T.C
HARRAN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

FPGA TABANLI MIPS MİMARİSİNDE 32-BİT YERLİ MİKROİŞLEMCİ TASARIMI

Yusuf EREN

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ŞANLIURFA
2021

İÇİNDEKİLER

ÖZET i
ABSTRACT ii
TEŞEKKÜR iii
ŞEKİLLER DİZİNİ iv
ÇİZELGELER DİZİNİ vi
SİMGELER ve KISALTMALAR DİZİNİ vii
1. GİRİŞ 1
2. ÖNCEKİ ÇALIŞMALAR 2
3. MATERYAL ve YÖNTEM 6
3.1. Mikro İşlemci Mimarileri 6
3.1.1. Cisc 6
3.1.2. Risc 7
3.2. MIPS Mimarisi 8
3.2.1. ALU (Aritmetik Lojik Ünite) 10
3.2.1.1. Toplayıcı Devre 11
3.2.1.2. Fark Alıcı Devre 14
3.2.2. Çarpıcı Devre 17
3.2.3. Bölücü Devre 20
3.2.4. Instruction Memory 21
3.2.5. Program Counter 22
3.2.6. Control Unit 22
3.2.7. Register File 22
3.2.8. Data Memory (RAM) 23
3.2.9. Sign/Zero Extender 23
3.3. MIPS ISA(Instruction Set Architecture) 23
3.4. Yöntem 26
3.4.1. FPGA 26
3.4.2. Verilog 29
3.4.3. ModelSim 37
3.4.4. Lojik Analizör 38
3.4.5. Power Play 39
3.4.6. Signal Tap Logic Analyzer 41
4. ARAŞTIRMA BULGULARI ve TARTIŞMA 46
4.1. Program Counter 48
4.2. ALU 48
4.3. Control Unit 49
4.4. Çevresel Birimler 50
4.4.1. I/O Unit (Giriş/Çıkış Birimi) 50
4.4.2. SPI 53
4.4.3. UART 54
4.4.4. Timer/PWM/Prescaler 55
4.5. Mikroişlemci Genel Yapısı 57
4.6. Test ve ölçümler 59
5. SONUÇLAR ve ÖNERİLER 63
KAYNAKLAR 66
ÖZGEÇMİŞ 68
EKLER 69
Ek-1 Çevresel Birimlerin RTL gösterimi 69
Ek-2 CPU Mimarisi RTL gösterimi 70

ÖZET

Yüksek Lisans Tezi

FPGA TABANLI MIPS MİMARİSİNDE 32-BİT YERLİ MİKROİŞLEMCİ TASARIMI

Yusuf EREN

Harran Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Doç. Dr Hasan GÖKTAŞ
Yıl: 2021, Sayfa: 70

Mikro işlemciler ortaya çıktıkları günden bu zamana kadar insanların hayatını kolaylaştıran teknolojik aygıtların temel bileşenlerini oluşturmaktadır. Bu bağlamda mikro işlemciler teknolojiyi yönlendiren, geliştiren gömülü devrelerdir. Mikro işlemciler temel olarak iki tip mimariye sahiptir.Bu mimariler CISC(Complex Instruction Set Computer) ve RISC(Reduced Instruction Set Computer) olarak isimlendirilir.CISC mimarisi oldukça karmaşık bir mimariye ve yüksek güç tüketimine sahip iken RISC mimarisi daha az karmaşık bir mimariye ve düşük güç tüketimine sahiptir. Ülkemiz çapında mikro işlemciler yurt dışından temin edilerek ilgili alanlarda kullanılmaktadır. Bu alanlar arasında askeri, tıbbi ve havacılık alanları hassasiyeti ve gizliliği esas alan, öncelikli alanlardır. Bu anlamda yurt dışından temin edilen mikro işlemcilerin bu esaslara aykırı düştüğü durumlara yol açabilmektedir. Öte yandan dünyada ki enerji ihtiyacının günden güne artması sebebiyle düşük güç tüketen elektronik aygıtlar giderek önem kazanmaktadır. Bu çalışma da; RISC mimarisini temel alan MIPS (Microprocessor without Interlocked Pipline Stage) mimarisinde, 32 bit veri, 12 bit adres yolu ve 32 bit komut setine sahip Türkiye'nin yerli mikro işlemcisi tasarımı yapılmış, FPGA üzerinde gerçeklenmiş ve testleri tamamlanmıştır. Tasarlanan mikro işlemci CISC mimarisinin işlem gücüne ve RISC mimarisinin düşük güç tüketimi ve esnekliğine sahiptir.

ANAHTAR KELİMELER: Mips Mimarisi, RISC, FPGA, Mikroişlemci, Mikrodenetleyici, Mikroişlemci Mimarileri

ABSTRACT

MSc Thesis

32-BIT MICROPROCESSOR DESIGN IN FPGA-BASED MIPS ARCHITECTURE

Yusuf EREN

Harran University
Graduate School of Natural and Applied Sciences
Department Electrical and Electronic Engineering

Supervisor: Associate Prof. Dr. Hasan GÖKTAŞ
Year: 2021, Page: 70

Microprocessors are the basic components of technological devices that have made people's lives easier since the day they were invented. Moreover, microprocessors are the integrated circuits that leads and develop the technology. They basically have two types of architecture: CISC (Complex Instruction Set Computer) and RISC (Reduced Instruction Set Computer). In contrast to RISC, CISC architecture has a highly complex architecture and high-power consumption. Microprocessors are imported from abroad and have been used in many fields in Turkey. Military, medical and aviation are the most essential ones among the application fields as they require confidentiality, accuracy and sensitivity. This fact makes it mandatory to design and fabricate our own microprocessors. Furthermore, the mobile devices getting more attraction now adays especially with IOT, therefore low power design is one of the critical design parameters. In this study, the domestic microprocessor with RISC based MIPS (Microprocessor without Inter-locked Pipline Stage) architecture was designed, realized and tested on FPGA board. It has 32-bit data bus, 12-bit address bus, and 32-bit instruction set. The microprocessor designed in this work has processing power of the CISC architecture while offering the low power and flexibility of RISC architecture

KEY WORDS: FPGA, Microprocessor, MIPS architecture, RISC architecture, Low power MIPS

TEŞEKKÜR

Tezin konusunun seçiminde, uygulamasında ve çalışmamda yardımlarını esirgemeyen danışmanım Doç. Dr. Hasan GÖKTAŞ'a; çalışmamda yardımcı olan laboratuvar arkadaşlarım Yasin ASLAN'a, Cihan EFİL'e teşekkür ederim. Ayrıca tez çalışmam boyunca bana sonsuz destek veren aileme teşekkür ederim.

ŞEKİLLER DİZİNİ

Sayfa No

Şekil 3.1 CISC Mimarisi (Binaryterms) 7
Şekil 3.2 RISC mimarisi (Binaryterms) 8
Şekil 3.3 MIPS mimarisi blok diyagramı (Omran,2017) 10
Şekil 3.4 ALU blok gösterimi 11
Şekil 3.5 Half Adder lojik diyagramı 12
Şekil 3.6 Full adder lojik gösterimi (URL-3) 13
Şekil 3.7 Dört bitlik toplayıcı devre blok gösterimi (URL-3) 14
Şekil 3.8 Half Substractor lojik gösterimi (URL-3) 14
Şekil 3.9 Full Substractor lojik gösterimi(URL-3) 15
Şekil 3.10 4-Bit Adder/Substractor blok gösterimi(URL-5) 17
Şekil 3.11 Binary çarpma işlemi basamakları(URL-7) 18
Şekil 3.12 Çarpıcı devre lojik ve blok gösterimi(URL-4) 19
Şekil 3.13 Binary bölme işlemi 20
Şekil 3.14 Binary bölücü devre blok gösterimi 21
Şekil 3.15 R type ISA 24
Şekil 3.16 I Type ISA 25
Şekil 3.17 J Tpye ISA 25
Şekil 3.18 FPGA çipi blok gösterimi (Toronto University) 27
Şekil 3.19 Mantık bloğu iç yapısı blok gösterimi(Toronto University) 27
Şekil 3.20 ALTERA Max10 Neek FPGA kartı (URL-9) 28
Şekil 3.21 EP2C5T144C8N Altera Fpga 29
Şekil 3.22 Quartus yazılımı anasayfa görüntüsü 30
Şekil 3.23 Derleme raporu sekmesi 31
Şekil 3.24 Flow Summary raporu 32
Şekil 3.25 TimeQuest Analizi 33
Şekil 3.26 Slack histogramı 34
Şekil 3.27 Data delay waveform 34
Şekil 3.28 RTL Netlist gösterimi 35
Şekil 3.29 And kapısı lojik gösterimi(URL-2,2020) 36
Şekil 3.30 6 girişli and kapısı (URL-6) 36
Şekil 3.31 32-bit girişli and kapısı 37
Şekil 3.32 ModelSim çıktıları 38
Şekil 3.33 Power Play Analyzer Tool ekran görüntüsü 40
Şekil 3.34 Power Play Quartus termal güç raporu 40
Şekil 3.35 Power Play kaynaktan çekilen akım Quartus raporu 40
Şekil 3.36 Signal Tap logic analyzer ekran görüntüsü 42
Şekil 3.37 Test parametrelerinin belirlenmesi 43
Şekil 3.38 Sinyal formlarının seçimi 44
Şekil 3.39 Sinyal konfigürasyonu 44
Şekil 4.1 Kogge Stone Adder 48
Şekil 4.2 Paralel Prefix Adder karşılaştırması(BAIS, 2016) 49
Şekil 4.3 I/O registerları 51
Şekil 4.4 Tri State buffer gösterimi 51
Şekil 4.5 Giriş çıkış kontrol lojik gösterimi 52
Şekil 4.6 I/O birimi blok gösterimi 52
Şekil 4.7 SPI Slave birimi 53
Şekil 4.8 UART Rx Tx birimleri 55
Şekil 4.9 Timer modülü blok gösterimi 56
Şekil 4.10 Tasarlanan mikroişlemci blok gösterimi 57
Şekil 4.11 Tasarlanan mikro işlemci blok diyagramı 58

Şekil 4.12 Kogge-Stone adder osiloskop çıktısı 59
Şekil 4.13 Kogge-Stone adder analizör çıktısı 60
Şekil 4.14CPU Simülasyonu 61
Şekil 4.15 Adress , Data ve Control Bus simülasyonu 61

ÇİZELGELER DİZİNİ

Sayfa No

Çizelge 3.1 Half Adder doğruluk tablosu 12
Çizelge 3.2 Full Adder doğruluk tablosu..... 13
Çizelge 3.3 Half Substractor doğruluk tablosu..... 15
Çizelge 3.4 Full substractor doğruluk tablosu 16

SİMGELER ve KISALTMALAR DİZİNİ

µm	Mikro Metre
AB	Adressbus
ADC	Analog to Digital Converter
ALU	Aritmetic Logic Unit
CPU	Central Processing Unit
CU	Control Unit
DB	Databus
DM	Data memory
FP	Floating Point
FPGA	Field Programable Gate Array
GP	General Purpose
IM	Instruction Memory
ISA	Instruciton Set Architecture
mA	Mili Amper
MHz	Megahertz
mV	Mili Volt
mW	Mili Watt
ns	Nano saniye
PC	Program Counter
PWM	Pulse Width Modulation
RAM	Random Acces Memory
RF	Register File
RTL	Register Transfer Level
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
V	Volt

1.GİRİŞ

Mikro-işlemciler günümüzde hemen her elektronik aygıtta kullanılan tümleşik devrelerdir. Temelde mantıksal işlemleri yaptırabileceğimiz insanlarla makinelerin haberleşmesine olanak sağlayan mikro-işlemciler ilk ortaya çıktıklarında bir programlama ilkesine dayandırılmadan, girişlerine gönderilen elektriksel sinyalleri işleyerek hesap yapmaktaydılar. Bura da sinyallerin gönderilmesini sağlayan mekanik anahtarlar bulunmakta ve bu şekilde mantıksal 0 ve 1 ikilik sistemi oluşturularak basit dört işlem yaptırılabilmekteydi. Ancak o bu şekilde mekanik olarak kontrol edilen bir işlemcinin verimi oldukça düşük ve maliyeti oldukça fazladır. Bu sebepten dolayı işlemcilerin kullanımı oldukça enderdir.

Mikro-işlemcilerin yaygınlaşması ve hesaplama güçleri, transistörün keşfi ile büyük bir ivme kazanmıştır. Çünkü mekanik anahtarların yerine transistörlerin anahtarlama özelliklerinden faydalanarak çok daha hızlı hesaplamalar yapılabiliyordu. Bu safhada mikro-işlemciler daha çok mantıksal işlemlerin yapılabildiği mantıksal kapı olarak adlandırılan entegreler halinde kullanılmaktaydı. Yani mantıksal kapılarla oluşturulan ve dört işlemi gerçekleştirebilecek halde bulunan bu mikro-işlemciler algoritma ve programlama yapılmadan girişlerine uygulanan değerleri işleyerek çıkışlarına ileten yapılardı. Bu yapılardaki esas problem, yapılacak işlemlerin çeşitliliği kadar farklı tasarıma sahip mikro işlemcinin üretilmesidir.

Bu soruna çözüm olarak 1950'li yılların başlarında Vonn-Neumann programlanabilir hafıza sistemini ortaya atmıştır. Bu bağlamda yeni mikro işlemciler programlanabilir bir hal almaktadır. Üretimi yapılan her bir mikro-işlemcinin dinamik olarak programlanabilmesi birçok farklı uygulamada kullanılabilmesine ve bu sayede de mikro-işlemciler giderek yaygınlaşmaya başlamıştır. Daha sonra mikro üretim teknikleri kullanılarak üretilen mikro işlemciler bugün ki zamanına gelmiştir. Elbette ki bu süreçte farklı mikro işlemci mimarileri ve tasarım üretim teknikleri ortaya çıkmıştır.

2.ÖNCEKİ ÇALIŞMALAR

Mikro işlemcilerin temeli bugün kullanılan bütün teknolojik ürünlerin de temeli olan silikonun keşfiyle atılmıştır. Berzelius(1823) tarafından silikon keşfedilmiştir. Daha sonraları Tesla (1923) tarafından elektriksel lojik devreler bir diğer ifade ile kapılar patentlenmiştir. Bu lojik devre, elektromekanikal anahtarlarla kontrol edilmektedir.

Transistörün keşfiyle Bell Laboratories (1947) bu elektromekanik anahtarların yerine transistörlerle anahtarlama fikrinin gelişmesi ve hayata geçirilmesi için çalışmalara başlanmıştır.

Transistörler kullanılarak geliştirilen ilk çalışan tümleşik devre Noyce(1958) tafarından tasarlanmış olup Texas Instruments firmasının temellerini atmıştır. Texas Instruments günümüzde de tümleşik devreler tasarlayan endüstriyi değiştirip geliştiren bir firmadır.

İlk TTL (Transistor-Transitor Logic) tekonolojisiyle tasarlanan yüksek sayıda lojik kapı içeren tümleşik devre tasarımı Sack ve ark. (1964) tarafından yapılmıştır. Bu gelişme mikro işlemcilerin ortaya çıkışına temel hazırlamıştır.

Bu tasarımların mikro teknikler kullanılarak üretilebilmesi için Beelitz ve ark. (1967) tarafından VLSI (Very Large-Scale Integration) tekniği geliştirilerek direkt olarak silikon plakalar üzerinde transistör tasarımı yapılmaya başlanmıştır.

Noyce ve Moore (1968) tarafından INTEL kurulmuştur. INTELve Hoff tarafından 1971 yılında (URL-7) dünyanın ilk mikro işlemcisi olan Intel 4004'ü tasarlamış ve üretmiştir. 4004 basit anlamda CISC mimarisinde üretilmiş bir işlemci olup 2300 adet transistöre ve 640 byte hafızaya sahiptir.

Intel 1973 yılında (URL-7), 8008 olarak adlandırdığı 4004'e göre görece daha performanslı ikinci mikro işlemcisini duyurmuştur. Intel 8008 ile birlikte mikro işlemcilerin uygulama alanları genişlemiş ve sensörlerden veri okuma, motor sürme gibi otomasyon uygulamalarında kullanılmaya başlanmıştır (Seim,1973; Seim,1975). Ayrıca mikro bilgisayarlaralar olan mikro denetleyiciler ortaya çıkmaya başlamıştır. (Davis,1973)

Cohen ve David (1980) tarafından Berkley üniversitesinde, RISC-I (Reduced Instruction Set Computer) açık kaynak kodlu mikro işlemci mimarisi duyurulmuştur. Bunun ardından bu mimariye sahip mikro işlemciler endüstriyel anlamda kullanılmaya başlanmıştır.

Hennessy tarafından 1981 yılında (URL-7) Standford üniversitesinde Standford MIPS mimarisi geliştirilmiştir. Elbette ki bu mimari RISC tabanlı bir mimaridir. Ancak RISC mimarisindeki yetersizlikler (Veri yolu, komut seti bit genişliklerinin az olması) iyileştirilmiştir. Hennessy tarafından 1984 yılında (URL-7)MIPS Computer Systems Inc. kurulmuştur.

MIPS Inc. tarafından 1986(URL-1) yılında ilk R2000 mikro işlemcisi üretilmiştir. Bu mikro işlemci Unix Workstation olarak adlandırılan bilgisayarlarda kullanılmaya başlanmıştır.

Bakoglu ve ark. (1990) tarafından IBM firmasında ilk RISC tabanlı sistem tasarımı yapılmış ve IBM tarafından üretilen aygıtlarda kullanılmaya başlanmıştır. Özellikle taşınabilir cihazların ortaya çıktığı bu dönemlerde RISC mimarisindeki mikro işlemciler oldukça verimlilik sağlamıştır.

MIPS Inc 1991 yılında (URL-1) R4000 mikro işlemcileri tasarlayıp üretmiştir.Toshiba gibi firmalar tarafından kullanılmaya başlanılan bu mikro işlemci ondalık hesaplama birimine sahip ilk MIPS mikro işlemcisidir. 250 Mhz çalışma frekansına sahip olan bu mikro işlemcinin kullanım alanları endüstriyel anlamda önem kazanmıştır.

Pierson ve ark. (1992) tarafından MIPS mimarisine sahip R6000 mikro işlemcisi tasarlanmıştır. R4000 mikro işlemcisine göre performans ve optimizasyon iyileştirmelerini içermektedir.

MIPS Inc. 1994 yılında (URL-1) R8000 mikro işlemcisi duyurulmuştur. Bu mikro işlemci 3.6 milyon transistöre sahip ve 0.7 μm silikon teknolojisiyle üretilmiştir. 591 adet pine sahip ve 13W güç tüketimi vardır.

MIPS Inc. tarafından 1994 yılında (URL-1)R3000 mikro işlemciler Sony Play Station, oyun konsollarında kullanılmıştır. 5 aşama pipline yapısına sahip bu mikro işlemci içerisinde dahili bir grafik görüntüleme birimi vardır. 0.35μm üretim teknolojisine sahiptir.

Yeager(1996) tarafından MIPS R10000 mikro işlemci tasarımı yapılmıştır. Bu mikro işlemci 64-bit veri yollarına sahip 4 farklı komut setine sahiptir. Ayrıca gecikme değerleri R6000 mikro işlemcilere göre oldukça azdır. Gerçek dünya uygulamalarında kullanılabilecek bir mikro işlemci tasarımıdır.

MIPS Inc. tarafından 1996 yılında (URL-1) R10000 mikro işlemcisi duyrulmuştur. Bu mikro işlemci Nintendo 64 oyun konsolunda kullanılmıştır. 0.35 μm üretim teknolojisine sahiptir. 527 adet seramik yapıda pine sahiptir.15-20 W arasında güç tüketimini vardır.

Martin (1997), tarafından MIPS R3000 mikro işlemcisi duyrulmuştur. 7 watt güç tüketimine sahip bu mikrı işlemci, dönemine göre oldukça yüksek komut işleme hızına sahiptir.

MIPS Inc. tarafından 1999 yılında (URL-1)duyurulan R5000 mikro işlemciler Sony Play Station 2 oyun konsolunda kullanılmıştır.

MIPS mimarisine sahip mikro işlemciler 2009 yılında android tabanlı aygıtlarda ve Linux işletim sistemi tabanlı aygıtlarda kullanılmaya başlanmıştır. Günümüzde

MIPS firması birçok farklı isimle bilinmektedir. 2013 yılında Imagination Technologies tarafından satın alınmış olup akıllı telefon ve otomobil sektöründe kullanılmak üzere mikro işlemci tasarımları yapmaya devam etmiştir. Genel işleyiş olarak tasarımlarını başka firmalara satmaktadırlar. 2009 yılından bu yana üretilen birçok akıllı telefonlarda kullanılan mikro işlemciler MIPS mimarisine sahiptir. Günümüzde üretilen otonom ve elektrikli araçlardan olan Tesla otomobillerinde MIPS mimarisindeki mikro işlemciler kullanılmaktadır. Günümüzde tasarlanan MIPS mimarisine sahip mikro işlemciler M- Sınıfı M5000 serisi, I-Sınıfı, I7000 serisi ve P sınıfı P5000 serisi olarak adlandırılmaktadır. Bu mikro işlemciler RISC mimarisine sahip en üst düzey performansa sahip mikro işlemcilerdir. Oldukça geniş bir kullanım yelpazesine sahiptir.

Endüstriyel olarak kullanımlarının yanı sıra MIPS mimarisi üzerinde akademik olarak çalışmalar da günümüzde devam etmektedir. Grafik işleme birimleri dahil edilerek MIPS işlemcilerinin kullanım esnekliği arttırılmıştır. (Novkovic ve ark.,2020)

3.MATERYAL ve YÖNTEM

Tasarlanan mikro-işlemci 12 bit adres ve 32 bit veri yoluna sahip, 32 bit komut setine sahiptir. Yapısal olarak RISC tabanlı mikro-işlemci mimarilerinden olan MIPS(Microprocessor without Interlocked Pipelined Stages) mimarisi kullanılmıştır. Bu mimaride bazı kararsızlık durumları gözlemlenip bu kararsızlıklar tasarımda iyileştirilmiştir.

