



T.C.
ULUDAĞ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ



İŞLEMCİ TASARIMI

HÜSEYİN KAYA
031411503

MÜHENDİSLİK TASARIMI I

BURSA 2019

T.C.
ULUDAĞ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

İŞLEMCİ TASARIMI

HÜSEYİN KAYA
031411503

Projenin Danışmanı: Dr. GÖKHAN YENİKAYA

Uludağ Üniversitesi Mühendislik Fakültesi tez yazım kurallarına uygun olarak hazırladığımı bu Mühendislik Tasarımı I çalışmasında;

- Tez içindeki bütün bilgi ve belgeleri, akademik kurallar çerçevesinde elde ettiğimizi.
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları, bilimsel ahlak kurallarına uygun olarak sunduğumuzu
- Başkalarının eserlerinden yararlanılması durumunda, ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumuzu
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimizi
- Kullanılan verilerde herhangi bir tahrifat yapmadığımızı,
- Bu tezin herhangi bir bölümünü üniversitemde veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımızı

beyan ederiz.

03.01.2019

Hüseyin Kaya

Danışmanlığında hazırlanan Bitirme Projesi çalışması, tarafımdan kontrol edilmiştir.

03.01.2019

Dr. Gökhan Yenikaya

ÖZET

4 bit işlemci tasarımı yapmak ve yapılabilirliği varsa bunu gerçeklemek. Öncelikle bu projede temel şeyleri tasarlayarak başladım. ALU gibi çünkü ALU tasarımı basit ve yapabileceği işlemler kısıtlı toplama çıkarma çarpma ve diğer aritmetik işlemler gibi ve bunu sonucunda gerekli kontrol sinyallerini elde ettim bundan başlamam da ki amaç buydu.

Ve sonuç olarak basit düşük frekanslarda çalışan ama işlevli bir işlemci mimarisi ortaya koyulmuştur.

ABSTRACT

Design a 4 bit processor and do it if feasible. I started by designing the basic things in this project. Like the ALU, because the ALU design is simple and the operations I can do are limited addition, subtraction, multiplication and other arithmetic operations, and I have obtained the necessary control signals as a result of which I did not start.

And as a result, a simple but low-functioning processor architecture has been put forward.

ŞEKİLLER DİZİNİ

Sayfa No

Şekil 1. İşlemciler.....	9
Şekil 2. Yarı iletkenler.....	11
Şekil 3. P-N katkı işlemi.....	12
Şekil 4. N tipi yarı iletken.....	13
Şekil 5. P tipi yarı iletken	14
Şekil 6. Diyotlar.....	15
Şekil 7. Transistörler.....	16
Şekil 8. Transistörlerin çalışması.....	17
Şekil 9. Entegreler.....	19
Şekil 10. Nand kapısı.....	20
Şekil 11. And kapısı.....	20
Şekil 12. Nor kapısı.....	20
Şekil 13. Or kapısı.....	20
Şekil 14. Not kapısı.....	20
Şekil 15. Xor kapısı.....	20
Şekil 16. Lojik kapılar doğruluk tablosu.....	20
Şekil 17. Entegrelerde gecikme.....	21
Şekil 18. Karnaugh map örneği.....	23
Şekil 19. Yarım toplayıcı.....	24
Şekil 20. Tam toplayıcı.....	25
Şekil 21. D flip flop.....	26
Şekil 22. 4 bit saklayıcı tasarımı.....	26
Şekil 23. Multisim ALU tasarımı.....	29
Şekil 24. Multisim CU tasarımı.....	31
Şekil 25. Multisim CU Gecikme.....	32
Şekil 26. Quartus programı inceleme.....	34

Şekil 27. CPU verilog.....	35
Şekil 28. ALU verilog.....	36
Şekil 29. CPU RTL.....	37
Şekil 30. ALU RTL.....	37
Şekil 31. ALTERA SIM simulasyon.....	38

ÇİZELGELER DİZİNİ

Sayfa No

Çizelge 1. TTL ve CMOS Karşılaştırması.....	19
---	----

İÇİNDEKİLER

ÖZET

ABSTRACT

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1.1 İŞLEMCİ NEDİR ?.....	9
1.2 İŞLEMCİ TARİHİ.....	10
2.1 YARI İLETKENLER.....	11
2.2 KATKI İŞLEMİ DOPİNG.....	12
2.3 N TİPİ YARI İLETKEN.....	13
2.4 P TİPİ YARI İLETKEN.....	14
3.1 DİYOTLAR.....	15
3.2 TRANSİSTÖRLERİN YAPISI.....	16
3.3 TRANSİSTÖRLERİN ÇALIŞMASI.....	17
3.4 ENTEGRE TANIMI VE SINIFLANDIRILMASI.....	18
3.5 ENTEGRASYON SEVİYELERİNE GÖRE.....	18
3.6 CMOS VE TTL ENTEGRE KARŞILAŞTIRMASI.....	19
3.7 TTL ENTEGRELER.....	19
3.8 CMOS ENTEGRELER.....	19
3.9 LOJİK KAPILAR.....	20
3.10 ENTEGRELERDE GECİKME VE LOJİK DEĞERLERİ.....	21
3.11 BOOLE CEBRİ.....	22
3.12 KARNAUGH HARİTASI.....	23
3.13 YARIM TOPLAYICI (LOJİK DEVRELER)	24
3.14 TAM TOPLAYICI (LOJİK DEVRELER)	25
3.15 SAKLAYICILAR.....	26
4.1 CPU DONANANIMI.....	27
4.2 ALU (ARİTMETİK MANTIK BİRİMİ)	28
4.3 ALU TASARIMI SONUÇLARI.....	29
4.4 CU (KONTROL BİRİMİ)	30
4.5 CU TASARIMI SONUÇLARI.....	31
4.6 CU ZAMANLAMA HATALARI SONUÇLARI.....	32
4.7 OPCODE KÜTÜPHANE OLUŞTURMA.....	33

5.1 QUARTUS PROGRAMINI İNCELEME.....	34
5.2 CPU VERİLOG TASARIMI.....	35
5.3 ALU VERİLOG TASARIMI.....	36
5.4 CPU RTL GÖRÜNTÜLEYİCİSİ.....	37
5.5 ALTERA SİM SİMULASYON SONUÇLARI.....	38
6. KAYNAKÇA.....	39
7. TEŞEKKÜR.....	40
8. ÖZGEÇMİŞ.....	41

1.1 İŞLEMÇİ NEDİR ?

İşlemci, kısaca bilgisayarın beynidir. Yani insanı nasıl beyin yönetiyorsa ve organlarının düzenli bir şekilde iletişimini, çalışmasını sağlıyorsa işlemci de bilgisayarı yönetiyor, parçaların düzenli bir şekilde çalışmasını sağlıyor. İşlemci için söylenen farklı isimler vardır. Bunlar CPU, mikroişlemci, MİB gibi kelimelerdir. Aslında hepsi aynı şeyi ifade etmektedir. Bazıları arama motorları üzerinden araştırma yaparken işlemci nedir şeklinde arama kutusuna yazıyorlar. Ama ne olursa olsun arama motorları o konuda bize doğru sonuçlara ulaştırıyor.

CPU kelimesi, Central Processing Unit kelimelerinin baş harflerinin alınarak bir araya getirilerek oluşturulmuş bir kelimedir. Central Processing Unit Türkçe’de merkezi işlem birimi anlamına gelmektedir. Anlayacağınız üzere MİB kelimesi de bu kelimelerin baş harflerinden oluşmuştur.

Mikroişlemciler açma kapama anahtarı gibi çalışan milyonlarca transistörden oluşur. Bu anahtarlarının programlanma durumuna göre elektrik sinyalleri bunların üzerinden akmaktadır. İşte bu sinyaller bilgisayarın yaptığı bütün matematiksel işlemlere indirebilmesini de sağlamaktadır. Ayrıca işlemci en basit sayma sistemi olan ikilik düzen yani 0 ve 1 sayılarını kullanmaktadır.



Şekil-1

1.2 İŞLEMCI TARİHÇESİ

1971 – Intel 4004, ilk tek yongalı CPU, saat hızı 740 kHz, veri 4 bit, komutlar 8 bit uzunluğundadır. Transistör sayısı 2300, pin sayısı 16, program hafızası 4KB, adreslenebilir hafızası 640 Byte'tır.

1972 – TMS 1000, yeterli bellek ve bir ROM programı için yeterli alan içeren ilk mikroişlemcidir.

1972- Intel 8008, saat hızı 0.5 MHz-0.8 MHz, yol genişliği 8 bit, 14 bit adres yolu, 16 KB bellek.

1974 – Intel 8080, saat hızı 2MHz, yol genişliği 8 bit, 16 bit adres yolu, 64 KB bellek, ev bilgisayarları için kullanılan ilk işlemcidir.

1975 – Motorola 6800.

1976 – Intel 8085, saat hızı 3,5-6 MHz, yol genişliği 8 bit, 16 bit adres yolu, 8 bit veri yolu, 64 KB bellek, 40 pin.

1978 – Intel 8086, saat hızı 5-10 MHz, yol genişliği 16 bit.

Intel 8086'nın ardından Intel 8088, 80186, 80188, 80286, 80386, 80486 geldi. Ardından 1993'te Pentium mimarisi ile Pentium işlemciler üretilmeye başlandı. Pentium'un ardından Intel firması Core ve Celeron işlemcileri üretmeye başladı. Bugün ise Intel Sandy Bridge teknolojisine sahip 2. nesil işlemcilerini tanıtmış bulunmaktadır.

