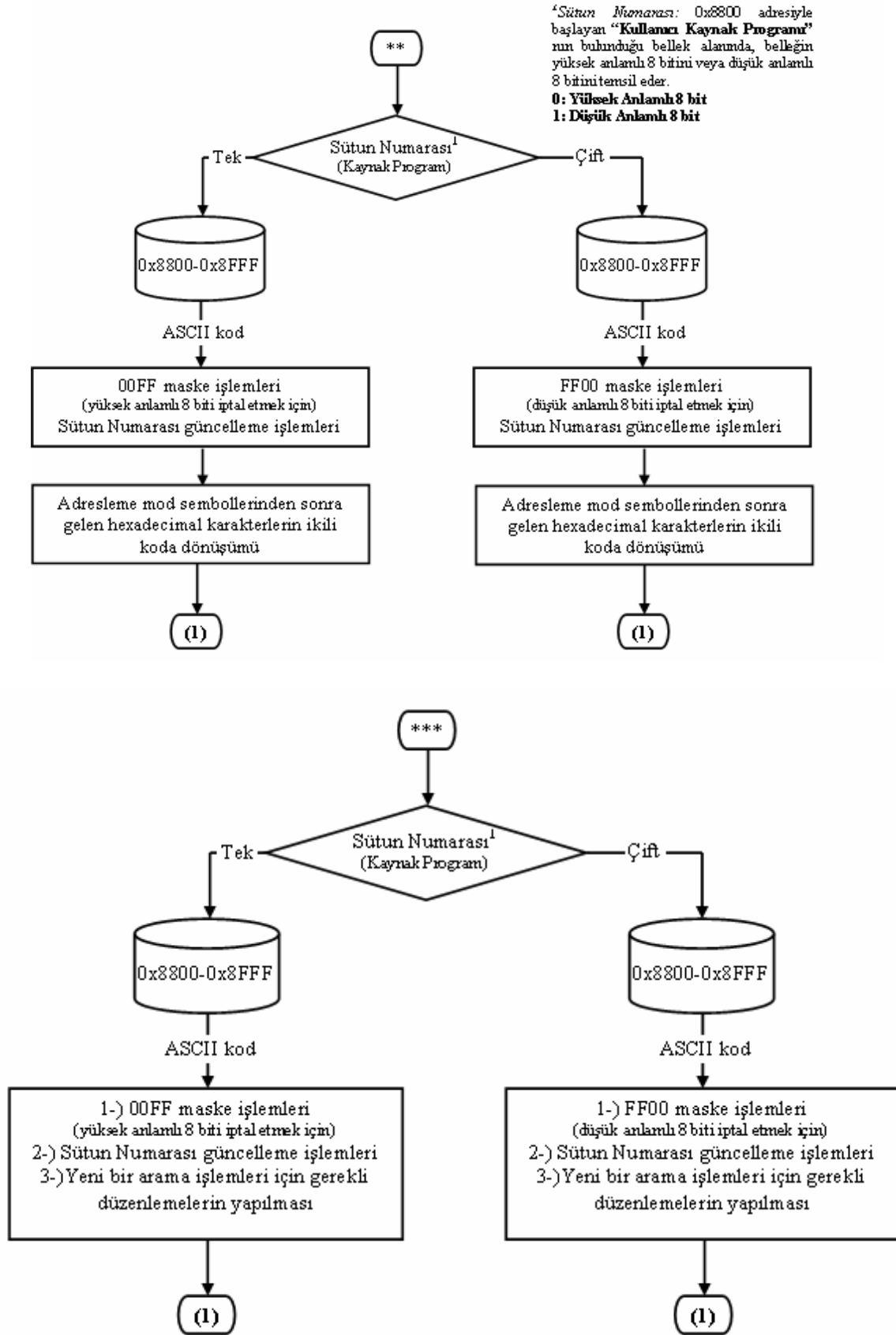


Şekil-3 Komut tablosu bellek alanında komut yerleşim düzeni [7]