Tasarlanan mikro işlemcinin bütün birimleri FPGA (Field O) üzerinde gerçeklenmiş ve testleri yapılmıştır. Ayrıca ilgili birimler lojik analizör ile test edilerek işlemcinin gerçeklikle olan bağıntısı analiz edilmiştir. Tasarım da INTEL'in yayımladığı Quartus ve Modelsim yazılımları, verilog donanım tanımlama dili kullanılmıştır.

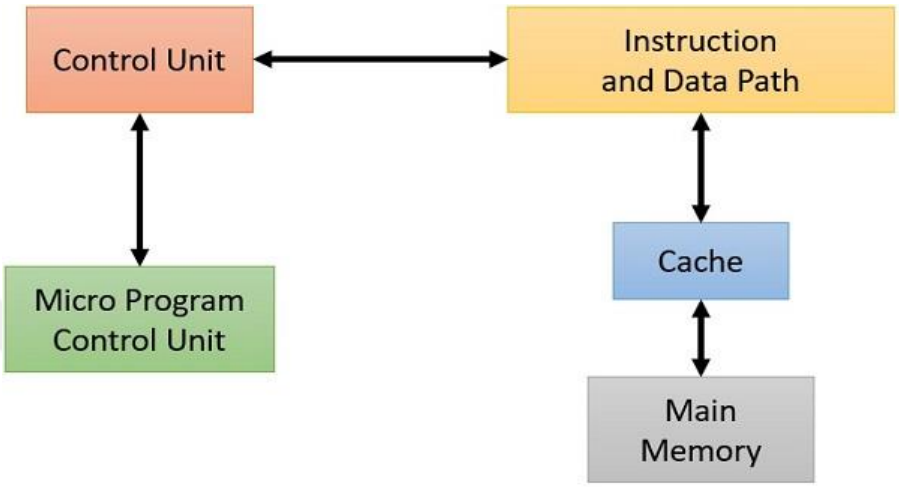
3.1. Mikro İşlemci Mimarileri

Mikro işlemciler ortaya çıkmaya başladıkları günden bu yana temel olarak iki farklı mimari üzerin de çalışılmış ve bunlarla birlikte farklı varyasyonlara sahip mikro-işlemciler üretilmiştir. Bu mimariler CISC (Complex Instruction Set Computer) ve RISC (Reduace Instruction Set Computer)'dir

3.1.1. **Cisc**

CISC mimarisi ilk mikro-işlemci mimarisidir. Karmaşık ve uzun komut setlerine sahip olduklarından dolayı hafıza kullanımları düşüktür. Ortaya çıktığı dönemlerde hafıza çiplerinin üretilmesi maliyetli ve zahmetli bir iş olduğundan düşük hafıza tüketimi ön plandadır. Mikro işlemciye gönderilen komutlar öncelikle mikro kodlara ayrılırlar ve bu şekilde saklanırlar. Bu bağlamda komut işleme döngüleri oldukça uzun olabilmektedir.

Şekil 3.1(a) da görüldüğü üzere CISC mimarisinde kodlar önce makine kodlarında daha sonra mikro kodlara devamında mikro komutlara ve mikro komutların çalıştırılması aşamasına geçmektedir. Bu bir komutun işlenebilmesi için 4 farklı aşamadan geçmesi gerektiği anlamına gelse de hafıza kullanımı bu şekilde en aza indirgenmiştir.



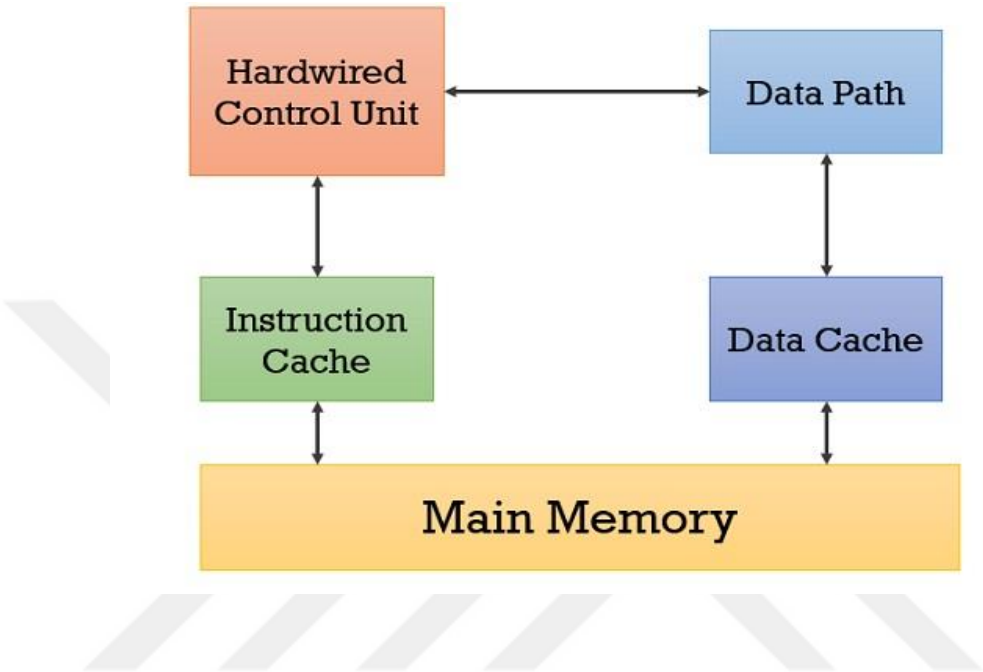
Şekil 3.1 CISC Mimarisi (Binaryterms)

Şekil 3.1 CISC mimarisinin basit blok diyagramını göstermektedir. CISC mimarisinde tasarlanan mikro-işlemciler genellikle donanım ağırlık çalıştıklarından dolayı derleyicilere ihtiyaç duymadan da çalışabilirler. Ayrıca direkt olarak ana hafıza birimlerine erişip okuma yazma yapabilirler bu sayede oldukça hızlı çalışırlar.Bu mimariyi kullanan mikro işlemcilere örnekler şöyledir; Ryzen 5 3600, INTEL i5 9100F, INTEL 8051 vs.

3.1.2. **Risc**

İndirgenmiş komut setine sahip bu mikro işlemciler CISC mikro işlemcilerin aksine oldukça basit komut setlerine sahiptirler. Basit olduklarından dolayı yazılacak komut satırlarının fazla olmasına sebebiyet verirler. Bu bağlamda kodları CISC mimarisine göre daha hızlı işlerken komut satırlarının fazla olması dezavantaj olarak kabul edilir. Ayrıca hafıza kullanımları çok daha fazla olmasıyla birlikte ana hafızadan okuma yazma yapmak için yazmaçlara ihtiyaç duyarlar.

RISC mimarisi yazılım ağırlıklı bir mimaridir yazılımın desteğine her zaman ihtiyaç duyarlar ve bu sebepten dolayı derleyiciler olmadan kodlamak olanaksızdır. Bu bağlamda donanım ağırlıklı uygulamalar için kullanışlı değillerdir. Ancak ortaya çıktıkları dönemlerde hızlı bilgisayarlarda kullanılmışlardır.



Şekil 3.2 RISC mimarisi (Binaryterms)

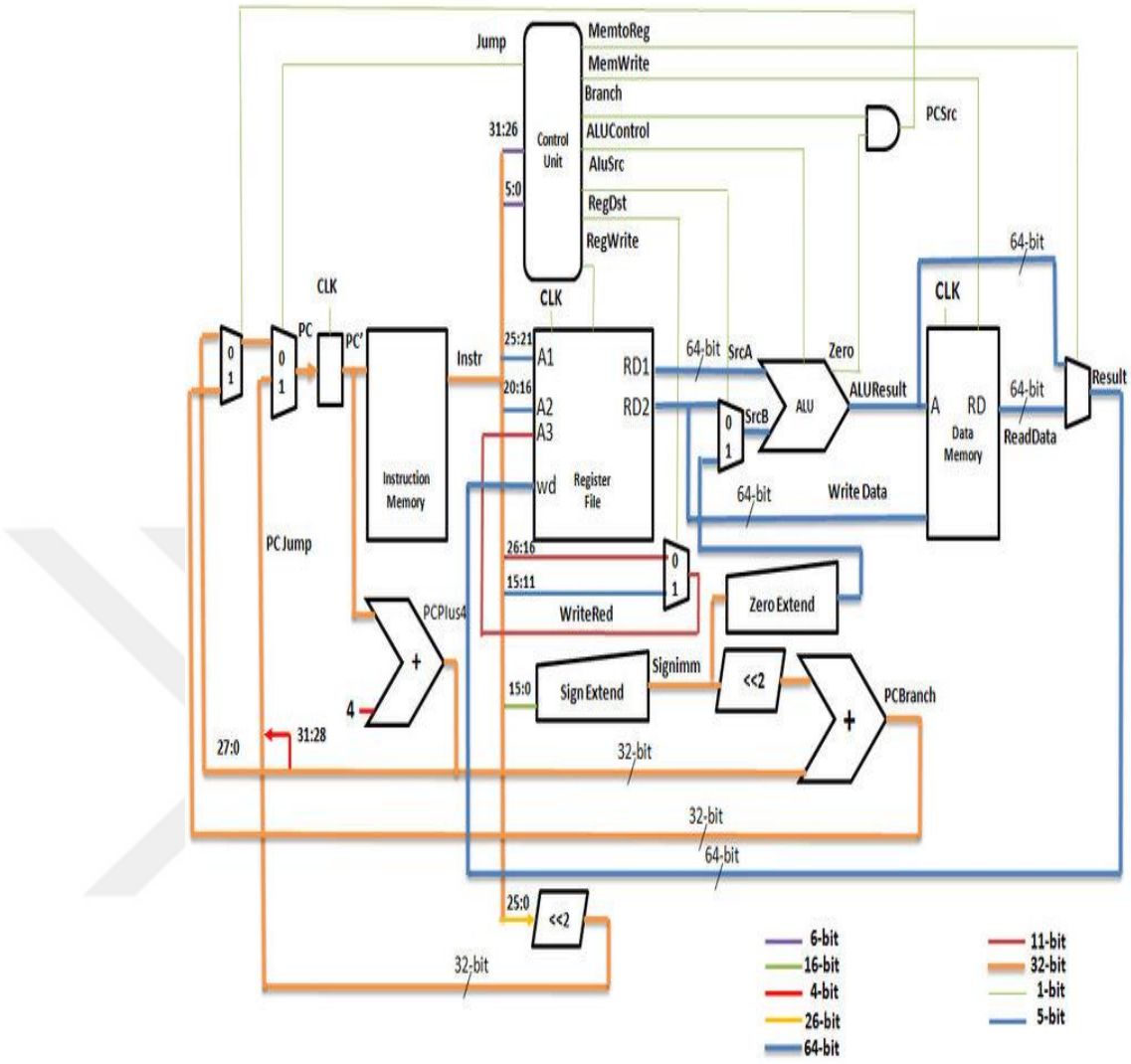
Şekil 3.2 de görüldüğü üzere RISC mimarisinde makine kodları direkt olarak çalıştırılarak işleme sokulurlar. CISC mimarisinde ki programın parçalara ayırıp işlenmesi aşamaları derleyiciler üzerinden haricen yapılır. Genel olarak bu mimariyi baz alarak ortaya çıkan bazı mikro işlemciler şöyledir; ARM, AVR, PIC, vs.

3.2. MIPS Mimarisi

MIPS mimarisi MIPS firması tarafından değiştirilmiş RISC tabanlı bir mimaridir. 32 bit veya 64 bit veri yoluna ve komut mimarisine sahiptir. RISC mimarisinde ki sınırlı yazmaç boyutları ve işlem kabiliyetleri anlamında eksiklikleri vardır. Bu eksiklikler genellikle kullanıcının hesaplamak istediği veri uzunluğu ve hesaplama çeşitliliklerinden kaynaklanır.

ARC (Advance RISC Computing) standartlarına sahip olan MIPS mimarisi bir çok grafik uygulamalarında kullanılmasının yanı sıra ortaya çıktığı günden bu yana oldukça gelişmiş sistem çözümlerine dahil olmuştur. RISC mimarisinden farklı olarak MIPS, RISC mimarisinin yazılımlara olana bağımlığını bir ölçüde azaltarak CISC mimarisinin donanımsal hızını kullanmıştır. RISC mimarisinde tasarlanan mikro işlemciler (PIC, AVR) genellikle işlem birimlerinde toplama ve çıkarma işlemleri yapabilme kabiliyetine sahip olup yazmaçları 8 bit uzunluğundadır. Çarpma ve bölme işlemleri yazılımsal olarak toplama ve çıkarma işlemlerinin tekrarlanarak elde edilmektedir. MIPS mimarisinde ki mikro işlemcilerin donanımsal olarak çarpıcı ve bölücüleri vardır. Bu bağlamda hesaplama yetenekleri oldukça yüksektir.

Şekil 3.3'te görüldüğü üzere MIPS mimarisindeki bir mikro işlemci ALU(Aritmetik lojik ünite), Register File, Data Memory, Sign Extend, Control Unit, Program Counter, Instruction Memory gibi birimlerden oluşmaktadır.

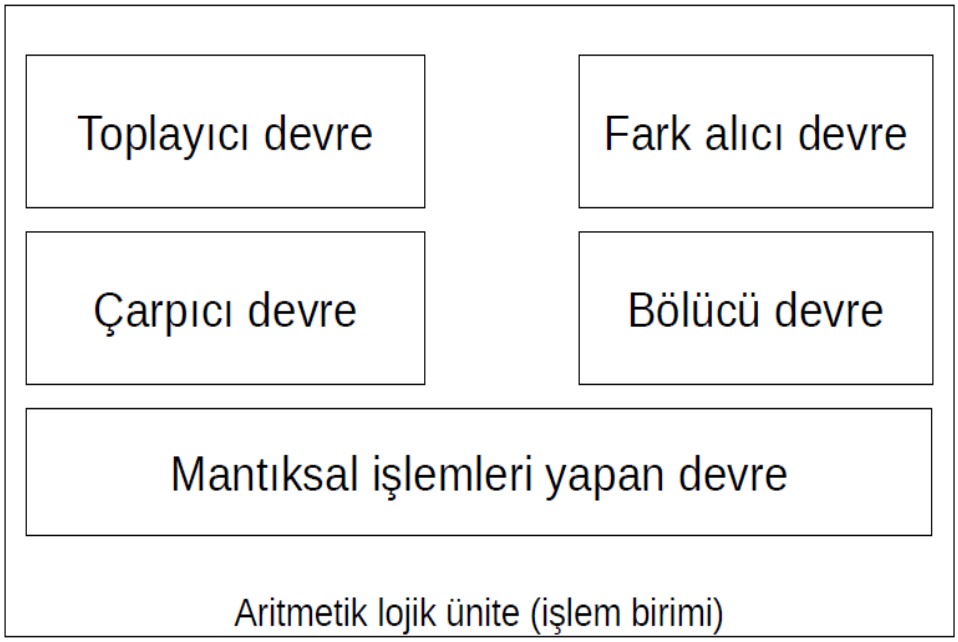


Şekil 3.3 MIPS mimarisi blok diyagramı (Omran ve Jumma ,2017)

3.2.1. **ALU (Aritmetik Lojik Ünite)**

ALU, temelde bir işlem birimidir. İki adet 64/32 bit girişi, bir adet 64/32 çıkışı ve bir adet bayrak çıkışı vardır. ALU içerisinde aritmetik dört işlem (toplama çıkarma çarpma ve bölme) ve lojik (And, or, xor vs.) gibi işlemler dahili olarak yapılır.

ALU işlem yapabilmek için 2 tane işlem yapılacak veri almalıdır. Bu verilerden biri registerlardan (yazmaç) alırken diğer işlem verisi kodlayıcı tarafından gönderilen veriden alınır.

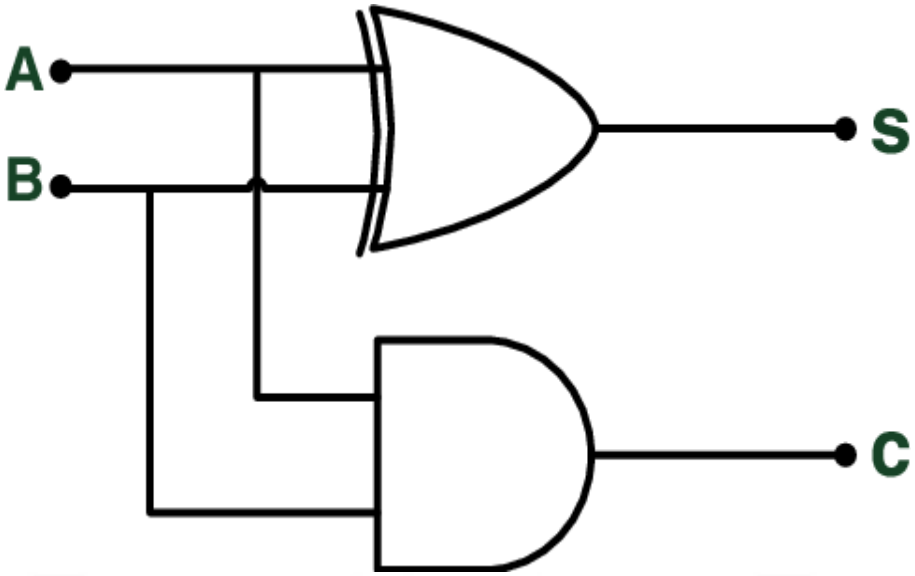


Şekil 3.4 ALU blok gösterimi

Şekil 3.4 ‘ te görüldüğü üzere ALU içerisinde toplayıcı, fark alıcı, çarpıcı, bölücü ve mantıksal işlem birimleri(devreleri) bulunur.

3.2.1.1. Toplayıcı Devre

Girişine uygulanan belirli bit genişliğinde ki verileri binary olarak toplayan tümleşik lojik devredir. Temelinde bir xor gate ve and gate bulunur ve sadece bu kapılarla oluşturulan toplayıcı devrelere half adder (yarım toplayıcı) devre denir.



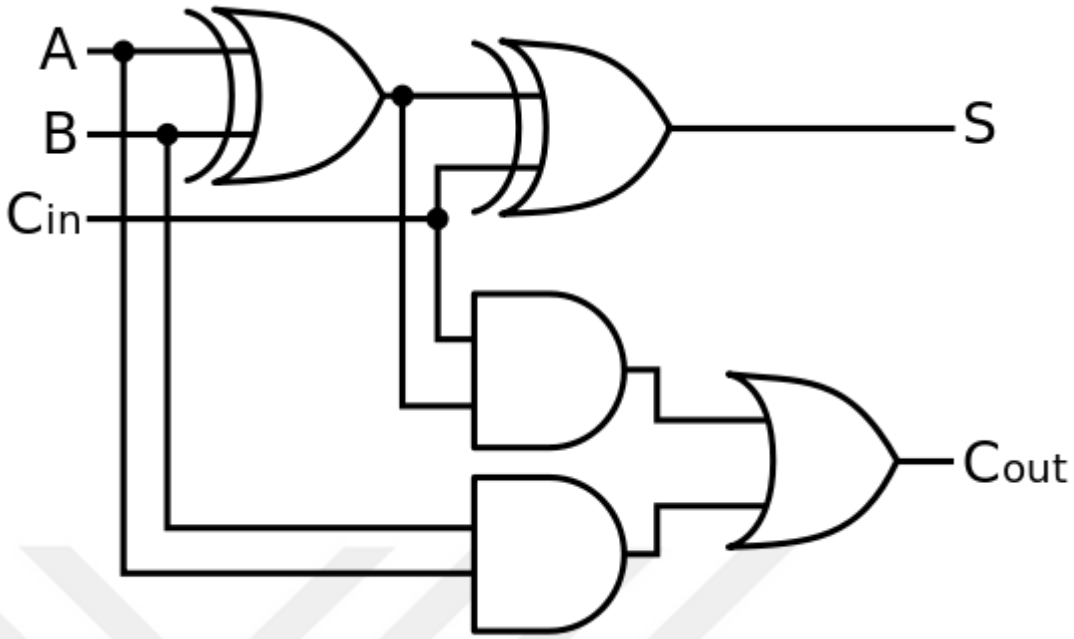
Şekil 3.5 Half Adder lojik diyagramı

Şekil 3.5'te görüldüğü gibi bir yarım toplayıcı da iki adet bir bitlik veri girişi (A,B) bir adet sonuç (S) ve elde biti olarak (C) vardır. Doğruluk tablosu, Çizelge 3.1 de gösterilmiştir.

Çizelge 3.1 Half Adder doğruluk tablosu

A (Giriş)	B(Giriş)	C(Çıkış)	S(Çıkış)
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	0

Yarım toplayıcıların elde biti girişlerinin olmaması sebebiyle eldeli bir sayı ile toplama yapmaları olanaksızdır bu sebeple iki adet yarım toplayıcı seri bağlanarak full adder (tam toplayıcı) oluştururlar.