Intel'in en güçlü rakibi olan AMD ise ilk işlemcisini AM386'yı 1991 yılında piyasaya sürdü. Aslında bu ürünü daha önce de çıkarabileceği halde, Intel'le yaşadığı sözleşme anlaşmazlığı bunu geciktirdi, en sonunda AMD'nin mahkemeyi kazanması sonucu bu ürün de satılmaya başlandı. AM386'nın ardından, AM486, AM5x86, K5, K6, K6-2, Athlon, Phenom, Tuiron ve Opteron gibi işlemciler geldi.

2.1 YARI İLETKENLER

Doğada bulunan atomlar elektriği iletip-iletmemeye durumuna göre iletken, yalıtkan ve yarı iletken olarak 3'e ayrılırlar. İletken maddelere örnek olarak demir, bakır, altın yalıtkan maddeler örnek olarak tahta, hava, saf su verilebilir. Yalıtkan madde deyince akla gelen iki önemli element silisyum ve germanyumdur.

İletkenlik bakımından iletkenler ile yalıtkanlar arasında yer alırlar, normal halde yalıtkandırlar. Ancak ısı, ışık ve magnetik etki altında bırakıldığında veya gerilim uygulandığında bir miktar valans elektronu serbest hale geçer, yani iletkenlik özelliği kazanır. Bu şekilde iletkenlik özelliği kazanması geçici olup, dış etki kalkınca elektronlar tekrar atomlarına dönerler. Tabiatta basit eleman halinde bulunduğu gibi laboratuarda bileşik eleman halinde de elde edilir. Yarı iletkenler kristal yapıya sahiptirler. Yani atomları kübik kafes sistemi denilen belirli bir düzende sıralanmıştır. Bu tür yarı iletkenler, yukarıda belirtildiği gibi ısı, ışık, etkisi ve gerilim uygulanması ile belirli oranda iletken hale geçirildiği gibi, içlerine bazı özel maddeler katılarakta iletkenlikleri arttırılmaktadır. Katkı maddeleriyle iletkenlikleri arttırılan yarı iletkenlerin elektronikte ayrı bir yeri vardır.

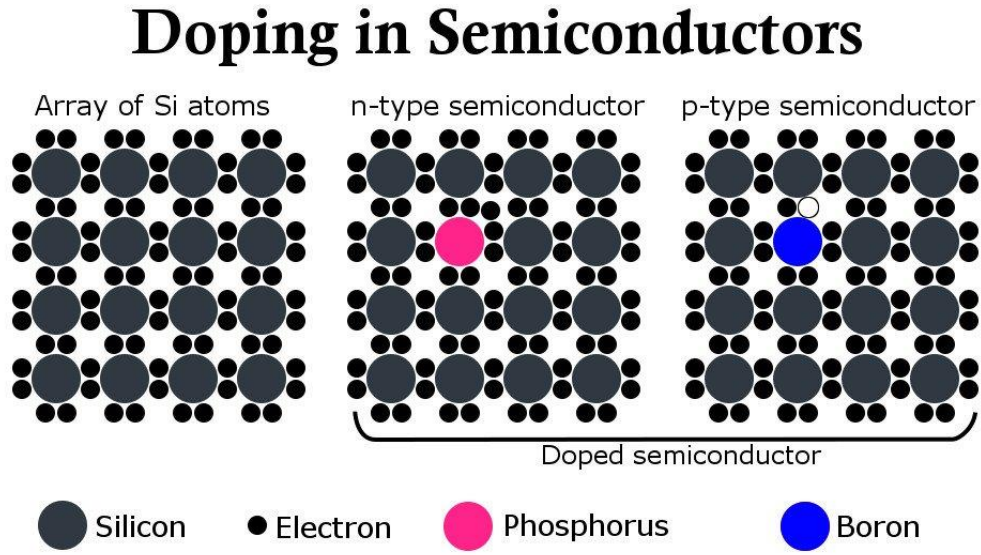
5 B Boron 2.34	6 C Carbon 2.62	7 N Nitrogen 1.251
13 Al Aluminum 2.70	14 Si Silicon 2.33	15 P Phosphorus 1.82
31 Ga Gallium 5.91	32 Ge Germanium 5.32	33 As Arsenic 5.72

©2001 HowStuffWorks

Şekil-2

2.2 KATKI İŞLEMİ (DOPİNG)

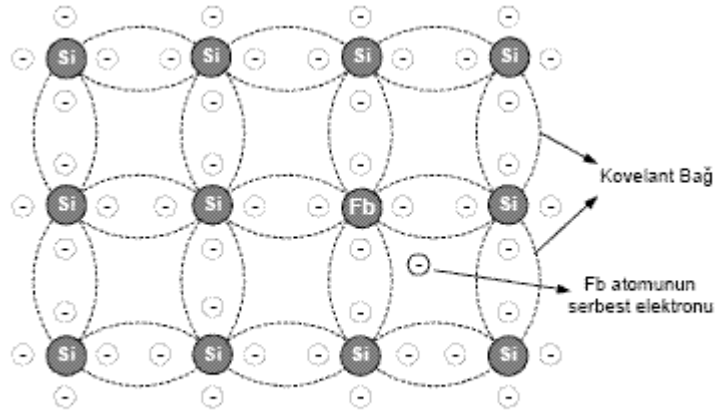
Silisyum ve germanyumun iletkenliđi kontrollü olarak artırılabilir. İletkenliđi kontrollü olarak artırmak için saf yarıiletken malzemeye katkı maddesi eklenir. Bu işleme “doping” denir. Akım taşıyıcılarının (elektron veya boşluk) sayısının artırılması malzemenin iletkenliđini, azaltılması ise malzemenin direnci artırır. Her iki doping olayının sonucunda N-tipi veya P-tipi madde oluşur.



Şekil-3

2.3 N-TİPİ YARI İLTKEN

Saf silisyumun iletkenlik bandındaki deliklerinin artırılması atomlara katkı maddesi ekleyerek yapılır. Bu atomlar, 5-değerli valans elektronları olan arsenik (As), fosfor (P), bizmut (Bi) veya antimon'dur. Silisyuma katkı maddesi olarak 5 valans elektrona sahip fosfor belli bir oranda eklendiğinde, diğer silisyum atomları ile nasıl bir kovalent bağ oluşturulduğu şekil-1.4'te gösterilmiştir. Fosfor atomunun 4 valans elektronu, silisyumun 4 valans elektronu ile kovalent bağ oluşturur. Fosfor'un 1 valans elektronu açıkta kalır ve ayrılır. Bu açıkta kalan elektron iletkenliği artırır. Çünkü herhangi bir atoma bağlı değildir. İletkenlik, elektron sayıları ile kontrol edilebilir. Bu ise silisyuma eklenen atomların sayısı ile olur. Katkı sonucu oluşturulan bu iletkenlik elektronu, valans bandında bir boşluk oluşturmaz.



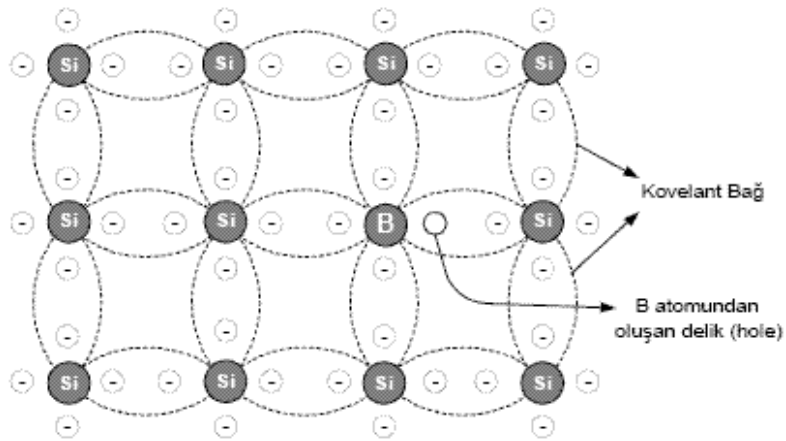
Şekil-1.3 N tipi yarıiletken maddenin oluşturulması.

Şekil-4

2.4 P-TİPİ YARI İLTKEN

Saf silisyum atomu içerisine, 3 valans elektrona sahip (3-değerli) atomların belli bir oranda eklenmesi ile yeni bir kristal yapı oluşur. Bu yeni kristal yapıda delik (boşluk) sayısı artırılmış olur. 3 valans elektrona sahip atomlara örnek olarak; alüminyum (Al), Bor (B) ve Galyum (Ga) elementlerini verebiliriz. Örneğin; saf silisyum içerisine belli bir oranda bor katılırsa; bor elementinin 3 valans elektronu, silisyumun 3 valans elektronu ile ortak kovalent bağ oluşturur. Fakat silisyumun 1 valans elektronu ortak valans bağ oluşturamaz. Bu durumda 1 elektron noksanlığı meydana gelir. Buna “boşluk” veya “delik=hole” denir. Silisyuma eklenen katkı miktarı ile boşlukların sayısı kontrol edilebilir. Bu yöntemle elde edilen yeni malzemeye P tipi yarıiletken malzeme denir. Çünkü boşluklar pozitif yüklüdür. Dolayısı ile P-tipi malzemede çoğunluk akım taşıyıcıları boşluklardır.

Elektronlar ise P tipi malzemede azınlık akım taşıyıcılarıdır. P-tipi malzemede bir kaç adet serbest elektronda oluşmuştur. Bunlar ısı ile oluşan boşluk çifti esnasında meydana gelmiştir. Bu serbest elektronlar, silisyuma yapılan katkı esnasında oluşturulamazlar. Elektronlar P-tipi malzemede azınlık akım taşıyıcılarıdır.



Şekil- 1.4 Silisyum kristaline 3 bağlı katkı atomu. Bor katkısı atomu merkezde gösterilmiştir.