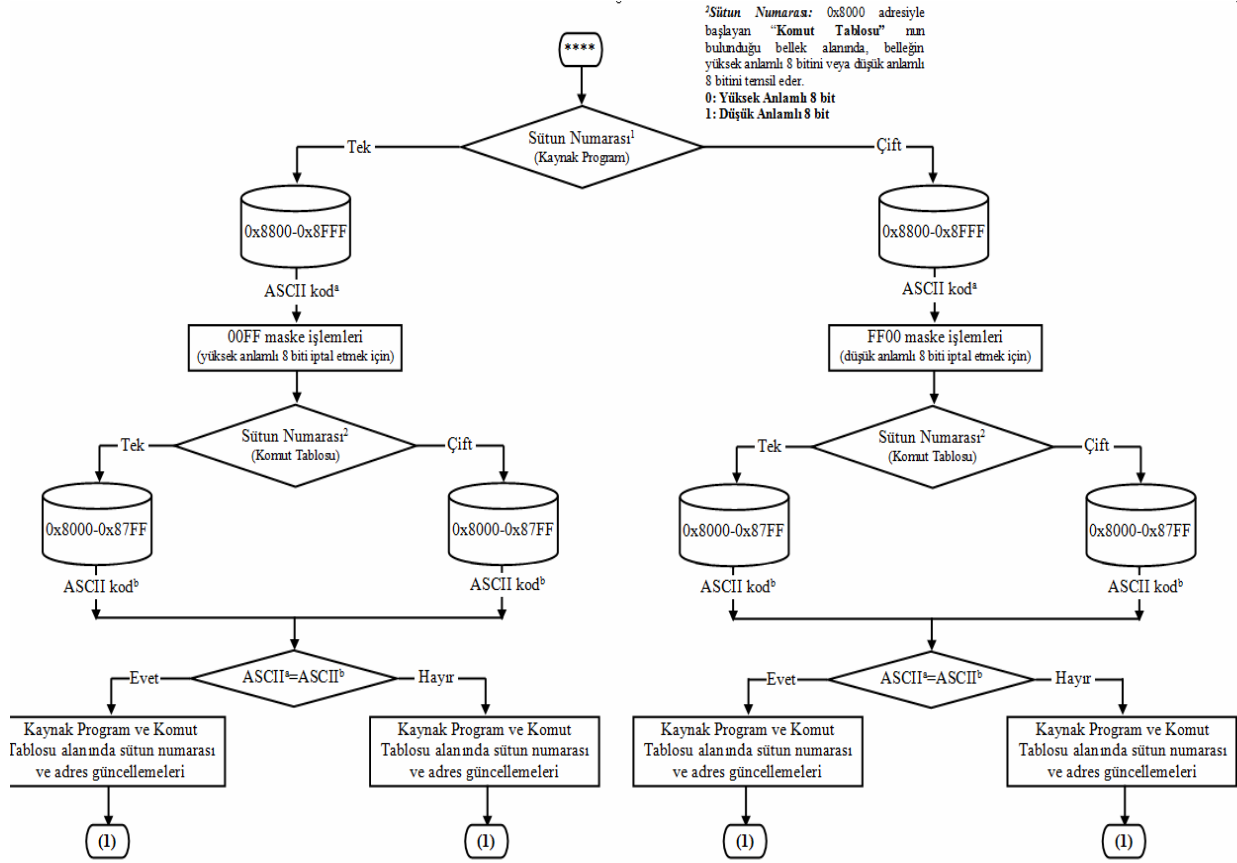
BZK.SAU.FPGA Mikro bilgisayar mimarisi için yazılan assembler programı, kullanıcı kaynak programını makine koduna dönüştürürken Brute-force Search Algoritması' nı kullanmıştır. Yazılan assembler programının akış diyagramı Şekil 4' te verilmiştir.



Şekil-4 BZK.SAU.FPGA Mikro bilgisayar mimarisi assembler programı akışı



Şekil-4 BZK.SAU.FPGA Mikro bilgisayar mimarisi assembler programı akışı (devam)



Şekil-4 BZK.SAU.FPGA Mikro bilgisayar mimarisi assembler programı akışı (devam)

6 Tartışma ve Öneriler

Assembler düşük seviyeli bir programlama dilidir. Ancak programlama açısından makine koduna göre daha anlaşılabilir ve kullanıcı açısından kod yazmayı makine koduna nazaran kolaylaştırır. BZK.SAU.FPGA Mikro bilgisayar mimarisi için yazılan assembler programı, kullanıcı kaynak programını makine koduna dönüştürerek kullanıcı programının mimari üzerinde çalışmasına olanak sağlamaktadır.