Şekil 3.6 Full adder lojik gösterimi (URL-3)

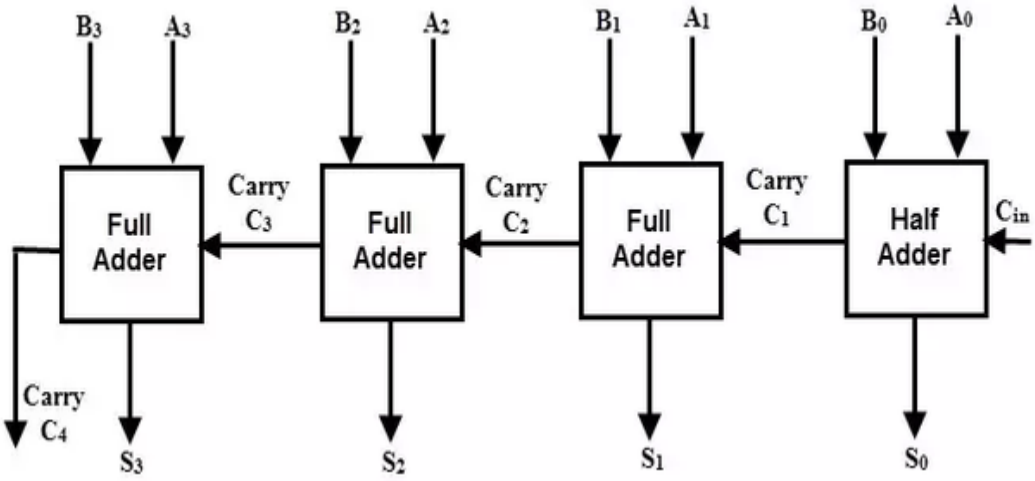
Şekil 3.6'da görüldüğü üzere bir full adder 2 adet xor kapısı, 2 adet and kapısı ve bir or kapısıyla oluşturulur. A ve B birer bitlik giriş verisi, Cin ise elde giriş verisini ifade eder. S toplamın sonucu ve Cout elde sonucunu ifade etmektedir. Çizelge 3.2 bir full adderın giriş ve çıkışları arasındaki lojik bağıntıyı göstermektedir.

Çizelge 3.2 Full Adder doğruluk tablosu

A(Giriş)	B(Giriş)	Cin (Giriş)	S(Çıkış)	Cout(Çıkış)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Birer bit olarak gösterilen full adder ve half adder, ALU içerisinde birden fazla bitlerle işlem yapmaktadırlar. Şekil 3.7'de dört bitlik bir toplayıcının blok diyagramı

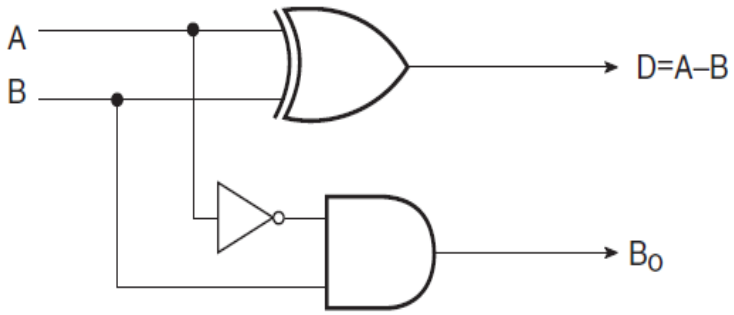
ve organizasyonu gösterilmektedir. Dört veya daha fazla toplama devreleri aynı organizasyon içerisinde devam etmektedir.



Şekil 3.7 Dört bitlik toplayıcı devre blok gösterimi (URL-3)

3.2.1.2. Fark Alıcı Devre

Fark alıcı devre(substractor), toplayıcı devrenin and kapısının değillenmiş halini ifade eder.Yani, temelde bir toplayıcı devrenin ters çalıştırılması prensibine dayanır. Şekil 3.8 bir half susbstractorun lojik devresini göstermektedir. Burada ki A ve B giriş bitleri D, işlem sonucu biti ve B0 olarak ifade edilen çıkış ise borç bitidir. Çizelge 3.3 half substractor doğruluk tablosunu ve bitler arasında ki ilişkiyi çıkarma işlemine göre göstermektedir.

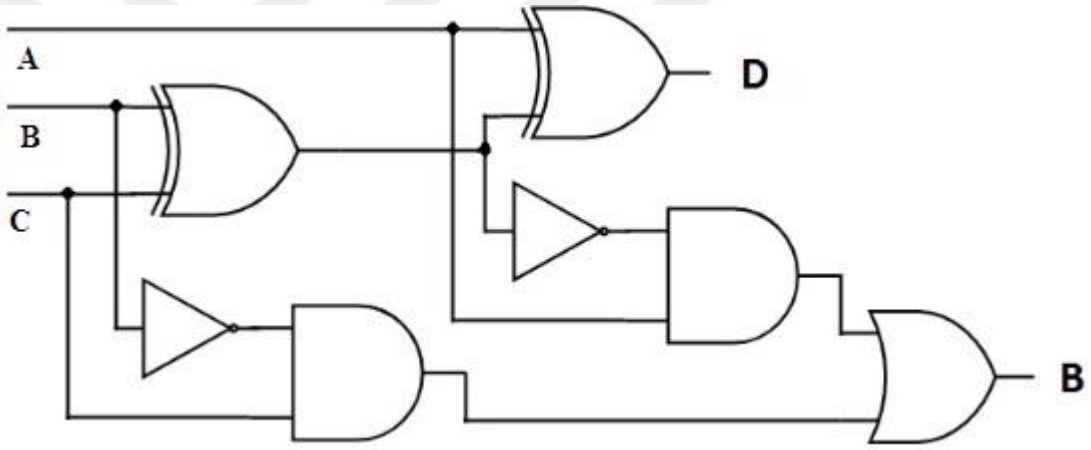


Şekil 3.8 Half Substractor lojik gösterimi (URL-3)

Çizelge 3.3 Half Substractor doğruluk tablosu

A(Giriş)	B(Giriş)	D(Çıkış)	B0(Çıkış)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Tam toplayıcılara parallel olarak, tam fark alıcılar da (Full substractor) vardır. Yine aynı şekilde tam toplayıcıların girişleri terslenerek elde edilirler. Lojik gösterimi Şekil 3.9' da gösterilmiştir. Çizelge 3.4 ise tam fark alıcı devrenin doğruluk tablosunu ifade etmektedir.

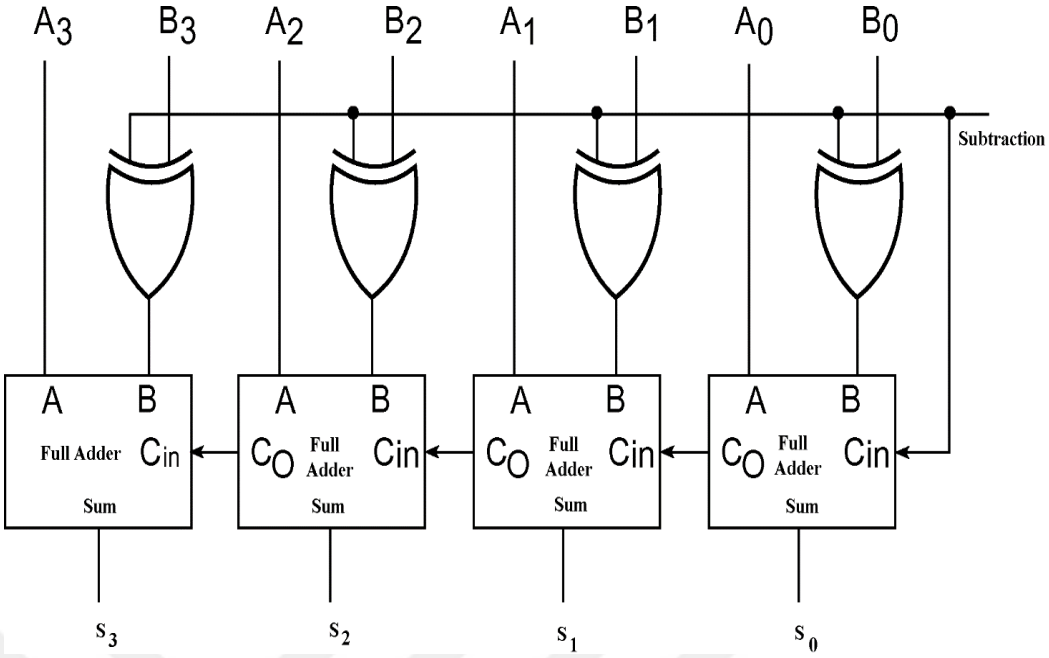


Şekil 3.9 Full Substractor lojik gösterimi(URL-3)

Çizelge 3.4 Full substractor doğruluk tablosu

A	B (	C	D	B)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

ALU içerisinde fark alma işlem için ekstra bir birim bulunmamaktadır. Çünkü toplayıcı devreler gibi çalışan fark alıcı devreler, XOR kapılarının full adder devrelerine eklenerek elde edilir. Yani toplayıcı devreler hem toplama işlemi hem de çıkarma işlemi yapabilmektedirler. Bu durum mikro üretimde ayrıca bir yükün azalmasını sağlamakla birlikte esnek bir yapı sunmaktadır.



Şekil 3.10 4-Bit Adder/Substractor blok gösterimi(URL-5)

Şekil 3.10 hem toplayıcı hem fark alıcı bir devrenin lojik ve blok gösterimini ifade eder. Burada ki Susbraction biti fark alma kontrolcü bitidir bu birin aktif olduğu durumlarda fark alma işlemi aktif olur ve Full Adder'lar, Full Susbstractor gibi çalışmaya başlarlar.

3.2.2. **Çarpıcı Devre**

Çarpıcı devre ALU'nun en çok işlem gücü gerektiren kısımlarındandır. Çarpma işlemi için iki adet giriş verisine ihtiyaç duyar. Çıkış verisinin bit uzunluğu, giriş verisinin bit uzunluğunun iki katı olmak zorundadır. Bunun sebebi iki sayının dijital çarpımı sonucunda elde edilen sonuç giriş verilerinin alabileceği maksimum değerden fazla olabilmesidir.

Çarpma işlemi dijital olarak, normal çarpma işlemine benzer olarak yap yapılır. Yani öncelikle her bir basamak kendi arasında çarpılır ve elde edilen sonuçlar toplanır. Bu bağlamda çarpıcı devre de toplayıcılar kullanılmaktadır.

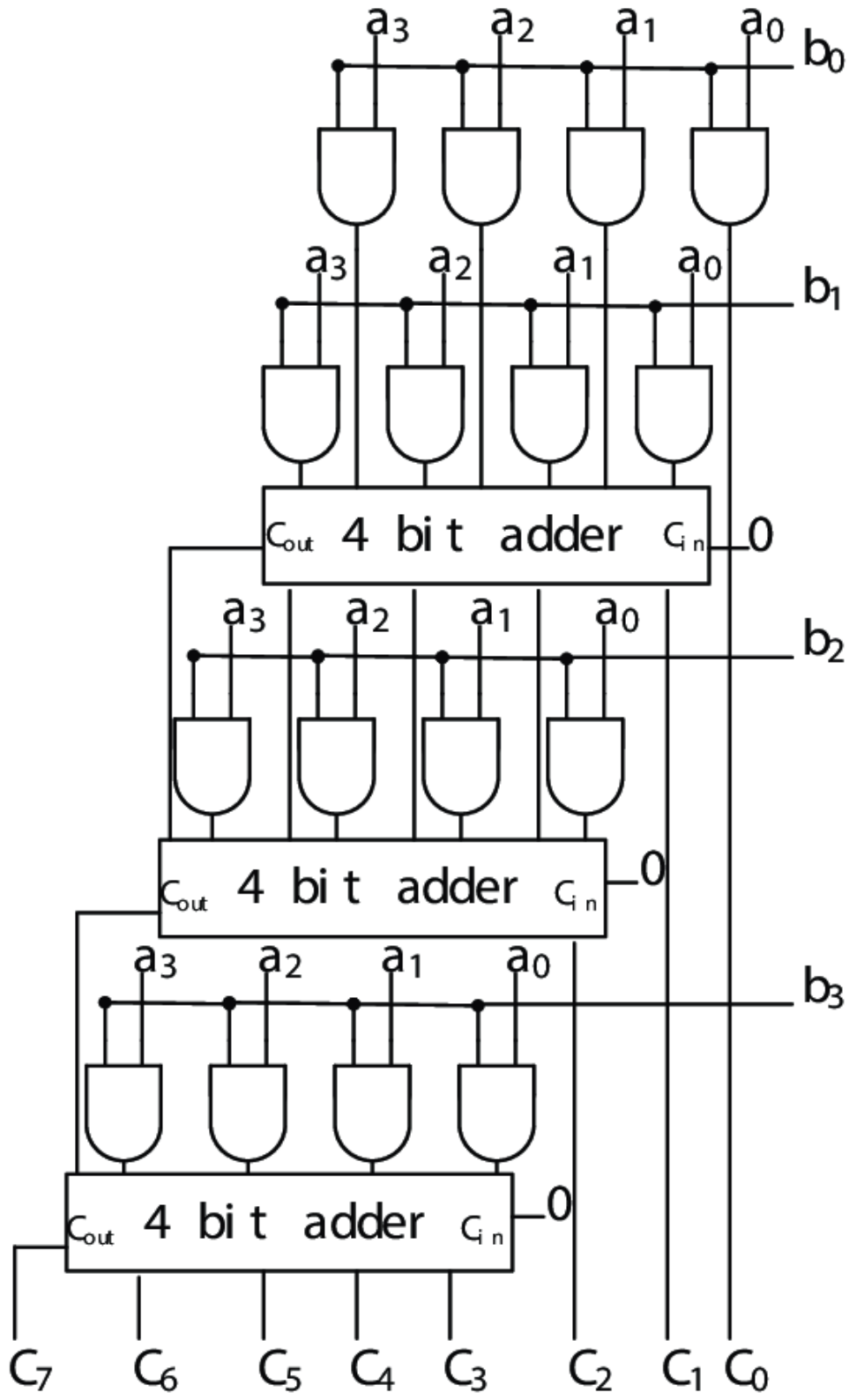
					a_3	a_2	a_1	a_0
$\times$					b_3	b_2	b_1	b_0
					a_3b_0	a_2b_0	a_1b_0	a_0b_0
				a_3b_1	a_2b_1	a_1b_1	a_0b_1	
			a_3b_2	a_2b_2	a_1b_2	a_0b_2		
$+$		a_3b_3	a_2b_3	a_1b_3	a_0b_3			
	y_7	y_6	y_5	y_4	y_3	y_2	y_1	y_0

Şekil 3.11 Binary çarpma işlemi basamakları(URL-7)

Şekil 3.11 de dört bitlik iki binary sayının çarpımını ifade eden işlem basamakları gözükmektedir. Her bitin ikili kombinasyonları birer basamak kaydırılarak toplanmaktadır. Yani b0 bitinin ikili kombinasyonları ilk satırı, b1 bitinin a bitleri üzerindeki ikili kombinasyonları bir basamak kaydırılmış ikinci satırı oluşturmaktadır. Bu prensiple devam eden çarpma işleminin sonucunda bütün satırlar full adderlar tarafından toplanarak y0:y7 8 bitlik çıkış verisi elde edilmektedir. Bu veri çarpımın sonucudur.

Şekil 3.12 bir çarpıcı devrenin genel olarak yapısını göstermektedir. A0:A3 ve B0:B3 dört bitlik giriş verileridir. Adderlar arasında kullanılan and kapıları çarpma işlemini sağlarken sonuç olarak C0:C7 çıktılarını vermektedir. Bu durumda çarpıcı devrelerin hızları ve güç tüketimleri toplayıcı devrelerdeki gibi olmaktadır. Bu sebeple temel olarak toplayıcı devrelerin optimize edilmesi oldukça önem kazanmakla birlikte tüm tasarımı etkileyen bir öneme sahiptir.

Ayrıca çarpıcı devrelerin verimleri oldukça önem arz etmektedir. Çünkü işlem yükünün en çok olduğu ALU yapısıdır. Bu bağlamda verimlilik hız ve hassasiyet tasarımlarında esas alınan parametrelerdir.

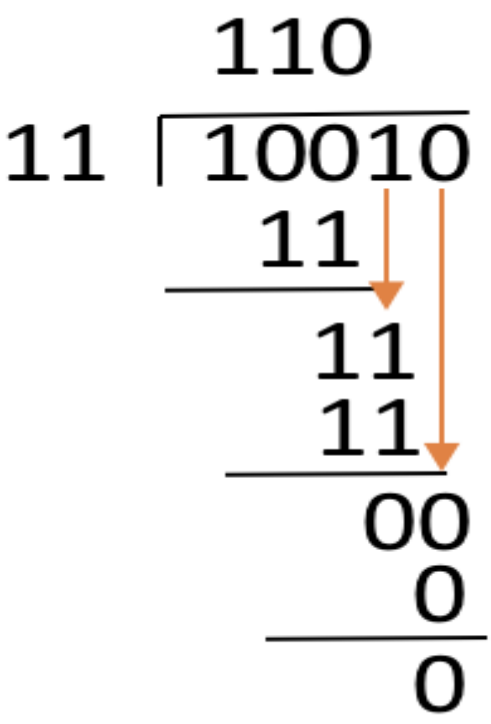


Şekil 3.12 Çarpıcı devre lojik ve blok gösterimi(URL-4)

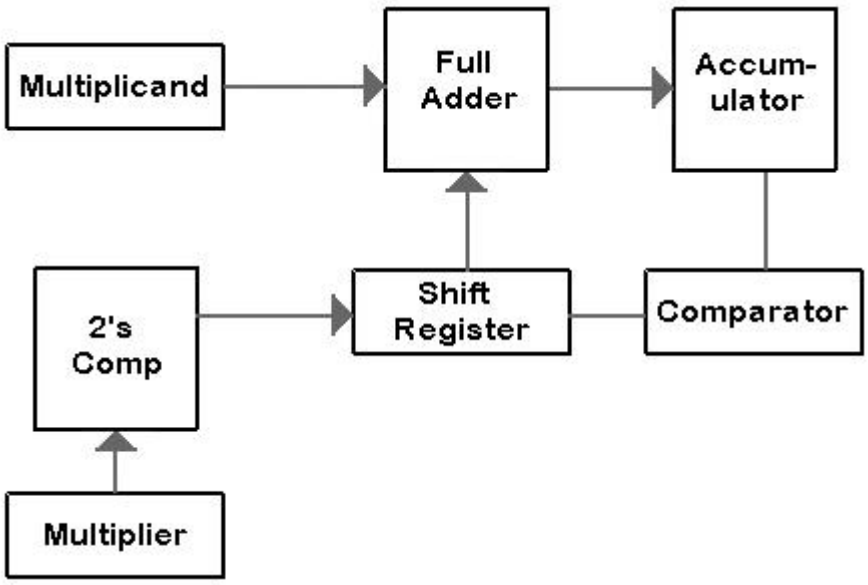
3.2.3. **Bölücü Devre**

Binary bölme işlemi yine çarpmada olduğu gibi, decimal bölme işlemine benzerdir. Ancak dijital olarak devre tasarımı ALU içerisindeki en karmaşık lojik ifadelerden biridir. Temel de bölenin bölünen ile farkının alınmasıyla elde edilir. Yani bir diğer ifadeyle substractor ile hesaplanır. Bura da alınan her fark ile bir artan bir sayıcı vardır. Sayıcının kaç kez değiştiği bölümü verirken son döngüde ki substractorun çıktısı kalan ifadesidir. Ayrıca her bir fark işleminin sonucunda bölüm ile bölünenden kalan ifade kontrol edilir. Kalanın bölümden büyük olduğu durumlarda sayaç çalışmaya devam etmektedir.

Şekil 3.13 binary iki sayının bölümünü gösteren bir diyagramdır. 11 bölen 110010 ise bölünen, 110 bölüm ve son ifade olan 0 kalandır.



Şekil 3.13 Binary bölme işlemi



Şekil 3.14 Binary bölücü devre blok gösterimi

Şekil 3.14 basit bir bölücü devrenin temellerini göstermektedir. Buradaki shift registerlar bölmede aşağı alınan sayıları sağlar ve karşılaştırıcılar sayesinde her seferinde bölünen ile kalan arasındaki ilişkiyi inceler.

3.2.4. **Instruction Memory**

Kodların saklandığı hafıza birimidir. Kullanıcı tarafından yazılan kodların derlenip makine kodu haline getirilmesinin ardından bu hafıza içerisinde saklanır. Temel olarak sadece okunabilir hafıza tiplerindendir. Verilerin yazılabilmesi için özel bir protokol gereklidir. İçerisinde sakladığı verileri adresleyerek saklamaktadır. Yani yazılan her bir satır kodun karşılığında, o koda ait bir addres belirlenir.

3.2.5. **Program Counter**

Yazılan kodların sırasıyla veya kod içerisinde ki haliyle işeleyebilmesi için IM'ye adres veren sayıcı yapısıdır. Yazılan kodun türüne göre farklı adreslere geçiş yapabilmesi için ekstra lojik devreler vardır. Bu devreler genellikle PC'nin akışını değiştirirler. Temel de özelleştirilebilen bir sayaç ve karşılaştırıcı lojiklerden oluşur.

Program counter aynı zamanda stack pointerları kontrol eden birimdir. Bu sayede programın dallanması veya jump yapması durumunda önceki durumu her zaman korumakadırlar.

3.2.6. **Control Unit**

Mikro işlemci içerisindeki bütün birimlerin organize çalışabilmesi için gerekli olan kontrol sinyallerini sentezleyen ve organizasyonu kontrol eden birimdir. Giriş olarak IM'den aldığı veriyi değerlendirir. Bu veri içerisinde yazılmış olan kodun karakteristiği vardır. Bu sayede gelen kodun hangi birimlerce çalıştırılması gerektiğine karar verir.

Temelde yapısında tri state buffer (Üç konumlu kuvvetlendirici), karşılaştırıcı ve registerlar vardır. Bu sayede sentezleyeceği sinyallerin koşul durumunu oluşturur.

3.2.7. **Register File**

ALU için gerekli verilerin depo edildiği adreslenmiş registerlardır. Bir sonraki yazmaya kadar üzerlerindeki verileri saklarlar. 32 bit uzunluğunda 32 tane registerdan oluşurlar. İki adet adres girişi iki adet veri çıkışı vardır. Bu çıkışlar ALU için işlem verilerin oluştururlar. Control unitten gönderilen sinyaller ile iki çıkıştan biri ALU için işlem verisi olarak verilir.

3.2.8. **Data Memory (RAM)**

Mikro işlemcinin depolama birimi olarak çalışan bir hafıza yapısıdır. Veri yazma, adres girişi ve okuma çıkışı bulunur. RAM, adresleri ALU tarafından hesaplanır. Veri yazma girişleri ise Register File tarafından sağlanır. Bir diğer ifade ile register file üzerinde saklanan verilerin değiştirilme durumu olduğunda RAM de saklanır.