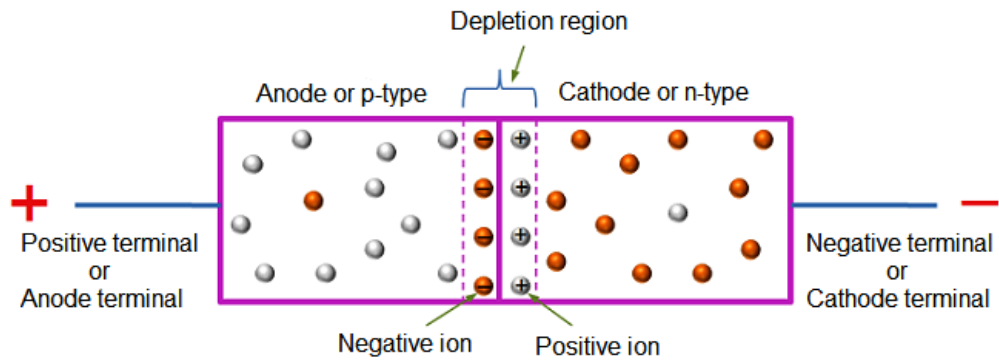
Şekil-5

3.1 DİYOTLAR

N tipi ve P tipi iki malzeme arasında bir bağlantı oluşturulduğunda ortaya çıkan yarı iletken devre elemanı diyottur. Bu devre elemanı bir yönde oldukça düşük direnç sağlayarak elektrik akımının geçmesine izin verirken, diğer yönde oldukça yüksek direnç göstererek elektrik akımının geçmesini engeller. Bu karakteristik yapı, elektrik akımının tek yönde geçmesi istenen devrelerde kullanılmasını sağlamıştır.

İdeal diyot teoride, tek yönde sonsuz miktarda akım geçişini sağlayabilirken, ters yönde hiç akım geçirmez. Ayrıca ideal diyota uygulanan en küçük potansiyel fark doğru kutuplanmışsa diyot iletme hemen geçecektir. Diğer taraftan, ters yöndeki kaçak akım değeri ise oldukça küçüktür.

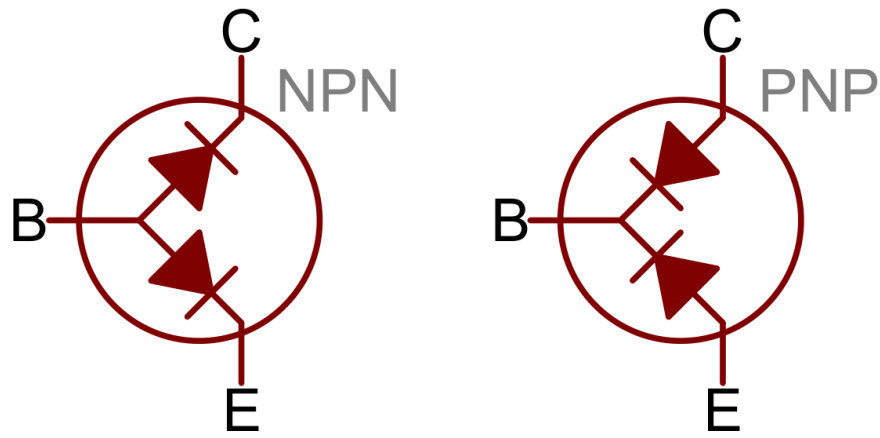
Bir yarı iletken diyotu oluşturan, P tipi yarı iletken materyal pozitif “+” kutup, N yarı tipi iletken materyal ise negatif “-” kutup olacak şekilde, ileri yöndeki eşik gerilimini (Silisyum materyal için 0.6-0.7 V, Germanyum materyal için 0.2-0.3 V) aşacak seviyede bir potansiyel farkı uygulanırsa diyot serbest bir şekilde akım geçirmeye başlar. Bunun tersi durumda ise, yani P tipi maddeye negatif, N tipi maddeye pozitif kutup gelecek şekilde ters yönde bir gerilim uygulanırsa (Diyotun ters kırılma gerilimini aşmayacak şekilde) diyottan bir akım geçişi olmaz. Eğer ki diyota uygulanan gerilim bu ters kırılma gerilimi aşacak olursa, diyotun bozulmasına sebep oluruz.



Şekil-6

3.2 TRANSİSTÖRLERİN YAPISI

Transistör bir gerilim ya da akım kaynağı ile başka bir akım ya da gerilim kaynağını kontrol etmeye yarayan elektronik devre elemanıdır. Transistör yan yana birleştirilmiş PN diyodundan meydana gelen, girişine uygulanan sinyal yükselterek akım ve gerilim kazancı sağlayan, gerektiği zaman anahtarlama elemanı olarak kullanılan yarı iletken bir elektronik devre elemanıdır. Transistör kelimesi transfer ve rezistans kelimelerinin birleşiminden türemiştir. Özel teknikle diyot yapısına üçüncü bir tabaka (P tp veya N tp) ilave edilerek oluşturulan üçüncü uç sayesinde elde edilen yeni elemanla dış devre akımları, devre yükü ve kaynak voltajı değiştirilmeden kontrol edilebilir. En çok kullanılan türler BJT ve FET'lerdir. BJT'ler akım ile çalışırken FET'ler gerilimin oluşturduğu elektrik alanla çalışırlar. FET'ler günümüzde daha çok tümleşik sayısal devrelerde kullanılmaktadır. Transistörler üç bağlantıya sahiptirler. Bunlar bir BJT transistörde Base, Emitter ve Collector iken FET'lerde ise Gate, Drain ve Source dur.

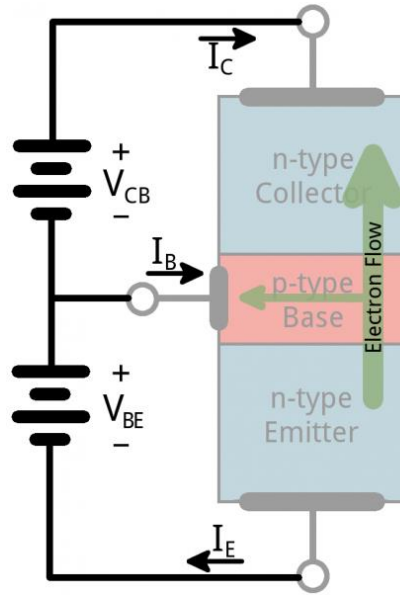


Şekil-7

3.3 TRANSİSTÖRLERİN ÇALIŞMASI

İster bir anahtar, ister bir yükseltici, isterse de bir üreteç işlevi görsün, bütün transistörler elektrik direncinin değişmesine dayalı olarak çalışır. Base akımı yada Gate gerilimi olmadığında Collector ile Emitter (yada Drain ile Source) arasındaki direnç o kadar yüksektir ki bu iki bağlantı arasında hemen hemen hiçbir akım geçemez. Ama Base bağlantısında küçük bir akım(Gate de küçük bir gerilim) aktarıldığında Collector ile Emitter (Drain ile Source) arasındaki dirençte çok büyük azalma olur. Dolayısıyla arasından akım geçebilir. Böylece transistör küçük bir akımın ya da gerilimin yardımıyla büyük bir akımı denetleyebilir.

Transistör bir anahtar olarak kullanıldığında zaman, giriş bağlantısına küçük bir akım (FET'lerde gerilim) verildiğinde güçlü bir elektrik akımının devresini tamamlamasına izin verir. Bir yükseltici ya da bir üreteç olarak kullanıldığında zaman zayıf bir sinyali güçlendirir. Zayıf sinyal küçük bir elektrik akımı ya da gerilimi biçiminde girişe (Base-Gate) uygulanır. Bu, Collector' den Emitter'e (Source'dan Drain'e) büyük bir akımın geçmesine izin verir. Böylece güçlü bir sinyal üretilmiş olur.



Şekil-8

3.4 ENTEGRE TANIMI VE SINIFLANDIRILMASI

CMOS ile TTL adlı yapılar, entegre olarak adlandırılan devre grubuna girerler. Bu sebeple önce entegre kavramını açıklayım. Bu sayede CMOS ile TTL tanımlarını daha rahat ve anlaşılır hale getiririz.

Entegre: Belli bir fonksiyonu gerçekleştirmek amacıyla çok sayıda direnç, diyot ve transistörlerin bir araya getirilerek oluşturulan devrelere entegre adı verilir. Entegreler, entegrasyon seviyelerine ve yapılarında kullanılan transistör tipine sınıflandırılabilirler. Lojik uygulamada seçilecek entegre devre familyası, devrenin özelliklerine göre belirlenir. Günümüzde çok özel devreler hariç genellikle devre gerçekleştirmede TTL ve CMOS familyasını entegre devreler kullanılmaktadır.

3.5 Entegrasyon Seviyelerine göre:

SSI (Small-Scale Integration): 12 den az transistör içeren entegreler

Medium Scale Integration (MSI): 12 ile 99 arası transistör içeren entegreler. (örneğin flip-floplar, sayıcılar)

Large Scale Integration (LSI): 100 ile 9.999 arası transistör içeren entegreler (örneğin hafıza elemanları EPROM, ROM)

VLSI (Very Large Scale Integration): 10.000-99.999 arası transistör içeren entegreler (örneğin 8-bit basit mikro işlemciler)

ULSI (Ultra Large Scale Integration): 100.000- ve fazlası transistör içeren entegreler (örneğin gelişmiş entegreler)

3.6 CMOS VE TTL ENTEGRE KARŞILAŞTIRILMASI

3.7 TTL (Transistor-Transistor Logic) Entegreler:

Yapılarında bipolar transistörler kullanılır. Besleme gerilimleri 5V' tur. CMOS entegrelere göre güç kayıpları çok fazladır.