Literatürde eğitimsel mikro bilgisayar mimarisi tasarımları bulunmasına karşın bu tasarımlara kullanıcının entegre olabilmesine olanak sağlayacak derecede esnek yapılar olmadığı görülmektedir. Ayrıca bir bilgisayar mimarisi üzerine sıfırdan bir assemblerın nasıl yazılacağı konusunda rehber niteliği taşıyacak eğitimsel bir doküman literatürde bulunmamaktadır. Tasarlanan modüler BZK.SAU mikro bilgisayar mimarisi için bir assembler ve temel seviyede bir işletim sistemi yazılmıştır. Oluşturulan assembler ve işletim sisteminin en temel özelliği, özgün bir mikro bilgisayar mimarisi donanımına has yazılmış olmasıdır. Burada amaçlanan, bir assemblerın sıfırdan herhangi bir donanım üzerine nasıl oluşturulacağı konusunda eğitimsel bir platform ve doküman oluşturulmasıdır.

Teşekkür

Bu çalışma, 110E069 numaralı TÜBİTAK 1001 projesi kapsamında desteklenmiştir.

References (Referanslar)

1. GARCIA, M.I., Rodriguez, S., Perez, A., Garcia, A., p88110: A Graphical Simulator for Computer Architecture and Organization Courses, IEEE Transactions on Education, 52, 2, 248–256, (2009).
2. NIKOLIC, Z., Radivojevic, J., Djordjevic, J., Milutinovic, V., A Survey and Evaluation of Simulators Suitable for Teaching Courses in Computer Architecture and Organization, IEEE Transactions on Education, 52, 4, 449–458, (2009).
3. STOJKOVIC, A., Djordjevic, J., Nikolic, B., WASP: A Web Based Educational System for Teaching Computer Architecture and Organization, International Journal Electrical Engineering Education, 44, 3, 197–215, (2007).
4. DJORDJEVIC, J., Nikolic, B., Milenkovic, A., Flexible Web-based Educational System for Teaching Computer Architecture and Organization, IEEE Transactions on Education, 48, 2, 264–273, (2005).
5. QINGQIANG, W., Langcai, C., Teaching Mode of Operating System Course for Undergraduates Majoring in Computer Sciences, 4th International Conference on Computer Science & Education, Xiamen- China, (2009) pp: 1412-1415.
6. The Joint Task Force on Computing Curricula, IEEE Computer Society and Association for Computing Machinery, Computin Curricula, (2001).
7. ÖZTEKİN, H., Eğitim Amaçlı Yapılandırılabilir Modüler Donanım Üzerine Gömülü İşletim Sistemi Tasarımı, (Doktora Tezi), Sakarya Üniversitesi Fen Bilimleri Enstitüsü, (2012).
8. Proje Raporu: Proje No: 110E069 Proje Adı: Uzaktan Erişilebilir Yapılandırılabilir Donanım Üzerine Eğitim Amaçlı Mikro Bilgisayar Mimarisi ve Gömülü İşletim Sistemi Tasarımı (2012).
9. OZTEKIN, H., TEMURTAS, F., GULBAG, A., “BZK.SAU.FPGA10.1: A Modular Approach to FPGA-Based Micro Computer Architecture Design for Educational Purpose” Computer Applications in Engineering Education, 22(2), (2014) 272-279.
10. http://en.wikipedia.org/wiki/ABC_80 (Erişim Tarihi: 20/02/2012)
11. en.wikipedia.org/wiki/List_of_home_computers_by_video_hardware (Erişim Tarihi: 20/02/2012)
12. MANO, M. M., Bilgisayar Sistemleri Mimarisi, MARŞOĞLU, A., 3. Basım, SUÇSUZ., N., Literatür Yayıncılık, pp. 129–159, İstanbul, 2002.