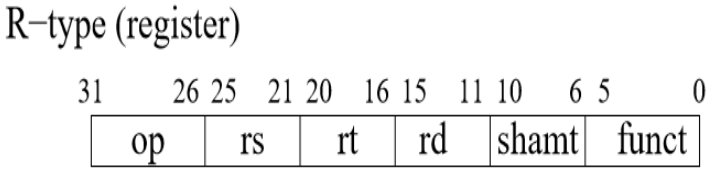
Register file'a benzeyen RAM'lerin okuma yazma hızları registerlara göre düşük olduğundan genellikle sadece depolanıp daha sonra yine kullanılmak istenen verileri ihtiva ederler. Örneğin; bir kod içerisinde tanımlanan değişken bir değer RAM'lerde saklanır. Depolama alanları oldukça büyüktür.

3.2.9. **Sign/Zero Extender**

IM'den çıkan kodlar 32 bit genişliğindedir. Bu 32 bitlik verinin 16 bitlik kısmı kodun karakteristiğini belirten kısımlardır. Kalan 16 bitlik veri ise programlayıcı tarafından gönderilen ALU işlem verisidir. Ancak ALU, 32 bit verilerle işlem yapabilmekte ve 16 bitlik bir veriyi işlemesi durumunda sonucun yanlış olduğu bir senaryo ortaya çıkmaktadır. Bu problemin önüne geçebilmek adına işaret veya sıfır uzatma birimleri vardır. 16 bitlik verinin işaret bitini veya sıfır bitini uzatarak 32 bitlik bir veriye dönüştürerek hatalı hesaplamanın önüne geçmektedir.

3.3. MIPS ISA(Instruction Set Architecture)

MIPS mimarisine sahip mikro işlemcilerin üç tip komut setleri vardır. Bunlar R-Type , I Type ve J Type olarak isimlendirilen komut setinin söz dizim kurallarını belirten ve mikroişlemci içerisindeki kontrol birimince anlaşılmasını sağlayan yapılardır.



Şekil 3.15 R type ISA

R-Type komut setinde 25 ile 21 bitleri arasında bulunan rs (register select) işlem yapılacak genel amaçlı yazmaçlardan ilkinin adresini barındıran bitlerdir.(Şekil 3.15)

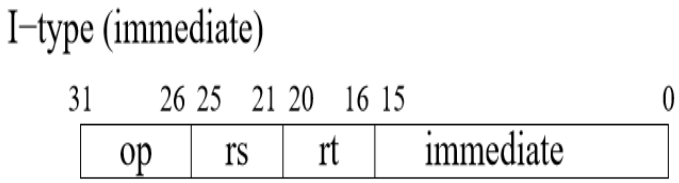
20 ile 16 bitleri arasında Rt (register target) işlem yapılacak genel amaçlı yazmaçlardan ikincisinin adresini barındırır. (Şekil 3.15)

15 ile 11 bitleri arasında Rd (Register destination) op'ta belirtilen işlem yapıldıktan sonra yazılmak istenen yazmacın adresini barındırır. (Şekil 3.15)

10 ile 6 bitleri arasındaki shamt(shift amount) desimal olarak bit kaydırma yapılacak ise ne kadar kaydırma yapılacağını barındıran bitlerdir. (Şekil 3.15)

Son olarak funct(Function) gelen komut setinde ki işlemin tam olarak ne olduğunu (çarpma bölme çıkarma toplama) belirtir. Sonuç olarak blok gösterimi şu şekildedir. (Şekil 3.15)

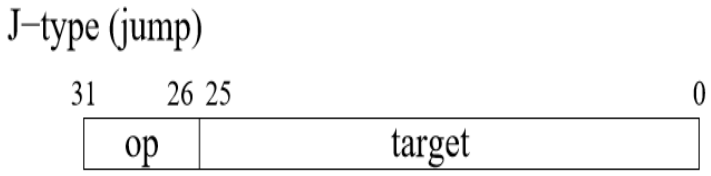
$$R[rd] \leftarrow R[rs] \text{ op } R[rt]$$



Şekil 3.16 I Type ISA

I Type komut setinde ise; 25 ile 21 bitleri arasında rs ve 25 ile 21 bitleri arasında rt bulunmaktadır.15 ile 0 bitleri arasındaki immediate bitleri ise rs adresindeki yazmacın içeriği ile belirtilen işlemi yapıp rt adresindeki yazmaca yazmaktadır. Blok gösterimi şu şekildedir; (Şekil 3.16)

$$R[rt] \leftarrow R[rs] \text{ op imm}$$



Şekil 3.17 J Tpye ISA

J-Type komut setinde ise 25 ile 0 bitleri arasında gidilecek adresin hexadesimal olarak karşılığı vardır. (Şekil 3.17)

Üç tip komut setinde ilk 5 bit op (operation code) bulunmaktadır. Bu mikro işlemciye gelen komutun kimliği niteliğindedir.

3.4. Yöntem

Bu çalışmada tasarımların tamamı Quartus üzerinde yapılmıştır. Quartus, HDL(Hardware Descrpition Language) derleyicisidir. Yani aygıt tasarımlarının derlenmesini söz dizimleri üzerinden yapar. Yapılan bu derleme sonucunda yapılan tasarım hakkında kritik verileri rapor etmektedir.

Ayrıca lojik devre RTL(Register transfer level) olarak görüntülenebilir ve eksiklikler görselleştirilerek gösterilebilmektedir. Bunun yanı sıra FPGA üzerindeki lojik hücreler ile gösterilen daha alt seviye RTL gösterimlerde mevcuttur.

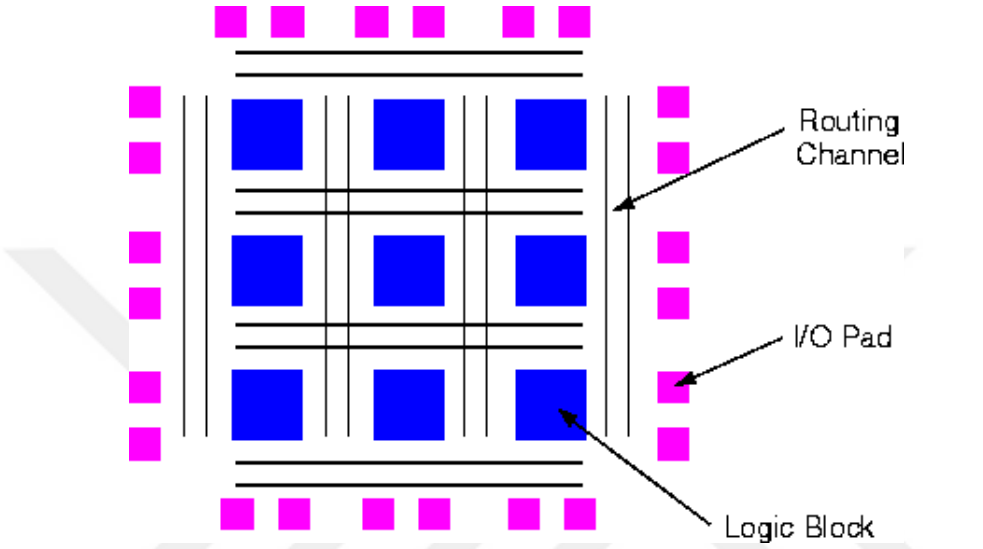
Quartus birden fazla HDL dilini desteklemektedir. Bunlardan bazıları; VHDL, Verilog ,System Verilog gibi donanım tanımlama dilleridir. Çalışmada Verilog kullanılmış olup bütün tasarımlar bu dil üzerinden sağlanmıştır.

3.4.1. **FPGA**

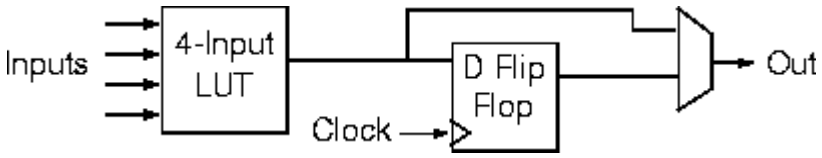
FPGA, tümleşik devreler için bir prototipleme çipidir. İçerisinde bulunan FPGA tipine göre farklılık gösteren lojik hücreler sayesinde tümleşik sistem tasarımları yapılabilir ve test edilebilir. Aynı zamanda mikro işlemci veya mikro denetleyici olarak da kullanılabilen, FPGA esneklik anlamında mikro işlemcilere göre çok daha avantajlıdır. Buradaki esas nokta tasarımcının ihtiyaç duyduğu sayısal lojik devreleri gerçekleştirebilmesidir. Örneğin bir mikro denetleyicinin çalışma frekansı, tasarımdaki lojik devreler sabittir ve değiştirilemez. Ancak FPGA üzerinde tasarlanan herhangi bir tümleşik devrenin herhangi bir kısmı değiştirilebilir silinip yeniden tasarlanabilir.

FPGA, genel olarak PCB üzerinde geliştirici kitler olarak üretilir. Aynı zamanda işlem gücü gerektiren görüntü işleme, derin öğrenme, yapay sinir ağları vs. uygulamalar için idealdirler. Ayrıca bütün DSP(Digital Signal Processing) problemlerine çözüm olacak DSP blokları barındırdıklarından dolayı bu uygulamalarda çoğun-

lukla tercih edilmektedirler. Yapılarında mantık blokları giriş çıkış blokları ve arabağlantı birimleri bulunur (Şekil 3.18). Mantık blokları temel düzeyde lojik kapılardan oluşur. Giriş çıkış blokları ise FPGA çipinin fiziksel pinlerini ifade etmektedir. Ara bağlantılar ise bloklar arasındaki mikron kalınlığında ki sinyal bağlantılarını ifade etmektedir.



Şekil 3.18 FPGA çipi blok gösterimi (Toronto University)

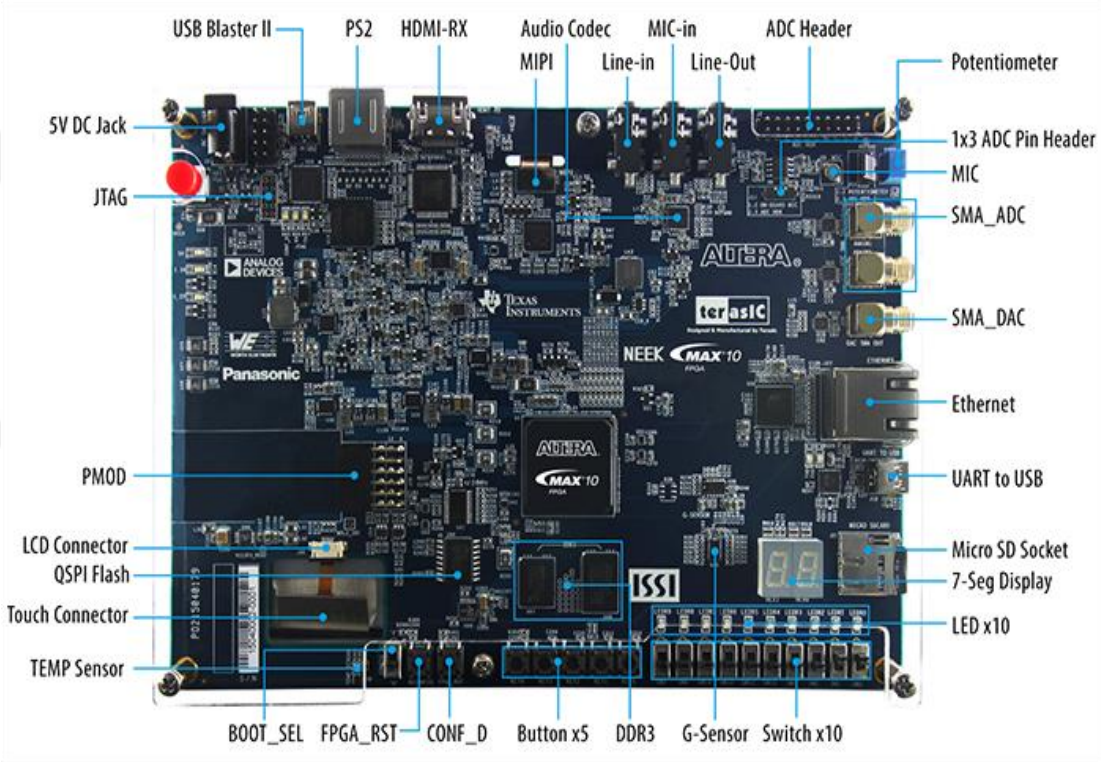


Şekil 3.19 Mantık bloğu iç yapısı blok gösterimi(Toronto University)

Bir mantık bloğu içerisinde LUT, (Look Up Table) mux, full adder ve D tipi flip flop bulunur. Burada ki LUT içerisinde bulunan özelleştirilmiş lojik kapılar sayesinde tasarımcı tarafından gönderilen a,b,c,d değerlerine göre hangi lojik kapıların çalışacağı belirlenmektedir. Full adder ise tasarımlarımlarda kullanılan hazır toplayıcı devrelerdir. Tasarımcı toplayıcı devre tasarımına ihitiyaç duymadığı zamanlarda direkt olarak bu adder üzerinden tasarımını gerçekleştirebilmektedir. (Şekil 3.19)

Tasarımcının tercih ettiği tasarım yöntemine göre mux seçim yaparak veriyi direkt olarak D FF üzerinden ya da LUT üzerinden ilgili bağlantıyı yapar. Bağlantılar için kullanılan fiziksel yollar vardır ancak bu yollar dijital olarak kontrol edilip bir birine bağlanıp ayrıştırılabilir.

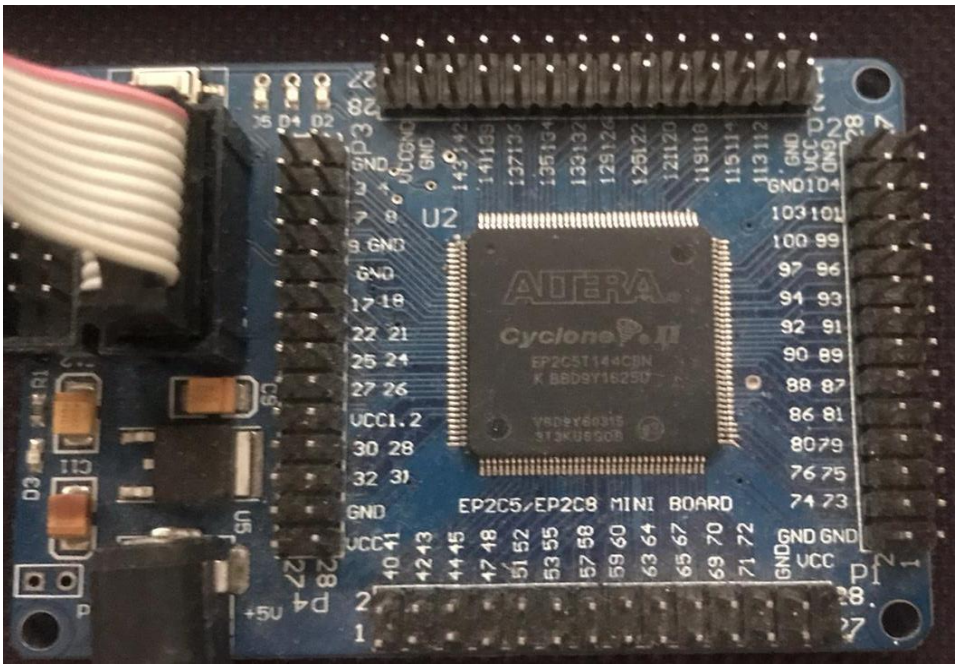
FPGA geliştirme kartları genellikle bir çok çevresel birime sahiptir. Görüntüleme birimleri vardır. Dahili kameraya dokunmatik ekranlara sahip olan çeşit ve modelleri de vardır. Kullanım amacı ve geliştirme alanına göre seçimler yapılmalır.



Şekil 3.20 ALTERA Max10 Neek FPGA kartı (URL-9)

Örneğin Şekil 3.20'de gösterilen FPGA kartı bir çok çevresel birime sahiptir. Sıcaklık sensörü dokunmatik sensor ADC (Analog dijital dönüştürücü) SMA ADC, SMA DAC gibi ses sinyali işleyebilmek için bir çok portlara sahiptir. Bu kartın DSP işlerinde kullanılmaya ayrıca giriş seviyesinde lojik tasarım yapmak amacıyla tasarlanmıştır. Ayrıca içerisinde hali hazırda kodlanmış bir ARM Cortex mikro işlemci barındırır. Yani SOC (System On Chip) tipi bir FPGA'dir. BY sebeplerden dolayı geliştirme için uygun olmayan bir platfordur.

Öte yandan Şekil 3.21 de gösterilen FPGA çevresel birimler olarak sadece giriş çıkış pinleri bulunan bir FPGA'dir. DSP veya her hangi bir sinyal işleme işlemi için uygun değildir. Çünkü sinyal işleme birimleri Max10 Neek' te olduğu gibi çeşitli ve dahili değildir. Giriş çıkış portalarının fazla olması geliştirme için uygun ve ileri seviye bir kart olmasına olanak sağlamaktadır. Pin sayılarının önemi geliştirme gerektiren tasarımlarda büyük avantaj sağlamakla birlikte testlerin yapılmasında kolaylık sağlamaktadır.



Şekil 3.21 EP2C5T144C8N Altera Fpga

3.4.2. **Verilog**

Verilog en temel donanım tanımlama dillerinden biridir. Söz dizimleri registerlar, bus hatları koşul blokları, kenar tetikleme gibi bir çok parametreden oluşur. Basit anlamda and or xor nand nor gibi lojik kapıların kodlar ile tanımlanıp geliştirildiği bir donanım dilidir.

Burada bir çok farklı tanımlama yöntemi vardır örneğin bir and kapısı aşağıdaki gibi ifade edilebilir;

Module and_gate (input X, input Y, output Z)

AND(Z,X,Y)

Endmodule

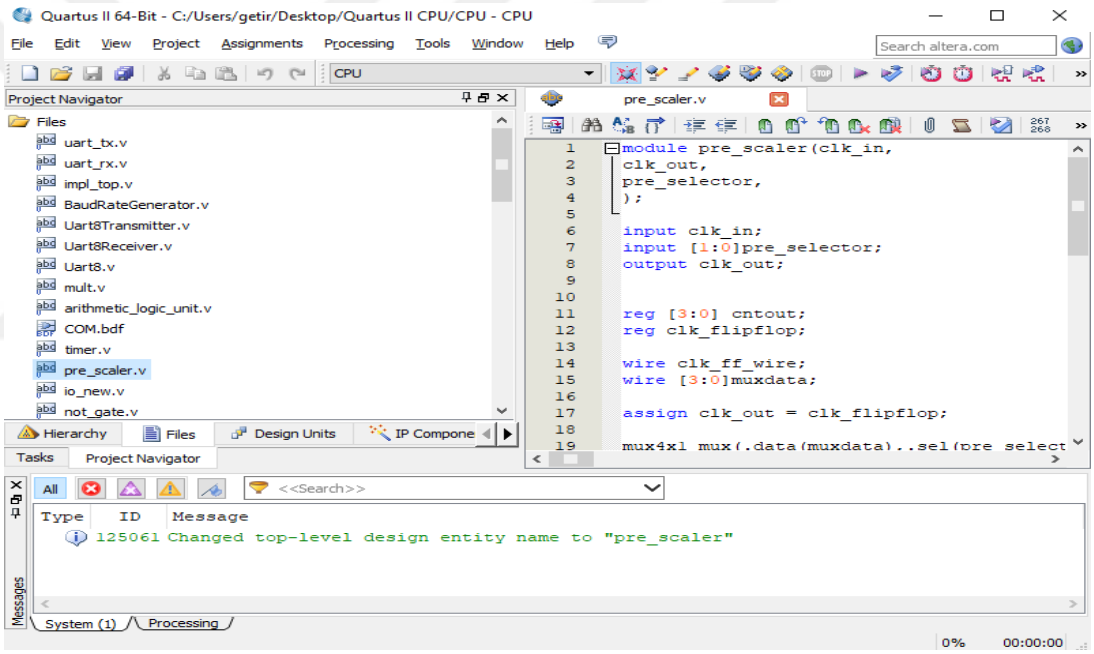
Bu tanıma gate-level tanımlama denir. Bu tip tasarımlar genellikle kısa kod satırlarında ve register tanımlamalarına ihtiyaç duyulmadığı durumlarda kullanılabilir.

Module and_gate (input X, input Y output Z)

Assign Z=X&Y;

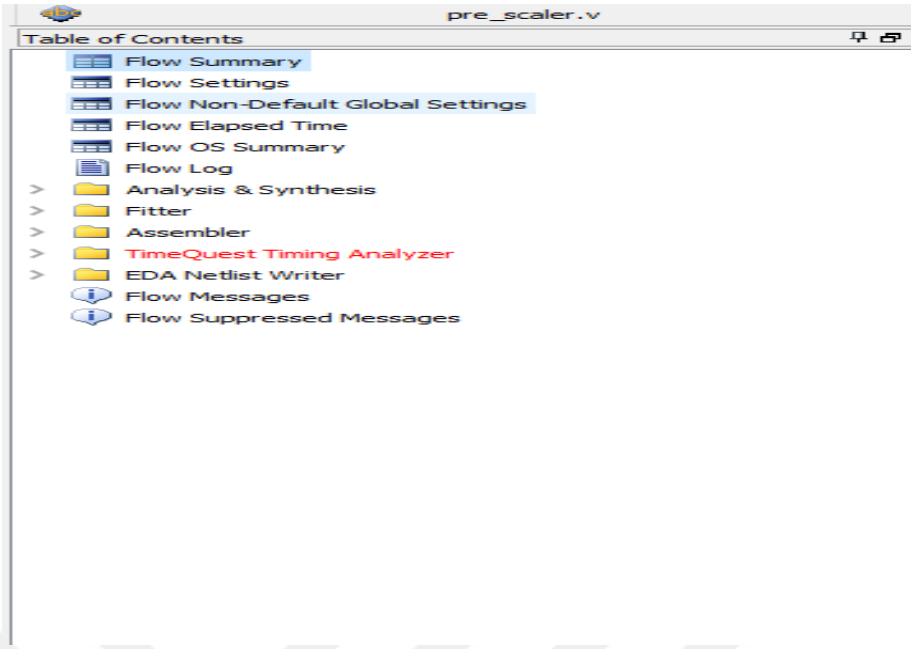
Endmodule

Yukarıdaki tanımlama ise veri akış yöntemi ile tanımlamadır. Bu iki yöntem kullanım uygunluğuna göre tercih edilmektedir.