Standard TTL (74XXX ailesi): en eski, yavaş ve güç kayıpları çok fazla

Low Power TTL (74LXXX ailesi): daha az güç kayıpları

Schottky TTL (74SXXX ailesi): hızlı fakat güç kayıpları fazla

Low Power Schottky TTL (74LSXXX ailesi): hızlı ve düşük güç kayıplarına sahip

Advanced LS TTL (74ALSXXX): hız-güç kayıpları oranı çok iyi

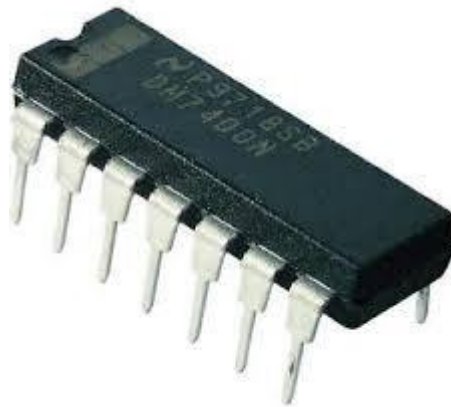
FAST TTL (74FXXX): hız ve güç kayıpları açısından en iyi TTL entegresi

3.8 CMOS (Complementary Metal-Oxide Silicon) Entegreler:

Yapıları FET türü transistörlerden oluşur. Besleme gerilimleri 3V ile 15V arasında olabilir. TTL den çok daha az güç kayıpları vardır.

40XX ailesi: En eski CMOS, yavaş, güç kaybı çok az.

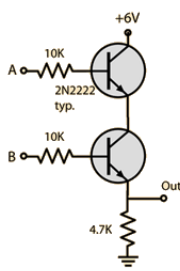
74HCTXXX ailesi: TTL uyumlu (besleme gerilimi 5V), CMOS un avantajları (çok düşük güç kayıpları) ile TTL in avantajlarını (çok hızlı) bir arada bulundurur. En popüler entegre.



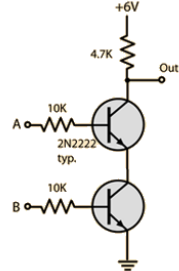
Şekil-9

3.9 LOJİK KAPILAR

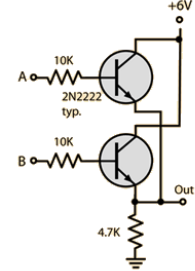
Lojik kapılar elektrik anahtarlama ilkesine dayalı elektrik aygıtlardır. Genelde entegre biçiminde üretilirler ve yapılarında transistör, diyot ve direnç gibi temel elektronik elemanlar kullanılır. Lojik kapılar üç ana mantık elemanından oluşur. Bunlar AND (VE), OR (VEYA) ve NOT (DEĞİL) kapılarıdır.



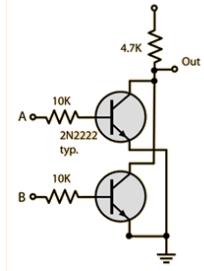
Şekil-10



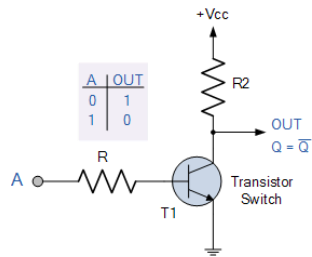
Şekil-11



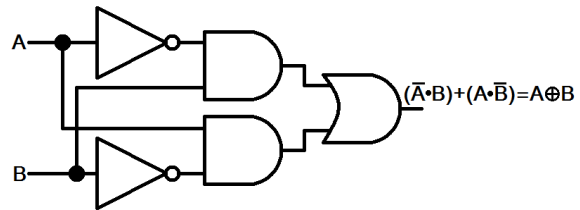
Şekil-12



Şekil-13



Şekil-14



Şekil-15

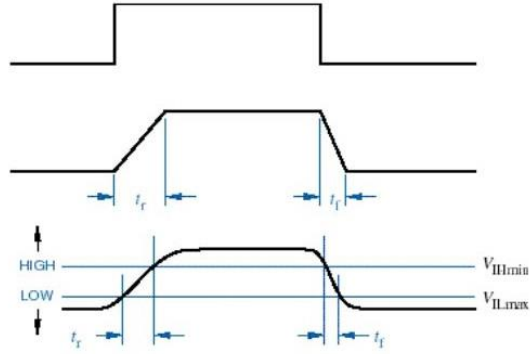
Logic Gates

Name	NOT	AND	NAND	OR	NOR	XOR	XNOR																																																																																																
Alg. Expr.	$\overline{A}$	AB	$\overline{AB}$	$A+B$	$\overline{A+B}$	$A\oplus B$	$\overline{A\oplus B}$																																																																																																
Symbol																																																																																																							
Truth Table	<table><tr><th>A</th><th>X</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	X	0	1	1	0	<table><tr><th>B</th><th>A</th><th>X</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	B	A	X	0	0	0	0	1	0	1	0	0	1	1	1	<table><tr><th>B</th><th>A</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	B	A	X	0	0	1	0	1	1	1	0	1	1	1	0	<table><tr><th>B</th><th>A</th><th>X</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	B	A	X	0	0	0	0	1	1	1	0	1	1	1	1	<table><tr><th>B</th><th>A</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	B	A	X	0	0	1	0	1	0	1	0	0	1	1	0	<table><tr><th>B</th><th>A</th><th>X</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	B	A	X	0	0	0	0	1	1	1	0	1	1	1	0	<table><tr><th>B</th><th>A</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	B	A	X	0	0	1	0	1	0	1	0	0	1	1	1
A	X																																																																																																						
0	1																																																																																																						
1	0																																																																																																						
B	A	X																																																																																																					
0	0	0																																																																																																					
0	1	0																																																																																																					
1	0	0																																																																																																					
1	1	1																																																																																																					
B	A	X																																																																																																					
0	0	1																																																																																																					
0	1	1																																																																																																					
1	0	1																																																																																																					
1	1	0																																																																																																					
B	A	X																																																																																																					
0	0	0																																																																																																					
0	1	1																																																																																																					
1	0	1																																																																																																					
1	1	1																																																																																																					
B	A	X																																																																																																					
0	0	1																																																																																																					
0	1	0																																																																																																					
1	0	0																																																																																																					
1	1	0																																																																																																					
B	A	X																																																																																																					
0	0	0																																																																																																					
0	1	1																																																																																																					
1	0	1																																																																																																					
1	1	0																																																																																																					
B	A	X																																																																																																					
0	0	1																																																																																																					
0	1	0																																																																																																					
1	0	0																																																																																																					
1	1	1																																																																																																					

Şekil-16

3.10 ENTEGRELERDE GECİKME VE LOJİK DEĞERLERİ

Geçiş zamanı lojik değeri entegrede değıştiğinde çıkışa gelen sinyal belirli bir gecikmeye uğrar ve 0 dan 1 e yükselirken olan zamana T_r , 1 den 0 a olan geçişte ise T_f zamanı olarak adlandırırız.



Şekil-17

Lojik kapılarda iki durum vardır ve bunlar lojik 0 ve lojik 1 değeridir. 0 lojik değeri +0 volta yakın değerlere den gelirken 1 lojik değeri +5 volta denk gelmekte aşağıdaki tabloda lojik değerlerin voltaj değerlerini göstermektedir.

Cihaz Tipi	Lojik 0	Lojik 1
TTL	0 - 0.8v	2.0 - 5v (V_{CC})
CMOS	0 - 1.5v	3.0 - 18v (V_{DD})

Çizelge-1

3.11 BOOLE CEBRİ

Cebir diğer matematiksel sistemler gibi sonuç çıkarma prensibine bağlıdır. Bir eleman kümesi, işlem kümesi ve belirli sayıda kanıtlanmamış postulat ile tanımlanabilmektedir. 1854 yılında George Boole, lojiği sistematik bir şekilde ele almak istemiş ve bunun sonucundan günümüzde Boole cebri olarak bilinen bir sistem geliştirmiştir. 1938’ de C. E. Shannon anahtarlama cebri olarak adlandırılan iki değerli bir Boole cebri türü tanıtılarak iki kararlı elektrik anahtarlama devre özelliklerinin bu cebirle temsil edilebileceğini gösterdi. Bu kısımda Boole cebrinin biçimsel tanımı için Huntington tarafından 1904 te oluşturulan postulatları bilip uygulayacağız.

Boole cebiri, şimdi sıralayacağımız Huntington postulatları sağlandığı takdirde, + ve ikili işlemleri olan bir B kümesi üzerinde tanımlanan cebirsel bir yapıdır.

+ işlemine göre kapalılık, + ya göre birim eleman 0 olarak belirlenmiştir. $x+0=x$

* İşlemine göre kapalılık , * ya göre birim eleman 1 olarak belirlenmiştir. $x*1=x$

+ ya göre değişme özelliği $x+y=y+x$

* ya göre değişme özelliği $x*y=y*x$

, + üzerinde dağılma özelliği vardır. $x(y+z)=(x*y)+(x*z)$

+, * üzerinde dağılma özelliği $x+(y*z)=(x+y)*(x+z)$

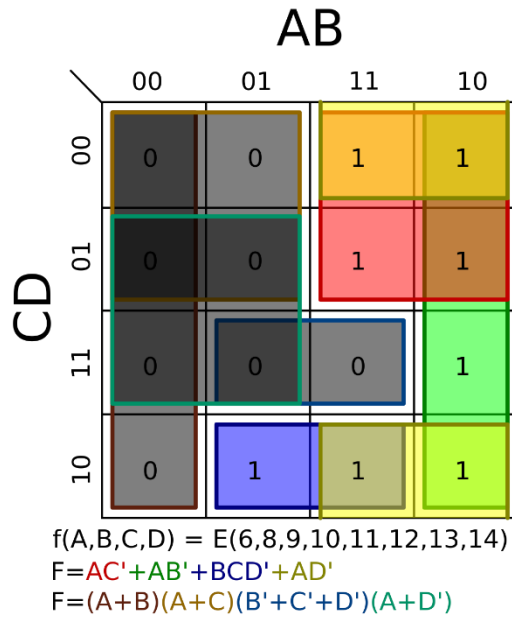
Her x elemanı B içi olsun ve x in değili (x') elemanıdır B’nin böylece (a) $x+x'=1$,
(b) $x*x'=0$ olur.

X, y’ye eşit olmadığı sürece en az iki x,y elaman B vardır.

3.12 KARNAUGH HARİTASI

Bool cebirinde verilen mantıksal gösterimleri sadeleştirmek için kullanılan haritadır. Buna göre bir mantıksal devrenin eleman sayısını azaltmak için de kullanılabilir. Örneğin 3 girişli (3 adet farklı binary (ikili) giriş değeri (0 veya 1 olabilen)) devrede kullanılan “ve” ve “veya” kapılarının sayısını azaltabiliriz.

Bu yöntemde giriş değerlerinin alabileceği bütün alternatifler bir tablo üzerinde gösterilerek bu alternatiflerden hangilerinde çıkış olması isteniyorsa bu değerlere 1 yazılır. Sonuçta yazılı olan 1 rakamları arasında komşuluk incelemesi yapılarak sadeleştirilirler.



Şekil-18

3.13 YARIM TOPLAYICI (LOJİK DEVRELER)

İkilik tabanda verilen iki giriş değerini toplayan devredir. Buna göre A ve B girişleri için aşağıdaki tablo elde edilir. (aşağıdaki tablodaki + işareti önermeler arası veya ile karıştırılmamalıdır. + işareti toplamayı ifade eder)

A	B	E	T
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

yukarıda verilen çizelgede A ve B sayılarının toplamı E ve T değerlerinde gösterilmiştir. Örneğin A 1 ve B 1 değerleri için toplam 2 olmaktadır ($1+1=2$) bu değer in ikilik tabandaki karşılığı 10'dır. 2 çıkış olmasının sebebi toplanan sayıların tek bit ile ifade edilememesindendir.

Yukarıda verilen toplam değerlerini veren devreyi tasarlarken E ve T ikillerini (Bit) ayrı ayrı düşünmek gerekir. Bu toplama işlemini yapan devrenin tasarımında karnaugh haritalarından faydalanılırsa T ve E ikilleri (bit) için aşağıdaki haritalar çizilebilir:

T	A	B
	0	1
0		1
1	1	

E	A	B
	0	1
0		
1		1

Şekil-19

yukarıda verilen şekilde T ve E bitlerinin çıkış değerleri karnaugh haritası üzerinde işaretlenmiştir. Komşuluk durumu olan 1 değeri olmadığı için iki bit değeri de sadeleştirilemeden aşağıdaki önermeler halinde yazılmak durumundadır:

$$T=AB'+A'B$$

$$E=AB$$

3.14 TAM TOPLAYICI (LOJİK DEVRELER)

3 bitlik giriş değerlerinin (iki tabanındaki girişler) toplamını veren devredir. Buna göre A, B ve C girişleri için aşağıdaki tablo elde edilir. (aşağıdaki tablodaki + işareti önermeler arası veya ile karıştırılmamalıdır. + işareti toplamayı ifade eder)

A	B	C	E	T
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

yukarıda verilen çizelgede A, B ve C sayılarının toplamı E ve T değerlerinde gösterilmiştir. Örneğin A 1, B 1 ve C 1 değerleri için (çizelgenin son satırı) toplam 3 olmaktadır ($1+1+1=3$) bu değer için ikilik tabandaki karşılığı 11'dir. 2 çıkış olmasının sebebi toplanan sayıların tek bit ile ifade edilememesindenidir.

Yukarıda verilen toplam değerlerini veren devreyi tasarlarırken E ve T ikillerini (Bit) ayrı ayrı düşünmek gerekir. Bu toplama işlemini yapan devrenin tasarımında karnaugh haritalarından faydalanılırsa T ve E ikilleri (bit) için aşağıdaki haritalar çizilebilir:

T	BC	00	01	11	10
A	0		1		1
A	1	1		1	

E	BC	00	01	11	10
A	0			1	
A	1		1	1	1

Şekil-20

yukarıdaki haritada, üç giriş için toplam değerlerini veren iki farklı çıkış (T toplam ve E elde var) bitlerinin değerleri gösterilmektedir. Bu değerler arası komşuluklar kırmızı daire içerisinde alınmıştır. Bu 1lerden oluşan adalar sadeleştirildiğinde:

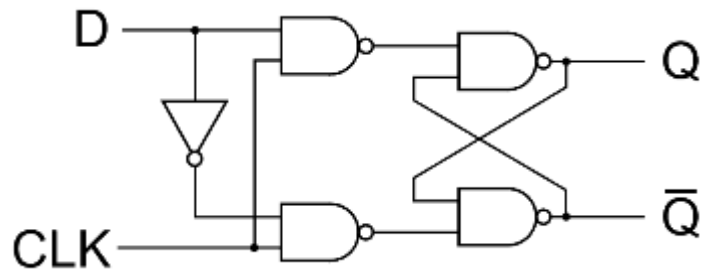
$$T = A'B'C + A'BC' + ABC + AB'C'$$

$$E = AB + AC + BC$$

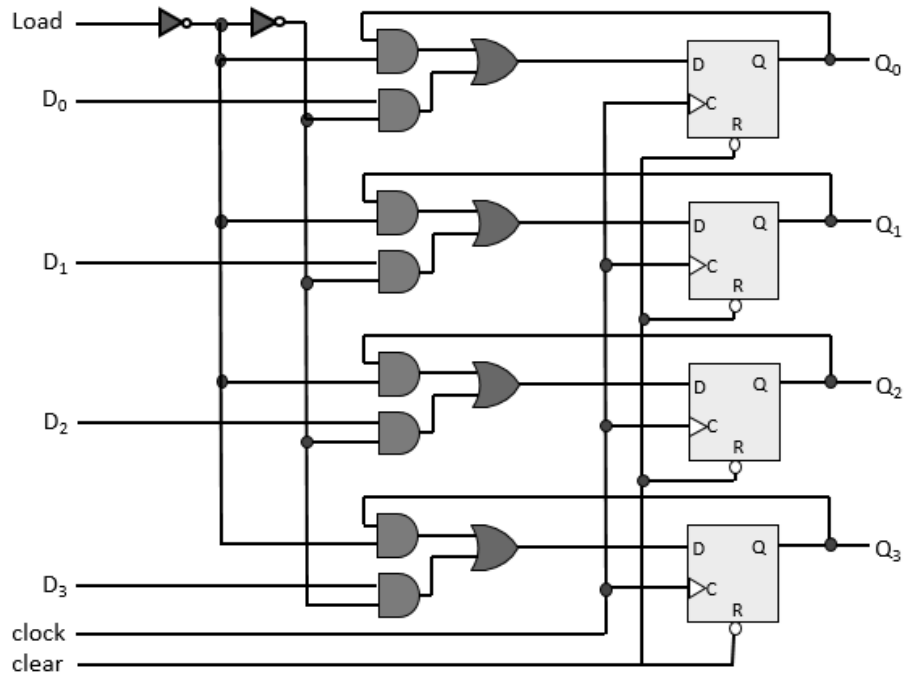
3.15 SAKLAYICILAR

Genel bir ardışıl devre: Saklayıcılar –Ardışıl devre analiz ve sentezi için iyi bir örnektir. Ayrıca daha büyük çaplı ardışıl devrelerin tasarımında kullanılabilirler.

Flip-flop'lar tek bir bit tutarken saklayıcılar daha büyük miktarda veri tutabilmektedir. Saklayıcılar modern işlemci tasarımının merkezidir. –Birden fazla çeşitte saklayıcılar vardır.



Şekil-21



Şekil-22

4.1 CPU DONANIMI

Komut alıřtırma iřlemlerini yapan blmdr. alıřtırma birimi (execution unit) olarak da bilinir. Bu nite komutları alıřtırır ve pipeline (iř hattı) denen yollarla beslenip tamsayıları kullanarak okuma, deėiřtirme ve komut alıřtırma iřlemlerini yapar. ekirdek ierisinde ALU, Genel amalı register, Durum registeri (Status Register-SR) ve Program sayacı (Program counter –PC) bulunmaktadır

ALU (Arithmetic Logic Unit-Aritmetik Mantık Birimi): İřlemci tarafından gerekleřtirilecek matematiksel ve mantıksal iřlemlerin yapıldıėı blmdr. İřlemcinin en nemli kısmını oluřturur. Geliřmiř iřlemcilerde noktadan sonraki sayılar iin matematiksel iřlem yapan FPU (Floating Point Unit–Kayan nokta nitesi) birimi bulunmaktadır. Bu nite tamsayı olmayan floating point (kayar nokta) hesaplamalarından sorumludur.

Kontrol Birimi: İřlemciye gnderilen komutların zlp (komutun ne anlama geldiėinin tanımlanması) iřletilmesini saėlar. İřlemci iindeki birimlerin ve dıřındaki birimlerin eř zamanlı olarak alıřmasını saėlayan kontrol sinyalleri bu birim tarafından retilir. Kontrol nitesinde Komut kaydedici (Instruction Register-IR) ve Komut zc (Instruction Decoder –İD) bulunur.