Şekil 3.22 Quartus yazılımı anasayfa görüntüsü

Şekil 3.22'de Quartus yazılımı ve .v uzantılı Verilog kodları gösterilmetedir. Tasarlanan her bir birim Quartus'ta modüller şeklinde files sekmesinde gösterilmektedir. Aşağıda bulunan terminal ise hata ve bilgi mesajlarının yer aldığı kısımdır.



Şekil 3.23 Derleme raporu sekmesi

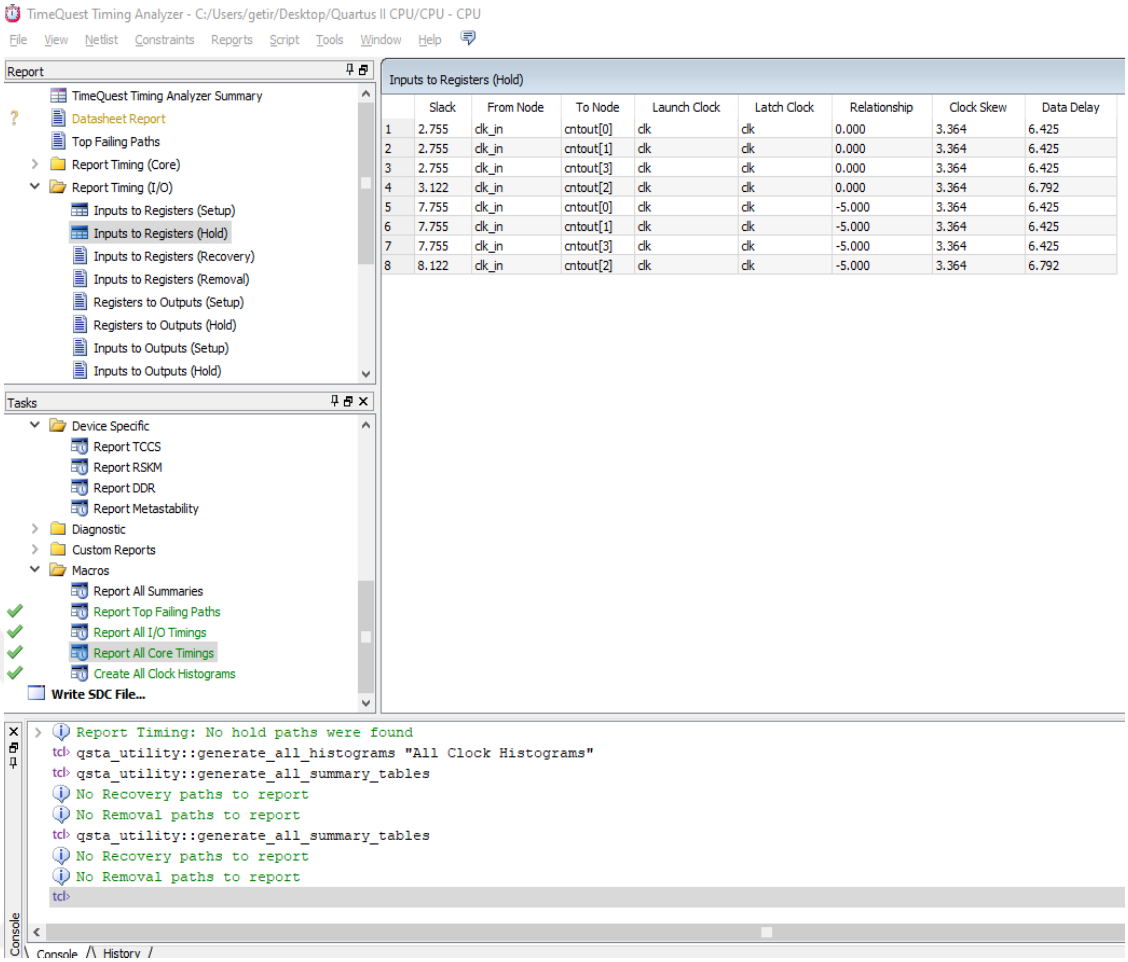
Şekil 3.23 derlenmiş bir Verilog modülünün derlenme sonucundaki raporlamalarının kategorize edildiği kısımdır. Kritik ölçüde önem arz eden TimeQuest Timing Analyzer, Fitter, Flow Summary sekmeleridir. Bu sekmeler tasarım hakkındaki birçok bilgiyi barındırırlar. Örneğin TimeQuest Anlyzer tasarımdaki elektriksel gecikmenin değerini hesaplayan bir araçtır. Yani tasarımın ne kadar hızlı çalıştığı hakkında bilgi verir.

Bu anlamda bu raporların incelenmesi tasarımın iyileştirilmesi anlamında önemlidir. Her bir analiz FPGA içerisinde bulunan programlanabilir lojik hücrelerin fiziki karakteristiğiyle ilgilidir. Yani yapılan tasarım FPGA tipine göre farklı raporlara karşılık gelse de mikro teknoloji ile üretilecek bir çip ile kıyaslanabilir ölçüde farklılaşır.

Flow Summary	
Flow Status	Successful - Tue Oct 27 00:39:36 2020
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	CPU
Top-level Entity Name	pre_scaler
Family	Cyclone II
Device	EP2C70F896C8
Timing Models	Final
Total logic elements	1,131 / 68,416 (2 %)
Total combinational functions	614 / 68,416 (< 1 %)
Dedicated logic registers	957 / 68,416 (1 %)
Total registers	957
Total pins	4 / 622 (< 1 %)
Total virtual pins	0
Total memory bits	79,872 / 1,152,000 (7 %)
Embedded Multiplier 9-bit elements	0 / 300 (0 %)
Total PLLs	0 / 4 (0 %)

Şekil 3.24 Flow Summary raporu

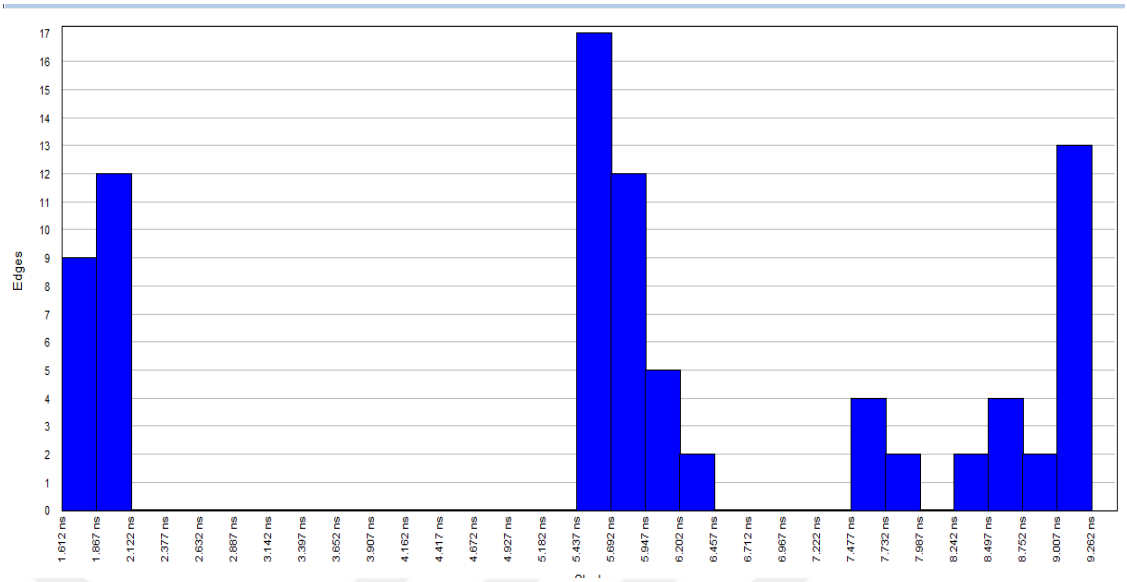
Prescaler modülü için Quartus tarafından sentezlenen Flow summary raporunda Şekil 3.24'de görüldüğü üzere, FPGA üzerinde ne kadar logic cell harcandığı ne kadar register harcandığı ne kadar memory bit kullanıldığı gibi bazı değerler çıktı olarak verilmiştir. Elbette ki bu değerler FPGA üzerinde harcanan değerlere karşılık gelmekte ve bu sayede gerçek bir tasarım hakkında doğru bir yorum yapabilmeye olanak sağlamaktadır.



Şekil 3.25 TimeQuest Analizi

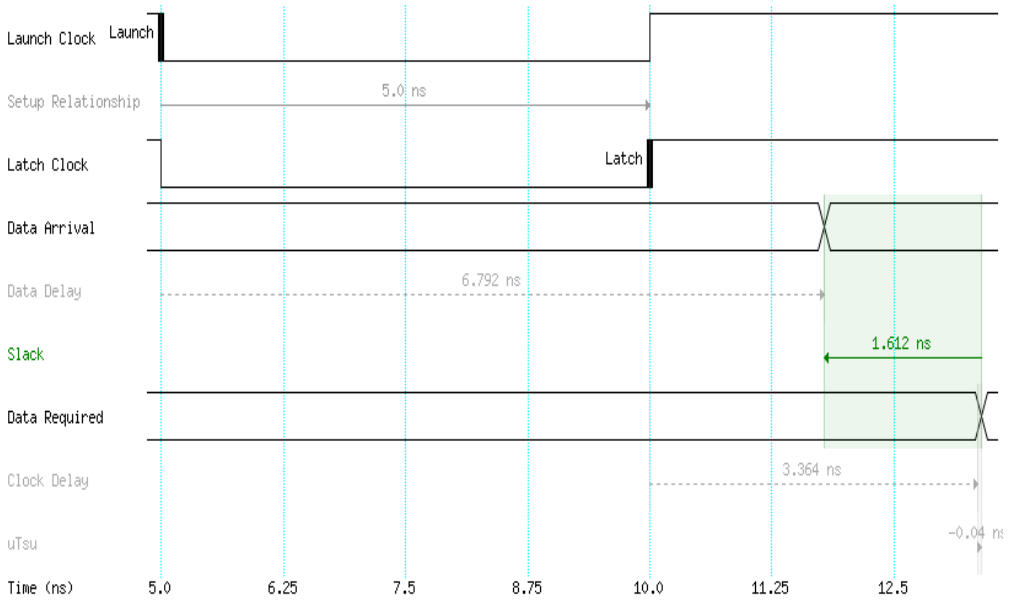
Şekil 3.25 PreScaler için elde edilen bir timing analizini göstermektedir. Sentezlenen bir clok sinyalinin tasarımdaki registerlara ulaşma süreleri clock sinyalinin varış kalitesi gibi bazı değerler gösterilmektedir. Slack sütununda tasarımdaki, giriş verisinin varması gereken teorik zamanın gerçek varış zamanındaki farkın ifadesidir. Teorik zaman hesabı ise clock gecikmesi ve dahili gecikmelerin toplamının maksimum çıkış gecikmesine göre farkını ifade eder. Data delay sütunun da ise verinin varış için harcadığı zamanı ifade etmektedir.

TimeQuest analizi tasarımın matematiksel zaman analizinde gerçek zamanlı olmadan gerçekleştirdiği için zaman hesapları için ekstra ölçümlere gerek duymaktadır. Bunun için modellemeler oluşturulmalı ve oluşturulan modelin gerçekliğe olan yakınlığı tartışılmalıdır.



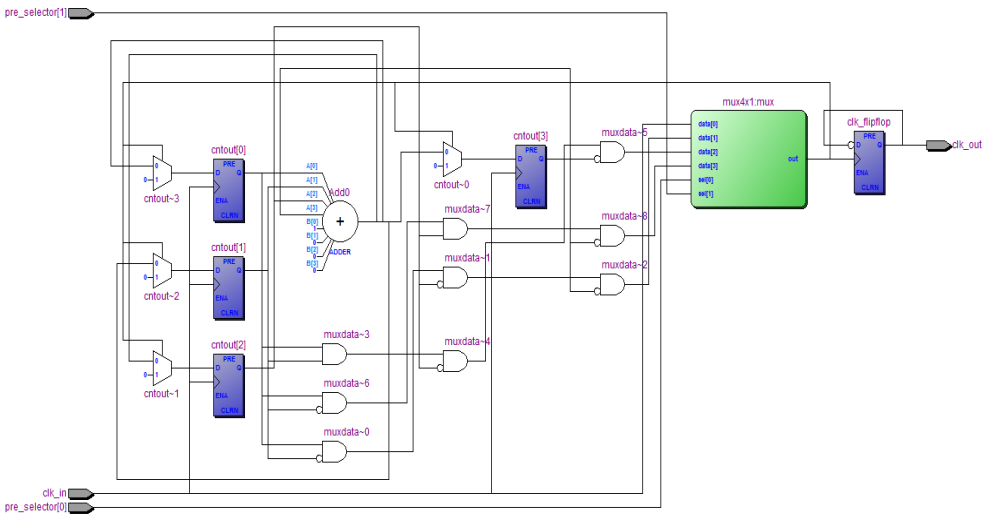
Şekil 3.26 Slack histogramı

Şekil 3.26 slack değerlerinin histogram halinde gösterimini ifade etmektedir. Yüksek gecikmeye sahip olan değerler buradan seçilerek değerlendirilmesi ve tasarımın optimize edilmesi gerekebilir ancak bu tasarımdaki slack değerleri optimize edilmiştir.



Şekil 3.27 Data delay waveform

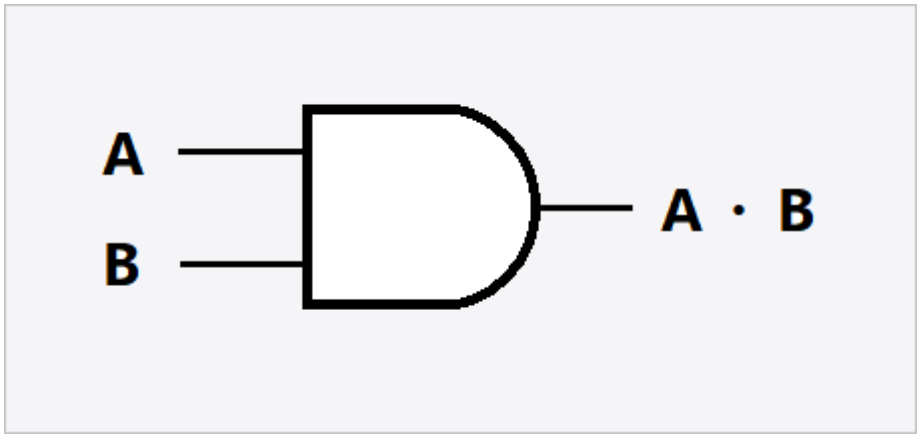
Şekil 3.27'de bir sinyalin hedeflenen yola ulaşması için gerekli zamanın gecikme değerleri gösterilmeketedir. Slack değeri 1.612ns ve toplam delay 8.404ns olarak ölçülmüştür. Bu değer KSA için baz alınan 10 ns ve üzeri gecikme değerinin daha altındadır. Elbette ki bu fark kullanılan platformlarla ilgili de olsa tasarım belirlenen şartlardan daha hızlı çalışmaktadır.



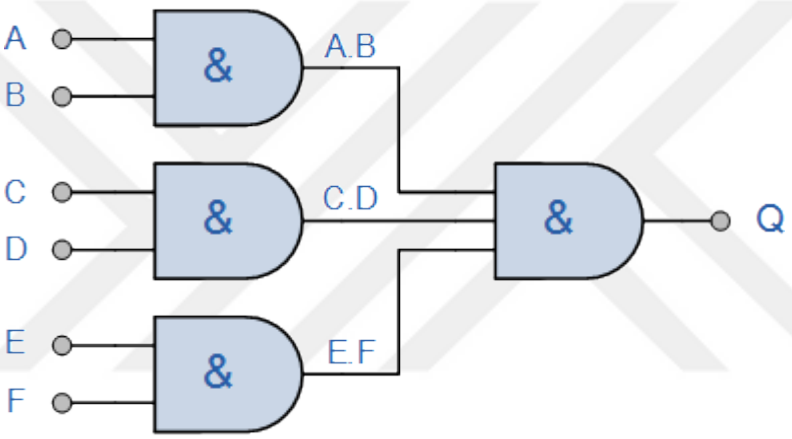
Şekil 3.28 RTL Netlist gösterimi

Tasarımı optimize etmek için kullanılan bir diğer araç ise RTL netlist gösterimidir.(Şekil 3.28) Bütün giriş çıkışlar registerlar lojik kapılar dahili modüller gösterilmekte ve tasarımın nasıl şekillendiğini gösterilmektedir.

Normal şartlar altında lojik kapılar birer bit olarak üretilmektedir (Şekil 3.29). Birden çok biti bir lojik kapıda değerlendirebilmek için birden fazla lojik kapı tercih edilmelidir (Şekil 3.30). Elbette yüksek sayıda girişe sahip and kapısı oluşturmak oldukça karmaşık bir hal almaktadır. Ayrıca oluşturulan devrenin bağlantı kalitesi ve kablo uzunlukları farklılaşabilir. Bu da bir and kapısının kararlılığı ve zaman gecikmelerinin referans bir değere sahip olmamasına sebebiyet vermektedir.

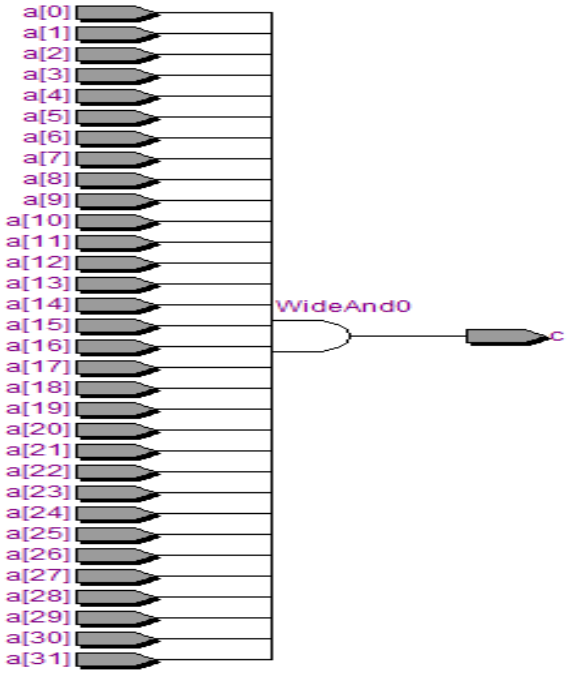


Şekil 3.29 And kapısı lojik gösterimi(URL-2,2020)



Şekil 3.30 6 girişli and kapısı (URL-6)

Çok girişli and kapısının birden fazla and kapısıyla oluşturulmasından kaynaklanan bu problemlerin önüne geçebilmek için FPGA üzerinde tasarlanabilen çok girişli and kapıları kullanılmalıdır. Burada FPGA içerisindeki yolların stabil ve belirli referans değerlere sahip olması bu problemi engelleyici birincil etken kabul edilmektedir (Şekil 3.31).



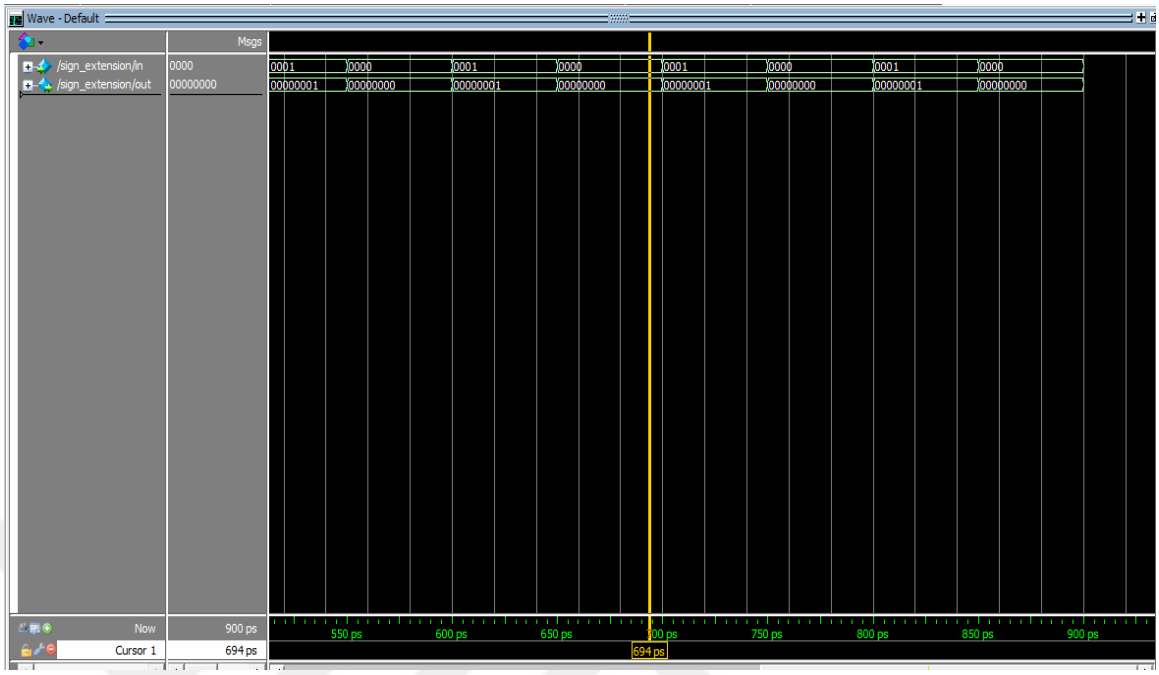
Şekil 3.31 32-bit girişli and kapısı

3.4.3. **ModelSim**

Tasarımların tamamı INTEL tarafından yayımlanan ModelSim yazılımıyla simüle edilmiş ve tasarımların simülasyonları yapılarak hataları giderilmiştir.ModelSim teorik olarak tasarımı test eder elde edilen sonuçlar waveform olarak gösterilir. (Şekil 3.32)

Elbette ki burada gösterilen zamanlar teorik zamanlara karşılık gelip hiçbir bağlantının veya lojik yapının gecikme sağlamadığı kabul edilerek gösterilir. Yani Şekil 3.28' de gösterilen gecikmeler ve slackler bu simülasyonlarda gösterilmemektedir. Bu simülasyon türüne RTL simülasyon denir. Bütün gecikme değerlerini gösteren simülasyonlarda vardır. Ancak Modelsim uygulamasının tam sürümüne sahip olmak gerekmektedir. Bu tip simülasyonlara ise Gate Level simülasyon denir. Model-Sim belirlenen FPGA türüne göre her bir mantıksal bloğun ortalama gecikmesini hesaplar ve

sinyal yollarının gecikmelerini de ekleyerek toplam gecikmeleri dalga formunda gösterir.



Şekil 3.32 ModelSim çıktıları

3.4.4. **Lojik Analizör**

Modelsim veya benzeri simülasyon platformları zamanlama olarak yeterince bilgi vermediğinde tasarımlarda ki zamanlama kaynaklı problemlerin tespit edilmesi mümkün değildir. Bu sebeple zamanlamaları ve her bir sinyalin davranışını gözlemlemek amacıyla lojik analizör kullanılmalıdır. Lojik analizörler, lojik tasarımlar için hata ayıklama araçları sayılabilirler. Her bir sinyalin gerçekte ne kadar geciktiğini, ya da diğer sinyallerle olan ilişkilerini incelemeye imkan sağlamaktadırlar. Yani başka bir ifade ile dijital dünyayı gerçek zamanlı olarak yakalayan, ölçen ve görselleştiren bir ekipmandır.

Lojik analizörler genelde test verisini pattern generator adı verilen test verisi üreteçlerinden alırlar. Bu üreteçler lojik sisteme ait bus girişlerinin her bir bitine binary sinyal bölme işlemi uygular. Yani her bir bit öncekinin 2 katı sürede tekrar eden lojik

1 ve lojik 0 sinyallerinden oluşur. Burada ki amaç tasarımın tamamının oluşturabileceği kombinasyonları test etmektir.

Bu çalışma da tasarlanmış bütün birimlerin testleri ve hata ayıklamaları lojik analizör tarafından yapılmıştır. Kullanılan lojik analizör pattern generator bulundurmamaktadır. Bu eksikliği gidermek amacıyla FPGA kullanılarak yeni bir pattern generator tasarlanmıştır. Test verileri FPGA üzerinden alınıp lojik analizöre gönderilmiş ve lojik analizörden alınan test verileri yine FPGA'e gönderilerek çalıştırılmış ve devamında da yine lojik analizör üzerinden okunmuştur.

3.4.5. **Power Play**

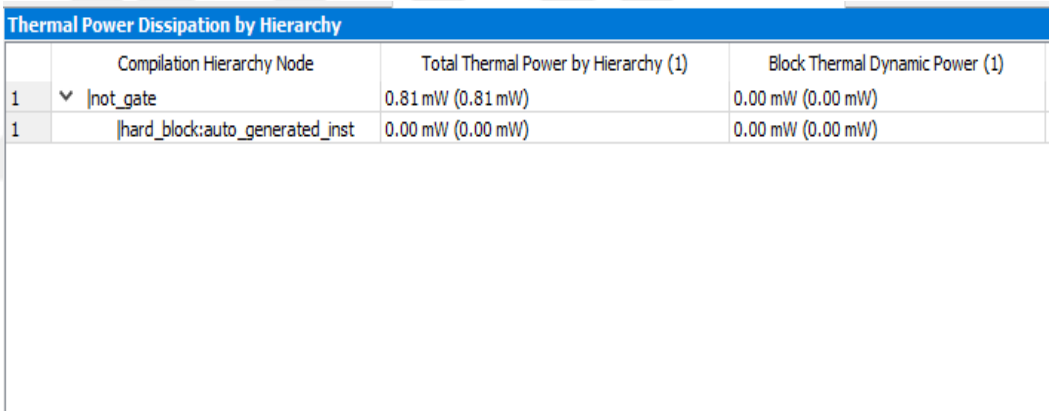
Tasarımların ne kadar enerji tükettiği günümüz de oldukça önem taşır. Çünkü taşınabilir teknolojik aygıtların güç gereksinimleri pillerle sağlanır. Bu bağlamda pilin ne kadar verimli kullanılacağı aygıt içerisinde güç tüketen tümleşik devrelerin, güç talebiyle ilgilidir.

Quartus yazılımı, tasarımcının güç optimizasyonu yapabilmesi için tasarlanan devrenin ne kadar güç talep ettiğini simüle ederek hesaplar. Burada temel esas talep edilen akım ve uygulanan voltajdır. Kaynak voltajının sabit olduğu göz önüne alınırsa kaynaktan çekilen akımın değeri direkt olarak güç ile doğru orantılıdır. Bu genel olarak tasarım için harcanan lojik hücrelerin güç gereksinimlerine göre belirlendiğinden üretimi yapılmış bir tasarımın yapısal ve sahip olduğu teknoloji ile değişimler gösterebilir

Bir diğer güç tüketimini belirleyen etmen tasarımın çalışma frekansıdır. Frekansın artması güç gereksinimlerinin de artması anlamına gelmektedir. Yani bir tasarımın farklı frekanslarda çalıştığı durumda talep ettiği güç miktarı da doğru orantılı olarak değişim göstermektedir. Şekil 3.33'te görüldüğü üzere Power Play Tool Analyzer tasarımın güç analizini yaptıktan sonra Quartus ara yüzünde bir rapor oluşturmasının yanında harici bir rapor dosyası da oluşturur.



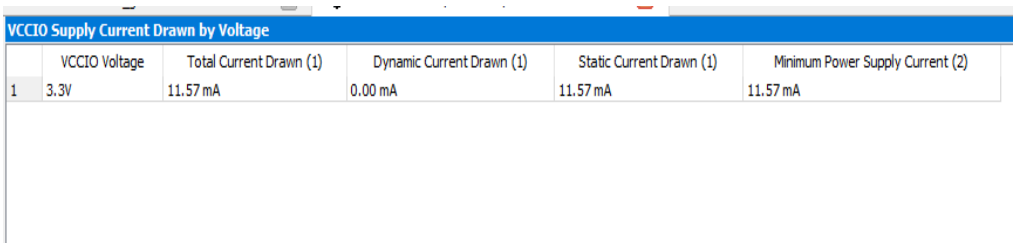
Şekil 3.33 Power Play Analyzer Tool ekran görüntüsü



Thermal Power Dissipation by Hierarchy

	Compilation Hierarchy Node	Total Thermal Power by Hierarchy (1)	Block Thermal Dynamic Power (1)
1	|not_gate	0.81 mW (0.81 mW)	0.00 mW (0.00 mW)
1	|hard_block:auto_generated_inst	0.00 mW (0.00 mW)	0.00 mW (0.00 mW)

Şekil 3.34 Power Play Quartus termal güç raporu



VCCIO Supply Current Drawn by Voltage

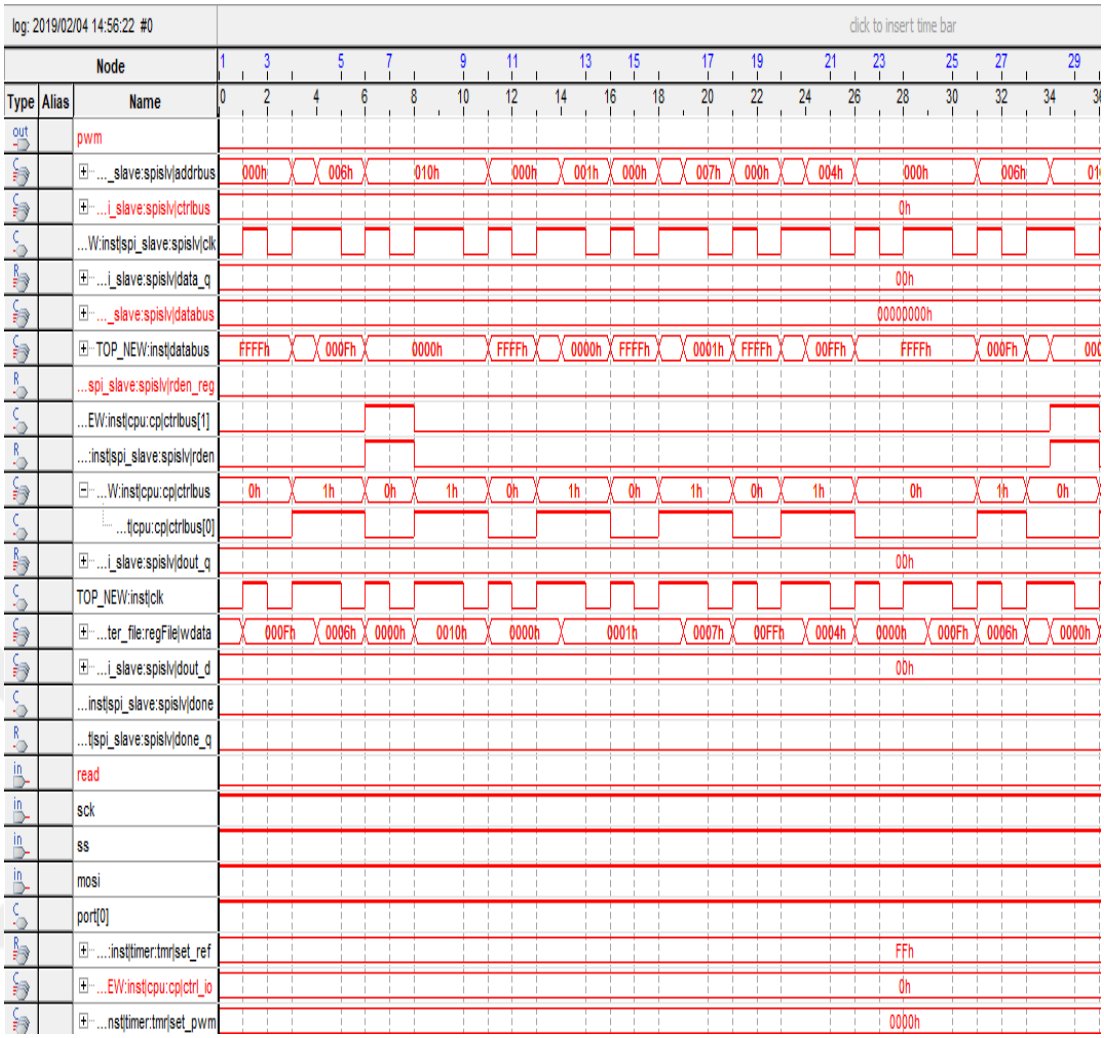
	VCCIO Voltage	Total Current Drawn (1)	Dynamic Current Drawn (1)	Static Current Drawn (1)	Minimum Power Supply Current (2)
1	3.3V	11.57 mA	0.00 mA	11.57 mA	11.57 mA

Şekil 3.35 Power Play kaynaktan çekilen akım Quartus raporu

Şekil 3.34'te görüldüğü üzere Power Play Analyzer termal güç tüketiminin ne kadar olacağını raporlamıştır. Ekran görüntüsü basit bir not kapısı üzerinde test edilip alınmıştır. 4 bitlik bir not kapısının toplamda talep ettiği güç 0.81mW olup kaynaktan çekilen akım miktarı 11.57 mA'dır (Şekil 3.35). Bu durumda çekilen akıma göre 3.3 V gerilim altında 11.57 mA tüketen bir devrenin toplam harcadığı güç 38.181mW olmalıdır. Ancak bu güç sadece tasarımın harcadığı güç değildir. FPGA çipinin ve kartının üzerinde harcanan güç ifadesidir. Bu durumda FPGA çipinin ve kartın harcadığı 38.1mW olarak hesaplanabilir.

3.4.6. **Signal Tap Logic Analyzer**

Altera firmasının ürettiği FPGA modelleri için hazırlanmış bir dahili lojik analizör ara yüzü bulunmaktadır. Bu ara yüz sayesinde lojik analizör ile görüntülenebilen dijital sinyalleri gerçek zamanlı olarak incelenebilir. Ancak burada ki sinyallerin zamanlamaları bir lojik analizör kadar hassas değildir. Temelde FPGA içerisindeki hafıza birimlerini kullanarak verilerin hafızda biriktirilip JTAG üzerinden bilgisayar ekranına gönderilmesi prensibiyle çalışır.



Şekil 3.36 Signal Tap logic analyzer ekran görüntüsü

Şekil 3.36'te görüldüğü üzere bir tasarımın dahili bütün birimlerinin görüntülenerek sinyallerin görselleştirilmesi ve gerçek zamanlı olarak incelenmesi signal tap lojik analizör ile sağlanmıştır. Bura da bir SPI haberleşme protokolü gerçeklenmiş olup veriler hexadecimal olarak ifade edilmiştir.

Node			Data Enable	Trigger Enable	Storage Enable	Storage Qualifier	Trigger Conditions
Type	Alias	Name	175	175	175	Transitional	1 Basic AND
		...i_slave:spislv|ctrlbus					Xh
		...W:inst|spi_slave:spislv|clk					
		...i_slave:spislv|data_q					XXh
		..._slave:spislv|databus					XXXXXXXXh
		TOP_NEW:inst|databus					XXXXh
		...spi_slave:spislv|rden_reg					
		...EW:inst|cpu:cp|ctrlbus[1]					
		...:inst|spi_slave:spislv|rden					
		...W:inst|cpu:cp|ctrlbus					Xh
		...t|cpu:cp|ctrlbus[0]					
		...i_slave:spislv|dout_q					XXh
		TOP_NEW:inst|clk					
		...ter_file:regFile|wdata					XXXXh
		...i_slave:spislv|dout_d					XXh
		...inst|spi_slave:spislv|done					
		...t|spi_slave:spislv|done_q					
		read					
		sck					
		ss					
		mosi					
		port[0]					
		...:inst|timer:tmr|set_ref					XXh
		...EW:inst|cpu:cp|ctrl_io					Xh
		...nst|timer:tmr|set_pwm					XXXXh
		...ve:spislv|databus_out					XXXXXXXXh

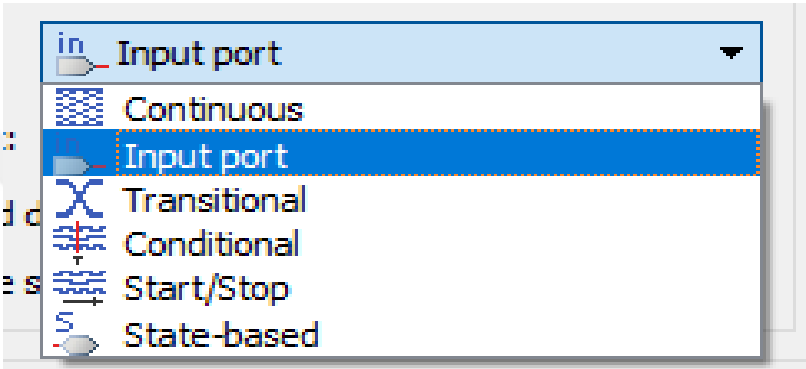
Şekil 3.37 Test parametrelerinin belirlenmesi

Şekil 3.37' te görüldüğü üzere test parametreleri tasarım içerisindeki veri yolları ve registerlardan oluşmaktadır. Bu sayede tasarımın hata ayıklaması gerçekleştirilmiştir. Burada kritik nokta test edilecek sinyallerin formlarının seçilmesidir. Bu test kurulumun da and kapılarıyla koşul şartları oluşturulmuştur. Yani tetikleme koşulu and kapılarıyla oluşturulmuştur.

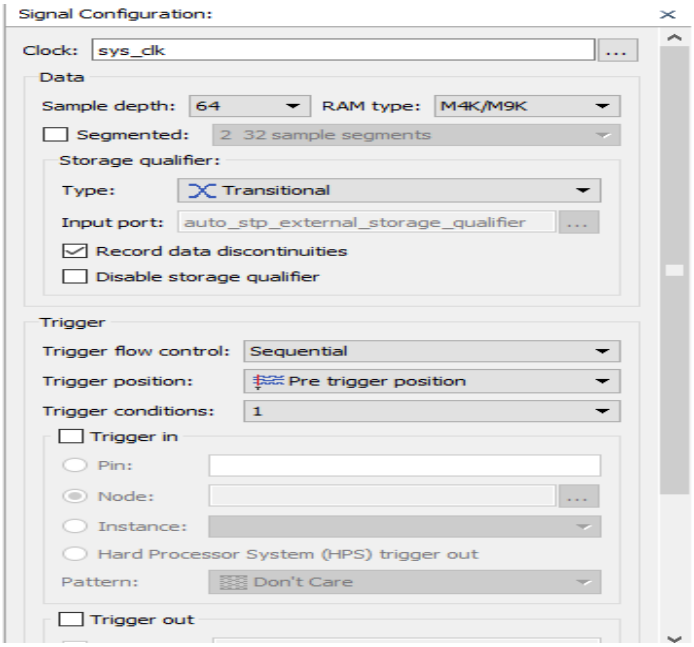
Tetikleme koşulları sinyalin belirlenen herhangi bir anındaki istenilen verinin incelenmesi durumlarında kullanılmalıdır. Sinyal formları ise geçişli sinyal (transitional) olarak belirlenmiştir. Bu tip sinyal formları genellikle verinin tamamının görülmesi incelenebilmesi için gereklidir. Zaman ile inceleme yapılması ve kararlılık testleri

için sürekli zamanlı sinyal formu kullanılmalıdır. Elbette ki farklı formlarda sinyaller de vardır. Bunlardan bazıları;

İstenilen zaman aralıklarında örnek almak amacıyla fiziksel portlardan verilen sinyale göre belirlenen sinyal formları, alınacak örneklerin belirli koşullar altındaki davranışlarını incelemek için önceden belirlenmiş şartlara göre şekillenen sinyal formları, belirli bir zaman aralığında sinyalleri incelemek için başla/bitir tipi sinyal formları ve incelenecek sinyallerin durum tabanlı sinyal formlarıdır(Şekil 3.38).



Şekil 3.38 Sinyal formlarının seçimi



Şekil 3.39 Sinyal konfigürasyonu

Sinyal formlarının belirlenmesi için sinyal konfigürasyon sekmesi kullanılmaktadır. Burada Clock olarak belirtilen referans clock sinyalidir. Yani her bir örneklem için gerekli olan zamanlama referansıdır. Data sekmesinin altında bulunan sample depth alınan örneklerin nasıl depo edileceği ve aynı zamanda ne kadar örnek alınacağını belirten konfigürasyon parametreleridir.

M4K/M9K FPGA çipinin dahili hafıza birimlerini temsil eder. Bu tipte ki hafıza birimlerinin seçilmesi FPGA içerisindeki lojik blokların işgal edilmeden olabildiğince fazla sinyali görünteleme amacı taşımaktadır. Triger sekmesi altında bulunan tetikleme kontrol akışı, sıralı olarak seçilmiştir. Yani alınan bütün örnekler seri olarak tetikleme şartına uyup uymadığı kontrol edilirken, tetikleme poziyonu ise sinyalin tetiklemenin başladığı noktadan itibaren olarak seçilmiştir. Bu tetikleme pozisyonu, tetikleme sinyalinin diğer sinyallerle olan ilişkisini kurmak amacıyla tercih edilmelidir (Şekil 3.39).

4.ARAŞTIRMA BULGULARI ve TARTIŞMA

Tasarlanan mikro işlemcinin temel bileşenlerini MIPS mimarisi oluşturur bu bağlamda MIPS mimarisine ait olan temel bileşenler tasarımda tekrar edilmiştir. Elbette ki bu tasarımda bazı değişliklere gidilmiş olup standart yapılardan ayrılmaktadır. Bu tasarımlar MIPS mimarisini iyileştirmek ve geliştirmek anlamında önem taşımaktadır. Elbette ki bu tasarımda bazı değişliklere gidilmiş olup standart yapılardan ayrılmaktadır.

1) MIPS ISA'da bulunan (Instruction Set Architecture) branch ve jump komutlarının optimize edilmesi ve bu işlemlerin tek bir blokta toplanmasıdır.Program Counter bloğunun içerisinde oluşturulan yapıda branch jump, beq(branch if equal),bne(branch not equal) komutları direkt olarak control unit üzerinden -kontrol sinyalleri vasıtasıyla taşınan sinyallerle- kontrol edilmektedir. Standart yapılarda bu işi yapan yapılar blok olarak değil de dağınık olarak bulunur. Bu durumda kontrol sinyallerinin sayısı artmakta ve bu işlemcide bazı kararsızlıklar yaşatabilmektedir. Çünkü işlemci içerisinde bulunan diğer kontrol sinyalleri ile zaman zaman çakışmalar yaşanabilir.
   a) Yapılan tasarımda ise bütün bu komutlar bir bloğa tek bir sinyal girişiyle bağlanmıştır ve bu da bu kararsızlık durumunun önüne geçmiştir
   b) Blok tasarımında ise kontrol sinyali girdisi bir MUX(multiplexer) ile kontrol edilip Program counter yazmacını dinamik olarak değiştirmektedir.
   c) Branch için gerekli olan göreceli adresleme işlemi daha kolay yapılmakta çünkü bunun için ekstra bir yazmaca ihtiyaç duymamaktadır.