4.2 ALU (ARİTMETİK MANTIK BİRİMİ)

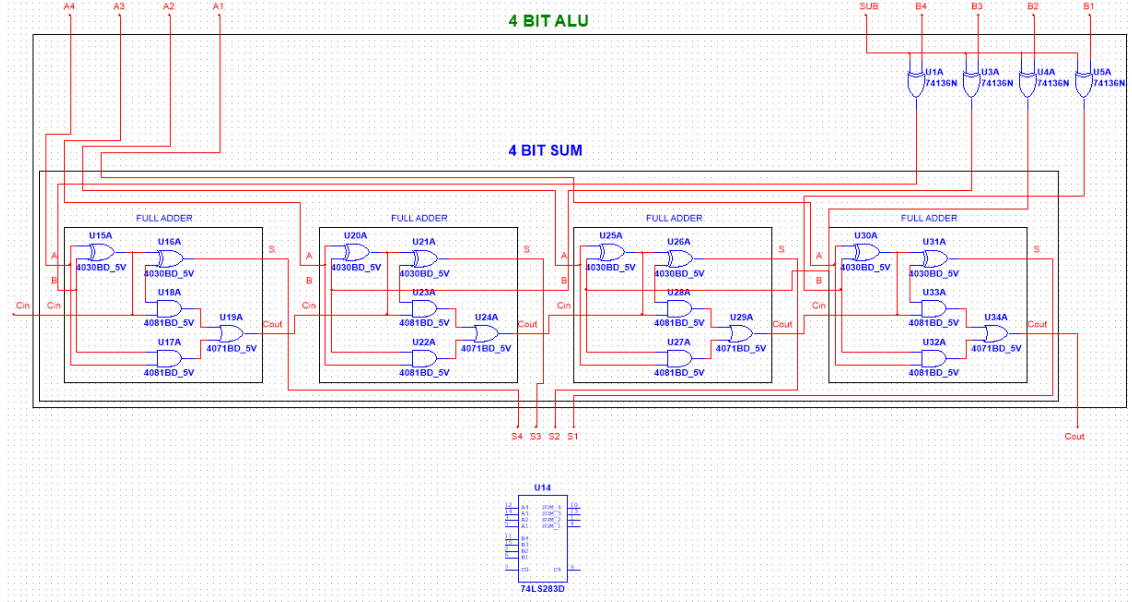
Mikro işlemcinin en önemli kısmını aritmetik ve lojik birimi (ALU) oluşturur (Şekil 2.5). Bu ünite kaydediciler üzerinde toplama, çıkarma, karşılaştırma, kaydırma ve döndürme işlemleri yapar. Yapılan işlemin sonucu kaydediciler üzerinde saklanır. Bazen de yalnızca durum kodu kaydedicisini etkiler. ALU'daki bir işlem sonucunda durum kodu kaydedicisindeki bayrakların birkaçı etkilenebilir veya hiçbiri etkilenmez. Programcı için çoğu zaman ALU'da yapılan işlemin sonucunda etkilenen bayrakların durumu daha önemlidir. Gelişmiş mikro işlemcilerin içindeki ALU'lar çarpma ve bölme işlemlerini yapabilmektedir. ALU'nun işlem yapabileceği en büyük veri, mikro işlemcideki kaydedicilerin veri büyüklüğü ile sınırlıdır. 8 bitlik mimariye sahip bir mikro işlemcideki ALU en fazla 8 bitlik sayılar üzerinde işlem yapar.

ALU'da yapılan aritmetiksel işlemler mikro işlemcinin yapısına göre çeşitlilik gösterebilir. 8-bitlik mimariye sahip bir mikro işlemcide toplama, çıkarma, çarpma, bölme işlemleri ve ondalıklı sayılarla matematiksel işlemler yapılabilmektedir. Gelişmiş işlemcilerde büyük ondalıklı sayılarla işlem yapmak için ayrıca matematik işlemci mevcuttur.

- Mantıksal işlemler
- Mantıksal çarpma VE işlemi
- Mantıksal toplama VEYA işlemi
- Özel VEYA, XOR işlemi
- Değil, NOT işlemi
- Karşılaştırma ($=$, $=<$, $=>$, $<>$ gibi) ve kaydırma gibi işlemler bu üniteye yapılır.
- Sağa veya sola kaydırma ve döndürme işlemleri
- İçerik artırma veya azaltma işlemleri

Bütün bu işlemler teknolojik yapısı değişik kapı ve flip-flop'lardan oluşan bir sistem tarafından yürütülmektedir.

4.3 ALU TASARIMI SONUÇLARI



Şekil-23

Yukarıda görünen 4 bit ALU tasarımında A ve B uçlarından gelen 4 bitlik bilgiler toplanıp S uçlarına gönderilmektedir. Bu ALU birimi toplama ve çıkarma yapmaktadır ancak istersek başka matematiksel işlem veya mantıksal işlem ekliyebiliriz.

Eğer ki SUB kontrol sinyali 0 olursa toplama işlemi yapmaktadır. Aşağıda bu işlemi görebiliriz.

$$S0 = A0 + B0$$

$$S1 = A1 + B1$$

$$S2 = A2 + B2$$

$$S3 = A3 + B3$$

Eğer ki A3 ve B3 lojik 1 değerine sahipse Cout değeri lojik 1 olarak çıktı verir.

4.4 CU (KONTROL BİRİMİ)

Mikro işlemcili sistemde bulunan birimler arasındaki ilişkiyi düzenleyen sinyallerin iletilmesi amacıyla kullanılan hatlar kontrol yolu olarak adlandırılır. Her bir mikro işlemciye ait komut kümesi ve belirli amaçlar için kullanılan sinyallerin farklı olması sebebiyle her mikro işlemcide farklı sayıda hattı içeren kontrol yolu bulunabilir. Kontrol yolunda bulunan sinyaller üç farklı işlemi gerçekleştirmek için kullanılır:

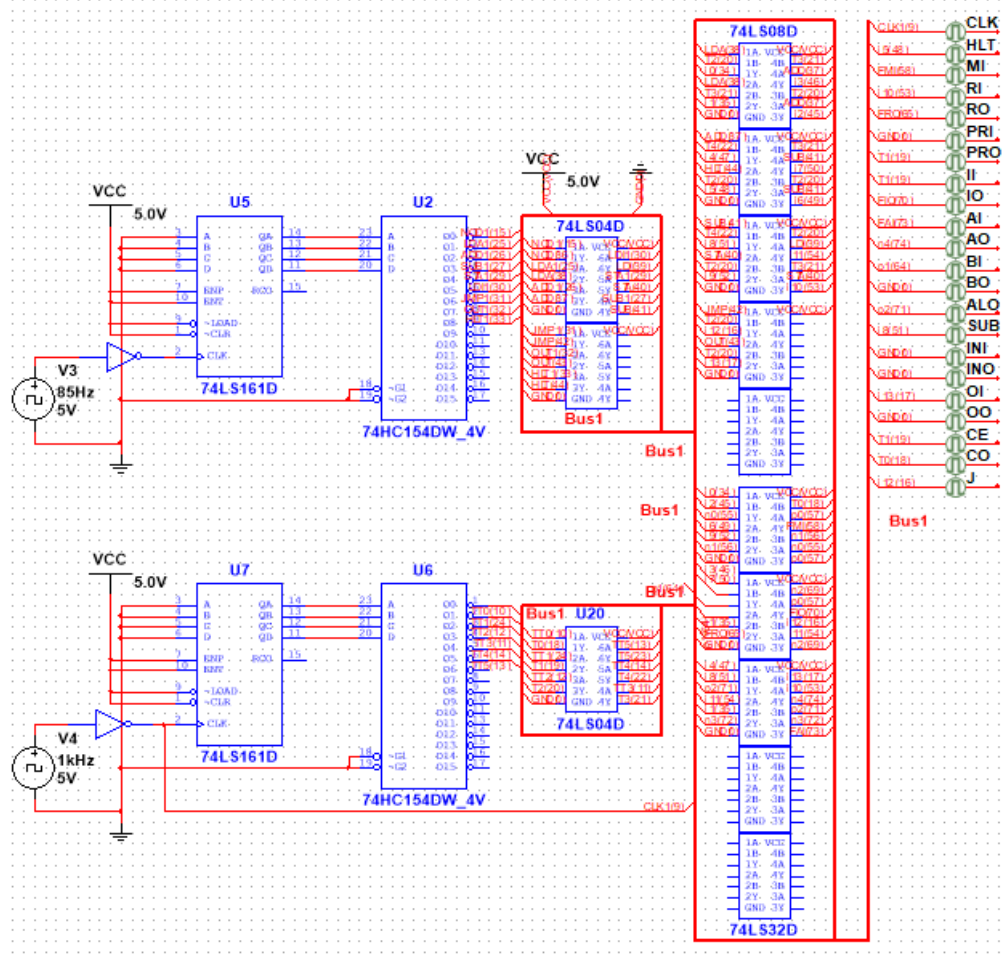
Sinyal seçimi: Sistemde kullanılacak sinyallerin ve sinyallerin uygulanacağı yerin belirlenmesi işlemini gerçekleştiren sinyaller

Yön tayini: Sistemdeki verinin ne yöne gideceğini belirleyen sinyaller (okuma veya yazma)

Zamanlama: Yapılacak işlemlerin sırasını ve zamanlamasını belirleyen sinyaller

Kontrol yolunda bulunan hat sayısı, mikro işlemcinin bit sayısına bağımlı değildir. Kontrol yolunu oluşturan hatların mikro işlemci içersinde ağ şeklinde yayılması sebebiyle kontrol yolu terimi yerine kontrol hatları terimi kullanılabilir. Mikro işlemcili sistemdeki birimlerin çalışması, kontrol hatları üzerinden iletilen tetikleme sinyalleri ile yönlendirilir. Mesela, bir bellekten veri okunacağı zaman, ilgili bellek entegresine aktif olmasını sağlayacak yetkilendirme (CS-Chip select) sinyali ile birlikte, okuma işlemi için gerekli uygun R/W sinyalinin uygulanması gerekir.