2) Bir diğer değişiklik ise işlemciden çıkan ve çevresel birimlere bağlanan databus, addressbus ve controlbus gibi yolların kontrolünde kullanılan tri state buffer kontrolleri yine direkt olarak kontrol ünitesinden sinyal almaktadır. Yani standart yapıda kullanılan komutları çözen(decode) muxlar ve yapılar dinamik olarak çevresel birimlere giden veri adres ve kontrol yollarını da kontrol et-

mektedir. Bu da sentezlenen her kontrol sinyalinin dinamik olarak kullanılabildiğini ve bunun sonucunda da tam anlamıyla senkronize kontrol sinyalleri sentezlenerek işlemcinin kararlılığı arttırılmıştır.Bu yöntem için işlemcinin içerisinde basit bir kaç lojik yapı oluşturulmuş sinyalin dinamik kullanılması sağlanılmıştır.

3) Ayrıca popüler olarak kullanılan bir çok mikro işlemciden farklı olarak Aritmetik lojik ünite çarpıcı ve bölücü (32 bit) içermektedir. Bu da mikro işlemcinin çarpma işlemi yapması için toplayıcı devreyi üst üste çalıştırması veya bölücü için çıkarıcı devreyi üst üste çalıştırması gerekmemekte bu da işlem gücü hız öncelikli uygulamalarda büyük avantajlar sağlamaktadır.

4) Güç Optimizasyonu: Bunların yanı sıra standart çevresel birimler de tasarlanmıştır. Tasarlanana diğer çevresel birimler (SPI, Prescaler, Timer vs) data yollarına ve adres yollarına bağlanmış ve kontrolleri adresleme yapılmış yazmaçlarla yapılmıştır.

Bir diğer çevresel birim olan RAM (Random Acces Memory) tasarımında 8 bitlik 4 adet farklı ram bloğu şeklinde yapılmış ve saklanacak verinin uzunluğuna göre bu 4 farklı ram blokları aktif veya de aktif olmaktadır. Yani veri 32 bit ise verinin ilk 8 biti ilk ram bloğuna ikinci 8 biti ikinci ram bloğuna kalan 8 er bitleri ise sırasıyla üçüncü ve dördüncü ram bloklarına yazılır. Bu işlem için ekstra işlem gücü veya clock harcanmaz paralel olarak bütün veriler kaydedilir. Bu işlemi kontrol eden kontrol sinyalleridir.

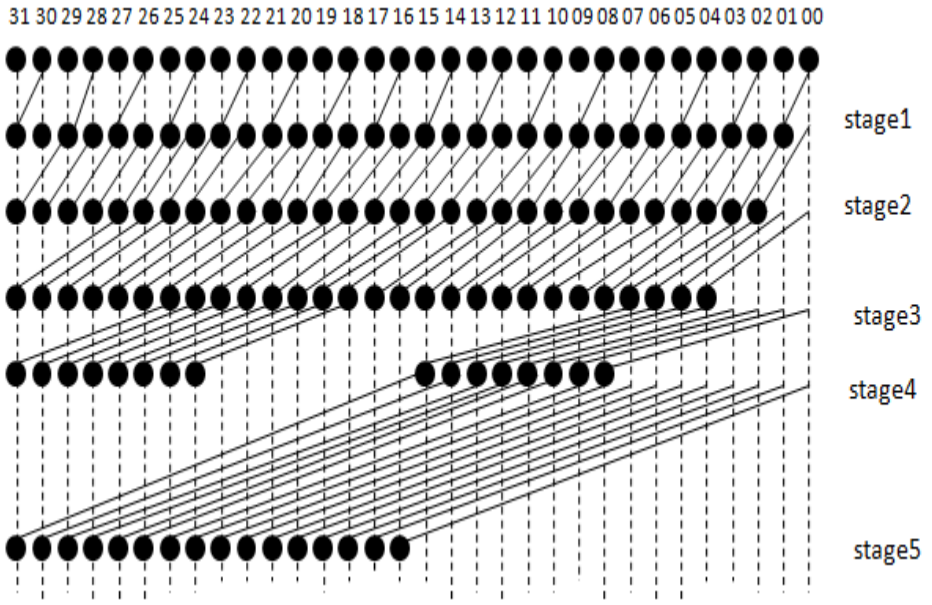
Alınan verinin uzunluğuna göre ilgili ram blokları aktif edilir ve kaydetme başlar. Bunun en büyük avantajı ise ram bloklarının sürekli olarak aktif kalmayıp güç tüketimi arttırması ve bunun yanı sıra yüzde yüz verim ile RAM kullanımı yapılabilmesi. Eğer tek bir RAM bloğuna sahip olsaydı 32 bitlik bilgi saklama kapasitesi bulunan her bir adrese 16 bitlik bir veri yazıldığında o adreste kalan 16 bit kullanılamayacaktır bu da kararlılık ve güç anlamında dezavantajdır.

4.1. Program Counter

Program counter, program akışını değiştirebilen tek birimdir. Bu değişimler jump, branch, jump a link, jump a register gibi komutlarla sağlanabilir. Bu komutlar standart bir MIPS yapısında mikro işlemci mimarisinde çeşitli lojikler ile sağlanır. Tasarlanan Program Counter ise jump ve branch gibi komutları işleyebilen tek bir yapı olarak düzenlenmiştir. Bu sayede jump ve branch gibi program akışını değiştirip düzenleyebilecek olan komutların stabil olarak çalışması ve yazılımın kararlı çalışmasını sağlamaktadır.

4.2. ALU

Tasarımın ALU biriminde Kogge-Stone tipi bir adder kullanılmıştır. Bu adder güncel adder tasarımları arasında en az gecikme değerine sahip olan adder olma özelliği taşımaktadır. Paralel Prefix Adder ailesine aittir. Bu tip adderlar toplama işlemlerini paralel olarak parçalayıp hesaplarlar ve sonuçta elde edilen veri toplama sonucunu verir. (Şekil 4.1)



Şekil 4.1 Kogge Stone Adder

Bu tip toplayıcılar arasında en bilindik olan Ripple Carry Adder (RCA), Carry Look-Ahead Adder(CLA) ve Kogge-Stone Adder (KSA)'dır

Type of Adder	Delay		
	Maximum Combinatioanl Path Delay(in ns)	Logic Delay(in ns)	Route Delay(in ns)
4 Bit RCA	12.008	7.540	4.468
4 Bit CLA	12.008	7.540	4.468
4 Bit KSA	11.786	7.540	4.246
8 Bit RCA	17.585	9.456	8.129
8 Bit CLA	17.585	9.456	8.129
8 Bit KSA	17.030	9.456	7.574
16 bit RCA	28.741	13.288	15.453
16 Bit CLA	27.076	13.288	13.788
16 Bit KSA	21.264	10.893	10.371

Şekil 4.2 Paralel Prefix Adder karşılaştırması(BAIS, 2016)

Şekil 4.2'de görüldüğü üzere KSA en az gecikme değerine sahip toplayıcı devre yapısıdır. (BAIS, 2016) Bu sebeple tasarımda KSA kullanılmıştır.

ALU'da bulunan bir diğer hesaplama birimi olan çarpıcı devre tasarımında ise Wallace Tree tipi bir çarpıcı devre kullanılmıştır. Wallace Tree yine paralel hesaplama yapabilen ve yapısında ki toplayıcı devrenin KSA olması sebebiyle hızlı hesaplama yapabilmektedir.

4.3. Control Unit

Control unit standart bir MIPS yapısının yanı sıra kontrol sinyallerinin optimizasyonu ve kararlılığının arttırılabilmesi için tasarım için optimize edilerek sentezlenecek her bir sinyalin bağımsız çalışması sağlanmıştır. Elbette ki Control Unit tek ba-

şına çevresel birimlerin kontrolünden sorumlu değildir. Her bir çevresel birim için harici bir kontrol birimi vardır. Bu birimler control unite dolaylı olarak bağlıdır. Yani kontrolcü sinyalini sentezleyen kontrol sinyallerinin türevleridir.

4.4. Çevresel Birimler

Bir mikro işlemci dış dünya ile irtibat kurabilmesi için çevresel birimlere ihtiyaç duyar bu çevresel birimler genellikle mikro işlemci mimarisine dahil edilmeyip adres ve veri yollarıyla mikro işlemciyle iletişim kurması sağlanır. Her bir çevresel birim kullanım uygulamalarına göre avantaj sağlamakla birlikte mikro işlemcinin esnekliğini geliştirir.

4.4.1. **I/O Unit (Giriş/Çıkış Birimi)**

I/O birimleri fiziksel olarak tasarıma eklenen ve dış dünyadan alınacak verilerin işlemciye doğru bir şekilde ulaştırılmasını sağlamaktadır. Örneğin bir sıcaklık sensöründen okunacak veri I/O biriminden alınarak işlemciye ulaştırılır. Ya da bir motor sürücüsü için sağlanacak olan PWM(Pulse Width Modulation) I/O tarafından sağlanır

I/O birimi hem giriş hem çıkış olarak ayarlanabilen pinler olmak zorundadır. Bunun sebebi giriş ve çıkış pinlerinin ayrılması durumunda oluşacak olan pin sayısının fazlalığı ve üretim maliyetlerinin artmasıdır. Ayrıca her pin için harcanacak güç miktarının artıp toplam güç miktarının fazla olmasına sebebiyet vermektedir.

I/O birimi okuma/yazma veya giriş/çıkış olarak seçilebilen aşamaları içermektedir. Bu sebepten dolayı adreslenebilir registerlara sahiptir. Bu registerlar;

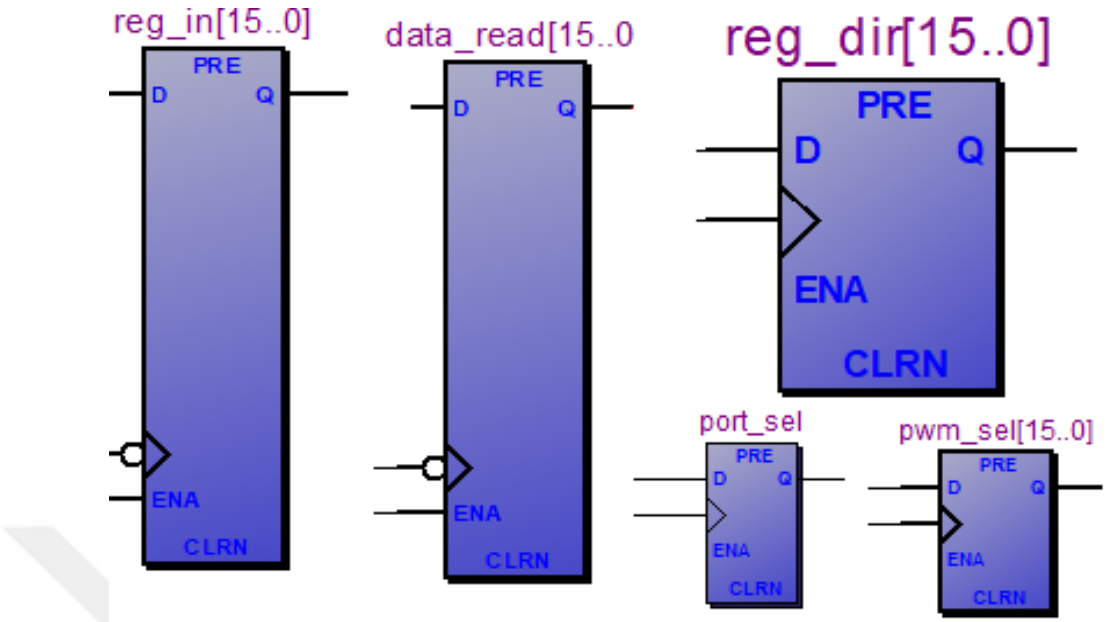
DIR: Portların giriş mi çıkış mı olduğunu belirleyen

IN: Giriş olarak ayarlanan portun verisini tutan

OUT: Çıkış olarak ayarlanan portun verisini tutan

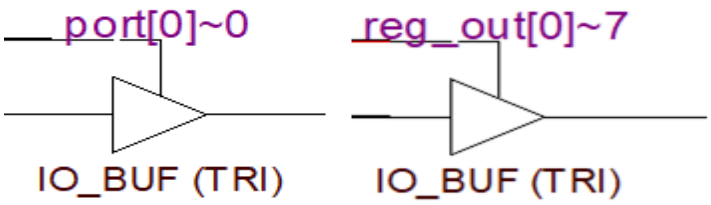
SEL: PWM olarak kullanılacak portun verisini saklayan

PMSEL: Portun pwm veya varsayılan olarak değiştirilmesini sağlayan özelleştirilmiş registerlardır (Şekil 3.4).



Şekil 4.3 I/O registerları

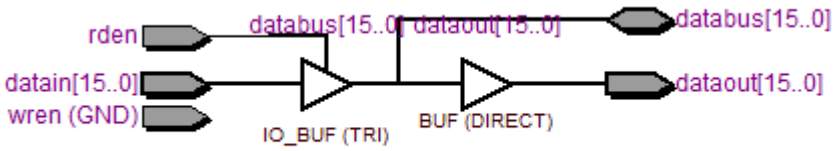
Registerların çıkışları tri-state buffer için kontrol girişlerine bağlıdır. Bu sayede bir port hem giriş hem çıkış olarak kullanılabilmektedir. Tri-state bufferlar girişlerindeki verileri kontrol sinyali aktif olana dek çıkışlarına iletmezler. Başka bir ifade ile giriş ve çıkışları anahtarlamış olurlar. (Şekil 4.3)



Şekil 4.4 Tri State buffer gösterimi

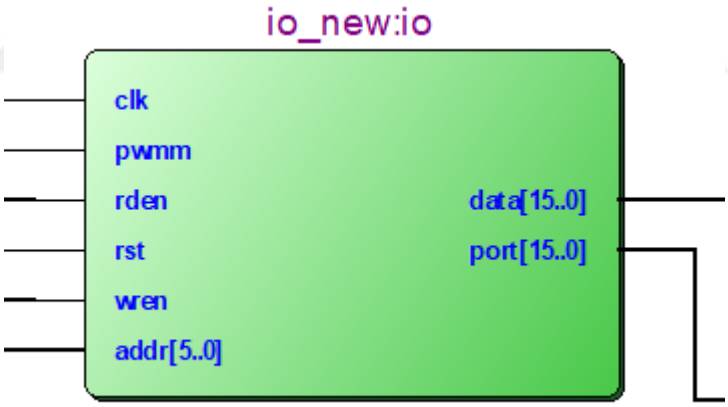
Bir portun giriş çıkış olarak kullanılabilmesi için özel bir lojik tasarım gereklidir. Bu tasarım iki adet tri state buffer kullanılarak yapılır. Giriş ve çıkış verilerinin bir

birileri ile çakışmamaları amacıyla aktif olan tri state buffer diğerini tersleyerek deaktif konuma getirir. (Şekil 4.4)



Şekil 4.5 Giriş çıkış kontrol lojik gösterimi

I/O biriminin girişleri clock, PWM birimi tarafından sentezlenmiş PWM sinyali, porta veri yazılıp yazılmayacağını kontrol eden wren, porttan veri okunup okunmayacağını kontrol eden rden, saklanan verileri sıfırlayacak olan rst ve adres bustan oluşmaktadır. Çıkışları ise direkt olarak fiziksel portlara ve data bus a bağlıdır. (Şekil 4.6)



Şekil 4.6 I/O birimi blok gösterimi

4.4.2. **SPI**

Serial Peripheral Interface olarak tanımlanabilecek bu birim seri haberleşme modüllerinden bir tanesidir. İlgili seri haberleşme protokollerini kullanan aygıtlardan aldığı veriyi işleyerek mikro işlemciye yönlendirir. Alınan veri okunmak istendiği takdirde SPI yazmaçlarına erişim sağlanarak veri alınır ve mikro işlemci de işlenir. Tasarlanan SPI modülü iki adet olup 16 bitlik yazmaçlara sahiptir.

SPI tasarımı için temel alınan tasarımda farklılıklara gidilerek adresleme sistemi değiştirilmiştir. Bunun sebebi standart tasarımlarda bulunan adres çakışmalarının önüne geçmektir. SPI tarafından alınan veri mikro işlemci tarafından gönderilen adres bilgisiyle okunmaktadır. Ancak SPI birimi için ayrılan registerlar doğrudan veri yoluna bağlıdır. Bu sebepten dolayı gönderilecek olan veri adres koşulu sağlandığı takdirde direkt olarak veri yoluna düşmesine ve bundan dolayı da adres çakışmalarının ortaya çıkmasına sebebiyet vermektedir. Bunun önüne geçmek için registerlar bitlerinin her biri tri-state bufferlar tarafından kontrol edilmiştir.

SPI kendi içerisinde SPI_Slave ve SPI Master olmak üzere iki birimden oluşur. Bu SPI haberleşmenin temelini oluşturan iki birimdir. SPI protokolünü kullanan bir aygıt ile haberleşme yapılacağı zaman, aygıtın karakteristiğine göre slave veya master olarak seçim yapılır. Bir birine göre ters çalışan bu iki birim aldıkları verileri registerlar üzerinde saklayarak, mikro işlemci tarafından okunana kadar tutarlar.



Şekil 4.7 SPI Slave birimi

Şekil 4.7 de gösterildiği üzere SPI slave/master birimi mosi (master out slave in) girişine yani master aygıtın çıkış sinyaline, sck (slave clok) slave veri okumanın senktronizasyonu ve kaydırma işleminin aktif olacağı döngüyü taşıyan sinyale, ss(slave select) birimin slave/master olarak seçen sinyale, okumanın başlamasını kontrol eden rden sinyaline ve reset girişlerine sahiptir. Çıkış olarak ise miso (master in slave out) yani slave olarak kullanılan aygıtın çıkış sinyaline, verilerin işlemci tarafından okunabilmesi için databus hattına ve okuma yazma işleminin bittiğini belirten done sinyaline sahiptir.

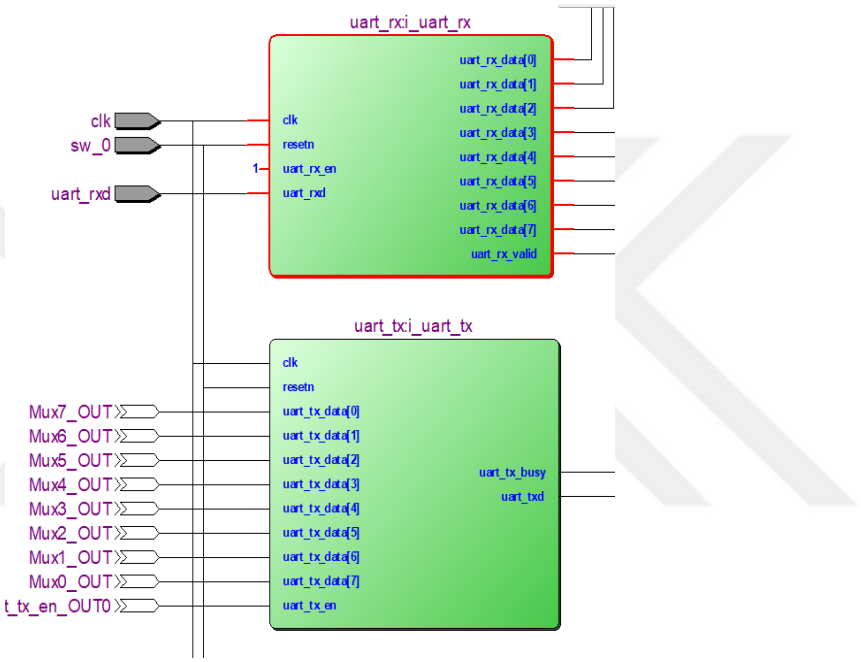
4.4.3. **UART**

Universal Asyncronus Reciever/Transmitter olarak tanımlanan UART seri haberleşme modüllerinin bir diğeridir. Bu tip seri haberleşme protokolünü kullanan aygıtlarla iletişimi sağlayarak verilerin akışını sağlar. Tasarımda UART modülü 11520 baud rate kullamakta ve okuduğu verinin genişliği dinamik olup 32/16/8 bit gibi değerlere sahiptir. İki adet UART modülü tasarlanmıştır.

Temel alınan tasarım üzerinde değişikliklere gidilmiştir. Bunlardan en önemlileri UART modülü, kendi stack yapısını kullanmaktadır. Bu saye de mikro işlemciyi meşgul etmeden verileri saklayıp öncelik atayabilmektedir. Mikro işlemci yapısında kesme öncelik tanılama birimi vardır. Bu birim çevresel birimlerden kesme sinyallerinin önceliklerini tayin eder. Bu sistem kendi içerisinde bir protokol barındırmaktadır. UART üzerinden alınan sinyalin öncelik sırasını belirten bit dizilimleri ile şifrelenmiş veriyi çözerek öncelikleri belirler.

UART modülleri alıcı(rx) ve gönderici (tx) gibi iki birimden oluşmaktadır. (Şekil 4.8) Bu birimler iletişimin kurulacağı cihazlara alıcı-gönderici, gönderici- alıcı olacak şekilde bağlanırlar. Bağlantılar birer pin olarak yapılır. Alınan her bir pinde ki değerler modül içerisinde ki registerlara yazılarak kaydırılır. Ne kadar kaydırılacağını kontrol eden kontrolcü birimler vardır. Bu birimler alınan verinin uzunluğunu okudukları registerlara sahiptirler. Bu sayede alacakları bit genişliği kadar çalışacak sayaç başlatılır ve sonuç olarak öteleme işlemi bit sayısı kadar yapılır.

Alınan veri modül içerisindeki stack dizinlerinde saklanarak mikro işlemci tarafından okunmaya hazırlanır. Ayrıca alınan veri tamamlandığında mikro işlemci için kesme yolu üzerinden verinin hazır olduğunu bildiren kesme sinyali açılır. Bu kesme işlemi mikro işlemci tarafından okunacak değerin doğruluğu için önemlidir. Çünkü verinin tamamının alınmadığı ve mikro işlemcinin okuma yaptığı bir senaryoda, gönderilen verinin tamamı okunmadığından eksik veri ile mikro işlemci sonsuz döngülere girebilir.