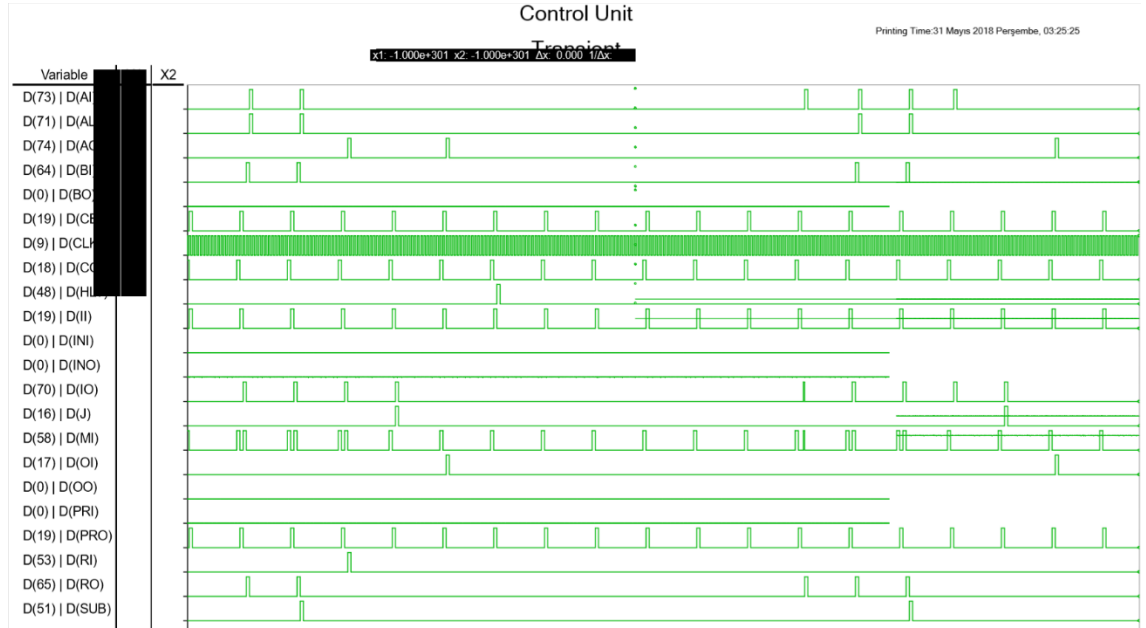
4.5 CU TASARIMI SONUÇLARI



Şekil-24

Bu tasarımımda yaklaşık 9 tane OPCODE yazılmıştır ve bu kodları çalıştıracak devrenin tasarımı yapılmıştır. Bu tasarım sonucu çıkışta işlemcimizde kullanacağımız kontrol sinyallerini görmekteyiz.

4.6 CU ZAMANLAMA HATALARI SONUÇLARI



Şekil-25

Burada kontrol biriminde tasarladığımız devrenin çıkışındaki kontrol sinyallerin sinyalleşmesini görmekteyiz ve bu sinyaller çıkışta gecikme yaşayabilmektedir bu gecikmeler entegrelerin yapısıyla alakalıdır.

Ve bu gecikmeler eşit olmalıdır bunun için gerekli tasarım yapılacaktır.

4.7 OPCODE KÜTÜPHANE OLUŞTURMA

LDA (0000)

- | | |
|-----------|-------|
| 1. CO MI | FETCH |
| 2. PRO II | |
| 3. CE | |
4. IO MI
5. RO AI

Bu akış da LDA komutunun çalıştırılmasını görmekteyiz ilk 3 adımı fetch komutu iken diğer adımlar LDA komutunu oluşturmaktadır.

Bu şekilde diğer Komutları da yazabiliriz.

CO: Sayacı Bus'a verir

MI: Bus da ki değer MAR giriş yapar

PRO: Program da ki değeri Bus'a verir

IO: Instruction saklayıcısındaki değeri Bus'a verir

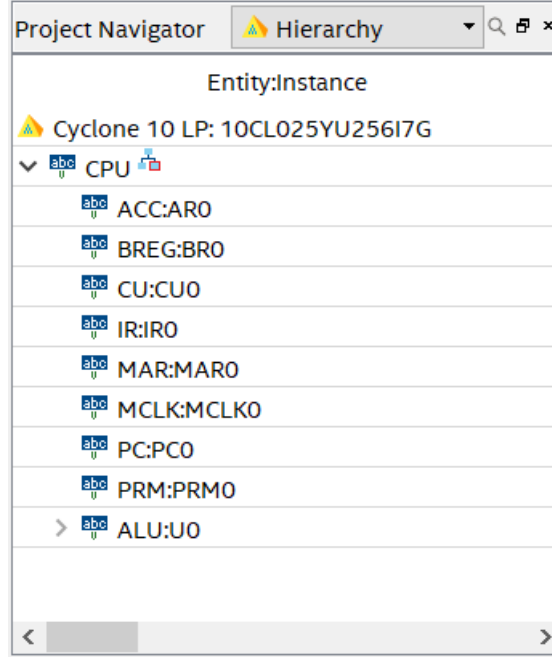
II: Bus da ki değer Instruction saklayıcısına gelir

CE: Sayacı bir arttırır

RO: Ram de ki değeri Bus'a verir

AI: Bus da ki değer A saklayıcısına gelir

5.1 QUARTUS PROGRAMINI İNCELEME



Şekil-26

Intel Quartus Prime , Intel tarafından üretilen programlanabilir mantık aygıtı tasarım yazılımıdır ; Intel'in Altera'yı devralmasından önce bu araca Altera Quartus II adı verildi.

Quartus Prime , geliştiricinin tasarımlarını derlemesini, zamanlama analizi yapmasını, RTL diyagramlarını incelemesini , bir tasarımın farklı uyarılara tepkisini simüle etmesini ve hedef cihazı programlayıcı ile yapılandırmasını sağlayan HDL tasarımlarının analizini ve sentezini sağlar . Quartus Prime, donanım tanımı, mantık devrelerinin görsel olarak düzenlenmesi ve vektör dalga biçimi simülasyonu için VHDL ve Verilog'un bir uygulamasını içerir.

5.2 CPU VERİLOG TASARIMI

Verilog elektronik sistemleri modellemek için kullanılan bir donanım tanımlama dilidir. Verilog (bazen “Verilog HDL” olarak da adlandırılır) analog, sayısal ve karışık işaretli devrelerin tasarımını, doğrulanmasını ve yürütülmesini değişik düzeylerde desteklemektedir. Verilog dilinin tasarımcıları dilin C programlama diline yakın bir sözdizimine sahip olmasını istemişlerdir. Böylece bu dile yatkın olan mühendislerin dili kolayca kullanmasını amaçlamışlardır. Dil küçük/büyük harf duyarlılığına sahiptir ve temel denetim akışının “if” ve “while” gibi anahtar kelimeleri, C'ye benzemektedir. Verilog birkaç temel yönde C'den farklıdır. Verilog bir blok kodu tanımlamak için kıvrık parantezler yerine Begin/End kullanmaktadır.

```
2 //
3 //
4 //
5 //
6 //
7 //
8 //
9 //
10 //
11 //
12 module CPU (data);
13 //inout [3:0] databus;
14 //inout [3:0] adressbus,controlbus;
15 output [3:0] data;
16 wire clk,ce,co,mi,mo,ii,io,cout,carry,pcr,ai,bi,alue,pre,prw,hlt;
17 wire [3:0] databus,adressbus,a,b;
18 wire [2:0] alu;
19 wire [2:0] op;
20 reg r,hld;
21
22 assign data = databus;
23
24 initial begin
25   r=0; hld=0;
26 end
27
28 MAR MAR0 (clk,mi,r,databus,adressbus);
29 IR IRO (clk,ii,io,r,databus,op,databus);
30 ACC AR0 (clk,ai,r,databus,a);
31 BREG BR0 (clk,bi,r,databus,b);
32 ALU U0 (a,b,databus,alu,cout,alue);
33 MCLK MCLK0 (clk,hlt || hld);
34 PRM PRM0 (databus,adressbus,pre,prw);
35 PC PC0 (databus,ce,co,pcr);
36 CU CU0 (clk,r,op,ce,co,mi,mo,ii,io,pcr,ai,bi,alue,pre,prw,hlt,alu);
37 //FLAG F0 (clk,cout,carry,cr);
38
39 endmodule
40
```

Şekil-27

Biz bu bölümde ana CPU modülü oluşturduk alt modüllerle ve burada kullanılan kodlarda output-çıkış pini , wire-ara bağlantı , reg-kaydedici tanımlamaları anlamına gelmektedir.

Assign - iki ara boğlantıyı birleştirebilmek için kullanıldı.

Initial - başlangıç değerleri belirleyebiliriz.

Endmodule – modülü sonlandırma.

5.3 ALU VERİLOG TASARIMI

```
8 //
9
10 module ALU (a,b,s,alu,cout,alue);
11 input [3:0] a,b;
12 input [2:0] alu;
13 output [3:0] s;
14 output cout;
15 input alue;
16 reg [3:0] tt;
17 wire [3:0] bb,ss0,ss1;
18
19 xor (bb[0],b[0],alu[0]);
20 xor (bb[1],b[1],alu[0]);
21 xor (bb[2],b[2],alu[0]);
22 xor (bb[3],b[3],alu[0]);
23
24 FA4 U0(a,bb,ss0,alu[0],cout);
25
26 and (ss1[0],a[0],b[0]);
27 and (ss1[1],a[1],b[1]);
28 and (ss1[2],a[2],b[2]);
29 and (ss1[3],a[3],b[3]);
30
31 assign s = alue? tt : 'bz;
32
33 always @(alu or ss0 or ss1)
34   case (alu)
35     3'b000 : tt = ss0;
36     3'b001 : tt = ss0;
37     3'b010 : tt = ss1;
38     3'b011 : tt = 4'b0000;
39     3'b100 : tt = 4'b0000;
40     3'b101 : tt = 4'b0000;
41     3'b110 : tt = 4'b0000;
42     3'b111 : tt = 4'b0000;
43   endcase
44 endmodule
45
```

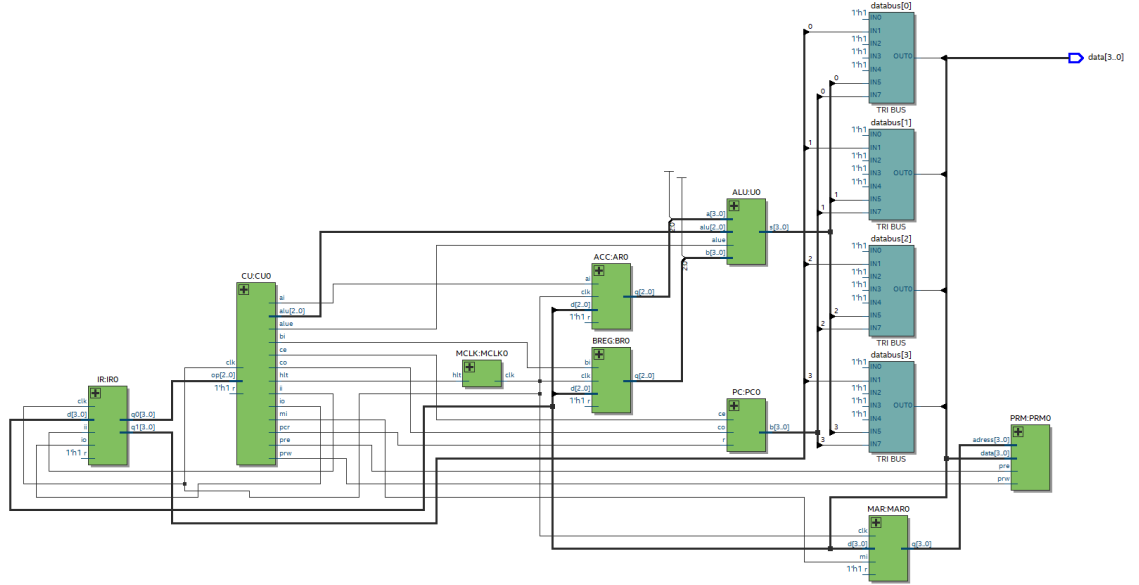
Şekil-28

Bir önceki konuda gördüğümüz CPU verillog tasarımındaki ALU alt modülünün verillog dilinde tasarımını görüyoruz.

ALU tasarımını tam toplayıcıları kullanarak FA4 modülünde olduğu gibi 4 bit tam toplayıcıyı alu kodu ile hangi işlem yapacağını seçeceğiz ve sonuç olarak a ve b girişlerini alu kontrol sinyalleri yardımıyla işlem yaptırıp alue giriş pini ile s çıkışına sonuç verilerini göndeririz.

Ve son olarak cout carry biti varsa flag registerlarına kayıt edilir.

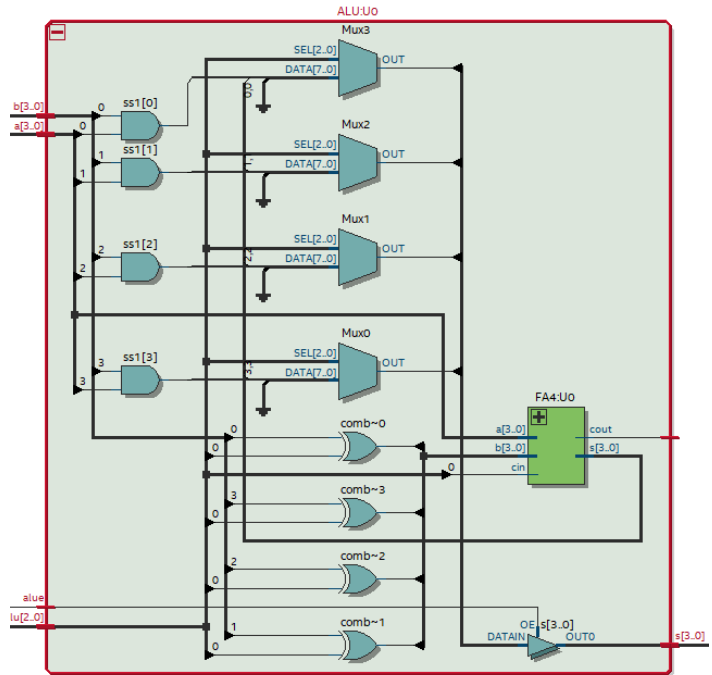
5.4 CPU RTL GÖRÜNTÜLEYİCİSİ



Şekil-29

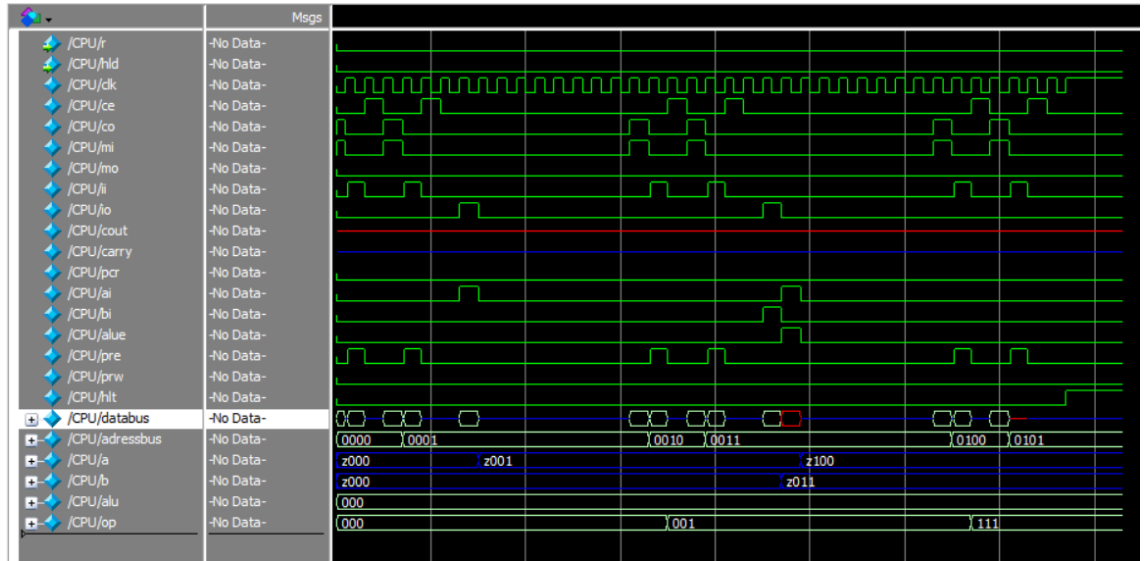
Gerçek zamanlı olarak CPU verilog olarak yazdığımız program kodları modülleri görselleştirerek daha iyi görüp analiz etmemize yarar ve veri yollarını burada daha iyi görebiliriz.

Ve yanda ALU alt modülünün içinin görselleştirilmiş halini görmekteyiz.



Şekil-30

5.5 ALTERA SİM SİMULASYON SONUÇLARI



Şekil-31

Bu görselde ise bu oluşturduğumuz modülü simülasyon yapıyoruz ve gelen sonuçlar üzerinden değerlendirme yaparak gerekirse tasarımda değişiklikler ve düzenlemeler yapılır.

Örneğin simülasyonda görünen kırmızı sinyaller bilinmeyen değer anlamına gelir ve pek istediğimiz durum değildir bunun tasarımda değişiklikler yapıp daha stabil çalışan bir sistem tasarlamalıyız.

6. KAYNAKLAR

<http://diyot.net/transistor/>

<http://www.elektrikport.com/universite/transistorler-nasil-calisir-elektrikport-akademi/8946#ad-image-0>

<http://www.odevsel.com/egitim/2583/cmos-nedir-ttl-nedir-cmos-ile-ttl-arasindaki-farklar-nelerdir.html>

<https://www.muhendisbeyinler.net/lojik-kapi-devreleri/>

Digital Computer Electronics (Albert Paul Malvino)

<http://www.elektrikport.com/teknik-kutuphane/boole-cebri-sayisal-devreler/15399#ad-image-0>

<http://bilgisayarkavramlari.sadievrenseker.com/2007/12/07/karnaugh-haritasi-karnaugh-map/>

<http://bilgisayarkavramlari.sadievrenseker.com/2007/12/08/tam-toplayici-full-adder/>

<http://www.yildiz.edu.tr/~bataslar/Saklayici.pdf>

<http://www.selcuk.edu.tr/dosyalar/files/054003/%C4%B0%C5%9Flemciler%2>

7. TEŞEKKÜR

Bu projenin hazırlanmasında emeği geçen, donanım ve yazılım aşamasında yardımlarını esirgemeyen, yapım aşamasında yol gösterici olan kıymetli hocamız Dr. Gökhan Yenikaya'ya teşekkür ve şükranlarımızı sunarız.

8. ÖZGEÇMİŞ

HÜSEYİN KAYA

1996 yılında Manisa’da doğdu. Turgutlu Anadolu Lisesinden 2014 yılında mezun oldu. 2014 yılında Uludağ Üniversitesi Elektrik Elektronik Mühendisliği Bölümü’nü kazandı. Halen öğrenim hayatına devam etmektedir.