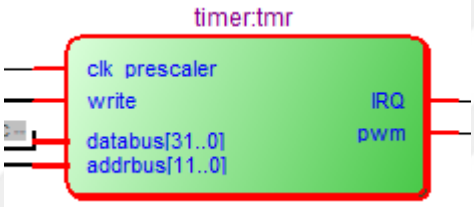
Şekil 4.8 UART Rx Tx birimleri

4.4.4. **Timer/PWM/Prescaler**

Timerlar mikro işlemci için dış dünyadaki zamanı simüle etme aracı olarak tanımlanabilir.Mikro işlemci içerisinde zaman gerektiren her uygulamada kullanılmaktadırlar. Tasarlanan timer birimi 32 bit veri genişliğine sahip yazmaçlardan oluşmaktadır. Bu sayede zaman hesabını oldukça hassas yapabilmektedir.

PWM birimleri değiştirilebilir döngü yoğunluğuna sahip görece yüksek voltajlı kare dalga oluşturmaktadırlar. Tasarımdaki PWM timer ile birlikte çalıştığından, yazmaçları 32 veri genişliğine sahiptir. Bu sayede yüksek çözünürlüklü PWM sinyalleri sentezleyebilir.

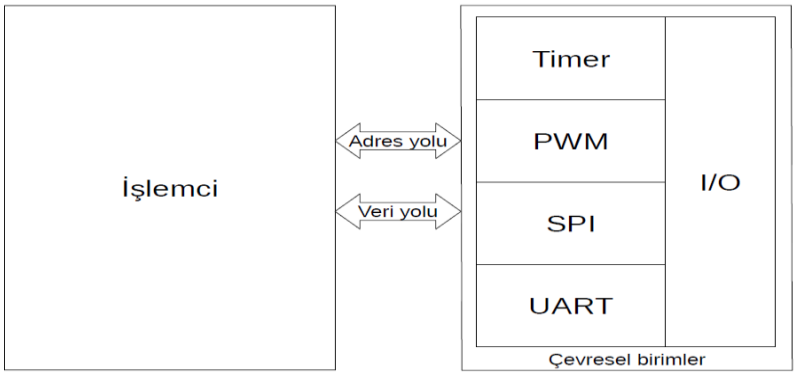
Prescaler mikro işlemci için gereken clock sinyani bölerek frekansının arttırılabilmesine olanak sağlayan bir birimdir. Tasarımdaki Prescaler ise 16 bit kadar clock frekansını bölebilir. Bu hassasiyetin öncelikli olduğu alanlar için oldukça önemlidir. Örneğin endüstriyel bir step motorun veya cerrahi bir robotun kinematik hesaplarına karşılık gelen denklemlerde ki motorların bir derecelik dönmesindeki hassasiyet oldukça önemlidir. Bu bir derecelik dönmeden sorumlu olan sinyalin hassasiyeti sentezlendiği birimin bit genişliği kadardır.



Şekil 4.9 Timer modülü blok gösterimi

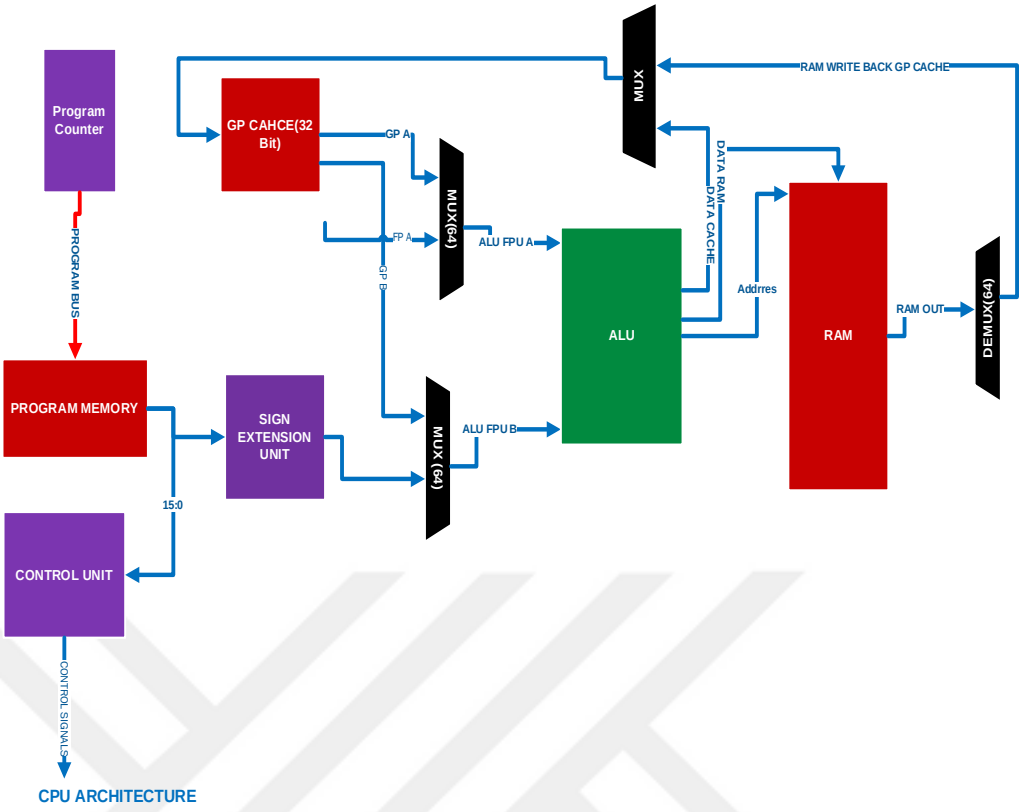
Timer modülü içerisinde PWM ve prescaler birimleri bulunmaktadır. Bu üç birim birbirinden aldıkları sinyallerle çalışmaktadır. Girişlerinde prescalerden geçmiş olan clock sinyali, yazma kontrol sinyali olan write, mikro işlemciden veri alıp veri gönderebilmesi için databus ve gönderilen her verinin doğru registerlar tarafından okunabilmesi için adresbus bulunmaktadır. Çıkışlarında ise, timer kesmesi olan IRQ ve PWM kesmesi olan pwm bulunmaktadır. (Şekil 4.9)

4.5. Mikroişlemci Genel Yapısı



Şekil 4.10 Tasarlanan mikroişlemci blok gösterimi

Yapılan tasarım çevresel birimlerden ve mikro-işlemciden oluşmaktadır. Çevresel birimler verileri alarak yazmaçlarında saklar ve işlemci ise yazmaçlara erişerek alınan verileri veri yollarıyla okur. Burada mikro işlemci ve çevresel birimler ayrı birer yapı olarak gösterilse de çevresel birimler adres ve veri yollarıyla birbirilerine bağlıdır. Veri yolu çift yönlü bir bus hattını göstermektedir. Tıpkı I/O ünitesinde ki çift yönlü okuma ve yazma yapıları veri yolu içinde geçerlidir. Çift yönlü olması verilerin taşınmasındaki kararlılığı olumsuz yönde etkileyebilmektedir. Bu olumsuzlukların önüne geçebilmek adına databus kontrolcüsü tasarlanmış olup kararlı bir şekilde verilerin akatarılabilmesi sağlanmıştır. (Şekil 4.10)



Şekil 4.11 Tasarlanan mikro işlemci blok diyagramı

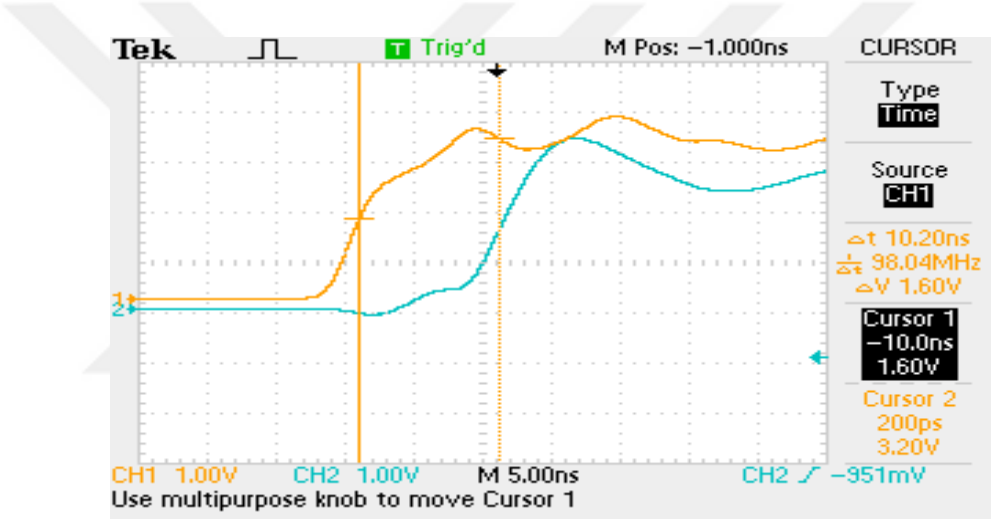
Şekil 4.11, tasarlanan mikro işlemcinin genel mimarisini göstermektedir. Burada program counter sıfırdan başlayarak birer birer saymaya başlar. Bu sayma işleminden elde edilen veriler program memory'e adres olarak verilir. Ve kodlayıcı tarafından yazılmış olan kodlar satır satır okunmaktadır.

Program memoryden çıkan kodların 16 bitlik kısmı control unit'e ve kalan 16 bitlik kısmı işaret uzatma birimine girer. Control unit, gerekli kontrol sinyallerini sentezledikten sonra ALU birimine girecek olan veriler hazırlanır. Burada gönderilen kodun tipine göre ALU verileri belirlenir. Eğer I tipi bir kod gönderilmiş ise işaret uzatma biriminin çıkışı direkt olarak ALU'ya bağlanır. Eğer R tipi bir komut gönderildiyse register adersleri control unit tarafından register file birimine yönlendirilir. Ve kod ALU üzerinde çalıştırılarak elde edilen veri RAM üzerinde ya da cache birimlerinde saklanır. (Şekil 4.11)

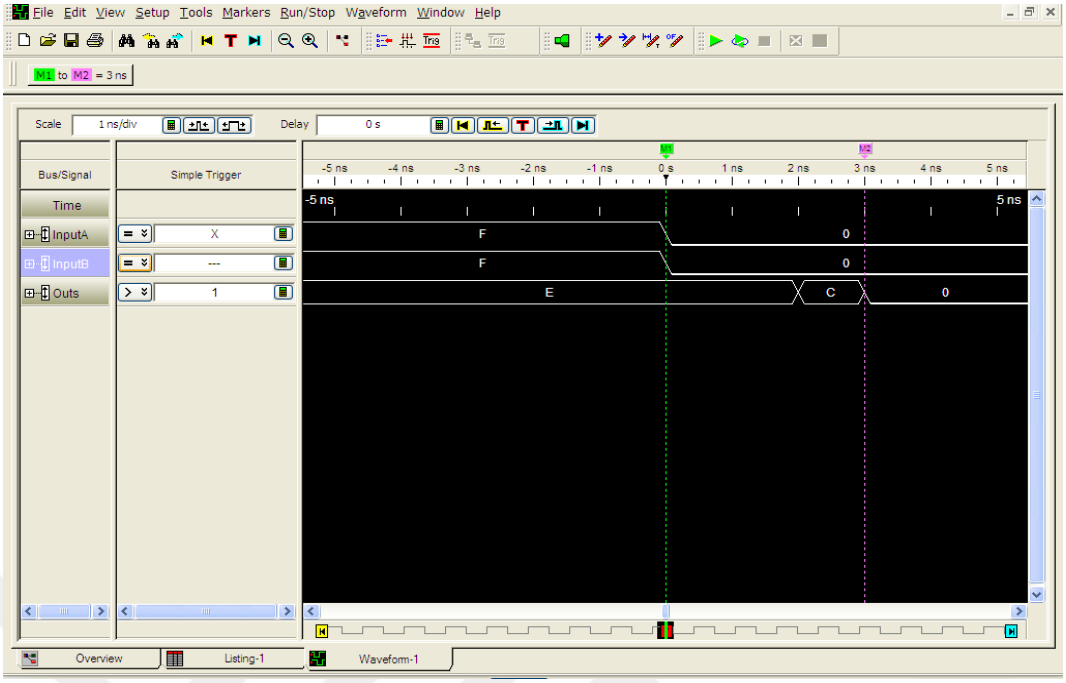
4.6. Test ve ölçümler

Tasarımı tamamlanan işlemci lojik analizör kullanarak test edilmiştir bu testler neticesinde her bir birimin senkronize ve mimariye uygun bir şekilde çalıştırılması sağlanmıştır. Ayrıca bu testler neticesinde birimlerin gecikme değerleri tespit edilmiştir.

Şekil 4.12' de görüldüğü üzere KSA gecikmesi osiloskop üzerinden okunmuştur. Burada referans gerilimi 3.2V ve zaman gecikmesi 10.20 ns olarak ölçülmüştür. Ayrıca test frekansı 98.04MHz'dir

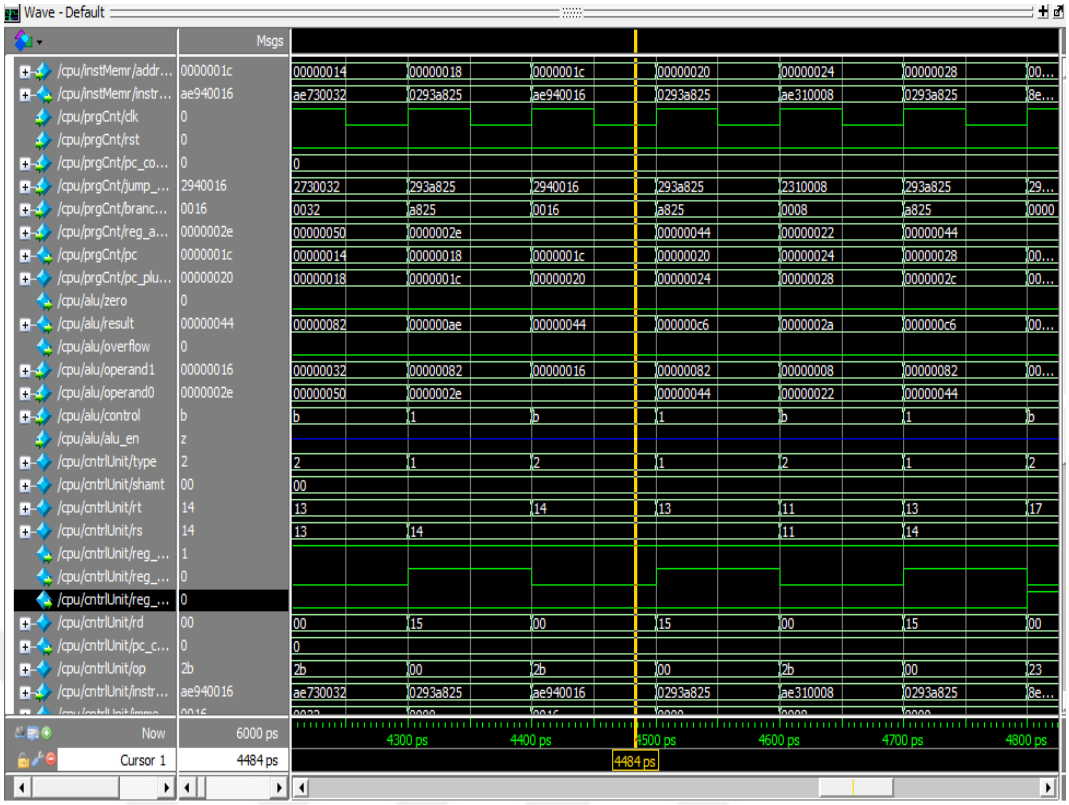


Şekil 4.12 Kogge-Stone adder osiloskop çıktısı



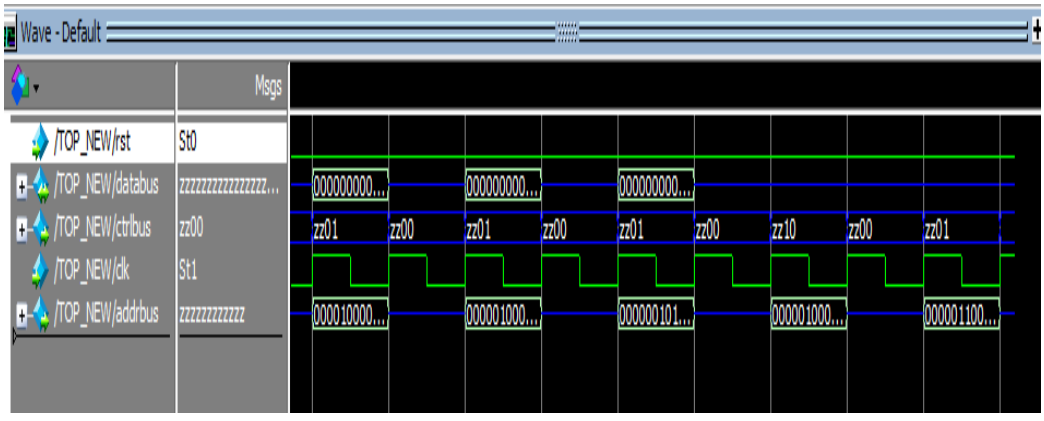
Şekil 4.13 Kogge-Stone adder analizör çıktısı

Şekil 4.13 KSA toplayıcısının LA üzerinden okunan en düşük gecikme süresini göstermektedir. Burada testler hex veri formatında yapılmıştır. Giriş olarak tanımlanan INPUT A ve INPUT B toplayıcı devrenin işlem verileridir. OUTs olarak tanımlanan değer ise çıkış verileridir. Giriş değerleri olarak verilen hex olarak F değerlerinin toplanması sonucunda çıkış 0 değerini göstermiştir. Bu durumda M1 işaretçisi verilerin gönderildiği ve M2 işaretçisi ise çıkış verisinin geldiği noktayı temsil etmektedir. Bu durumda M1 to M2 ifadesi de aradaki gecikmeyi sağlayan değerdir. En düşük KSA gecikmesi 3 ns olarak ölçülmüştür.



Şekil 4.14CPU Simülasyonu

Şekil 4.14 CPU'nun ModelSim üzerindeki simülasyonunu göstermektedir. IM'ye yazılan bir hesaplama programının veri akışları ve her bir birimin spesifik sinyalleri gösterilmektedir.



Şekil 4.15 Adress , Data ve Control Bus simülasyonu

Şekil 4.15'te görüldüğü üzere çevresel birimler eklendikten sonra CPU mimarisinin temel taşıyıcıları olan AB ve DB üzerinden geçen sinyallerin akışı ModelSim ile simüle edilmiştir. Burada ki kazanım, AB ve DB hatlarının kararlılığının gözden geçirilmesi ve iyileştirilmesidir.

5.SONUÇLAR ve ÖNERİLER

Tasarıma eklenen çevresel birimler ile birlikte bir mikro denetleyici elde edilmiştir. Tasarımı ve testleri tamamlanan mikro denetleyici güncel tercih edilen mikro denetleyicilerle kıyaslanmak amacıyla testlere tabi tutulmuştur. (Testler her mikro denetleyicide aynı uygulama çalıştırılarak gerçekleştirilmiştir)

PIC18F4520 mikro denetleyicisi ile ADC'den (Analog Digital Converter) okunan analog değerin (potansiyometre ve sensör) PWM (Pulse Width Modulation) referansına aktarılarak fiziksel porttan alınan PWM sinyalinin görev döngüsü (Duty Cycle) değiştirilmiştir. Bu şartlar (8MHZ osilatör frekansı ve oda sıcaklığında) altında mikrodenetleyicinin çektiği akım (Aktif Modda) 7mA ve besleme gerilimi 5V olarak ölçülmüştür. Bu mikro denetleyicinin kaynaktan tükettiği güç ise P(Güç)=I(Akım)xV(Voltaj) denkleminden 35mW'tır.

AVR mimarisine sahip Atmega 328p için aynı test uygulaması (ADC to PWM) ve aynı koşullarda (8MHZ osilatör frenaksı ve oda sıcaklığı) mikro denetleyicinin çektiği akım 17mA ve besleme voltajı 5 volttur. Yine P=IxV denkleminden tüketilen güç 85mW olarak ölçülmüştür.

FPGA üzerinde tasarlanan mikro denetleyici de ise yine aynı test uygulamasıyla test edilmiştir. Ancak burada bazı farklılıklar söz konusudur. FPGA üzerinde analog bir tasarım yapılamadığından tasarlanan mikro denetleyici henüz ADC birimine sahip değildir. Bunun için harici bir ADC entegresi kullanılıp potansiyometreden okunan değer, SPI(Serial Peripheral Interface Bus) haberleşmesiyle entegreden okunup mikro denetleyiciye gönderilmiştir.

Bu test kurulumunda diğer iki mikro denetleyicide olduğu gibi kullanılan çevresel birimlerin (Timer, Port,CCP(Compare/Capture/PWM),ADC,Prescaler) yanı sıra SPI çevresel birimi de kullanılarak ölçüm sağlanmıştır. Bu şartlar altında ve 10MHZ

osilatör frekansıyla mikro denetleyicinin çektiği akım (ADC entegresinin akımıda eklenerek) 10mA ve besleme gerilimi 3.3V - 5V olarak ölçülüp tüketilen güç 33mW-50mW arasındadır. Elbette göz önünde bulundurulmalıdır ki mikro denetleyicinin çalışma frekansıyla tükettiği güç doğru orantılıdır ve bunun ışığında her bir mikro denetleyicinin yine aynı test kurulumu sağlanarak ve aynı test uygulamasıyla 20MHZ çalışma frekansında ölçülen değerler şöyledir :

Atmega 328P mikro denetleyicisinin 20MHZ aktif modda çektiği akım 22mA besleme gerilimi 5.5V ve gücü ise 110mW'tır.

FPGA üzerindeki mikro denetleyicinin 20MHZ aktif modda çektiği akım 15mA besleme gerilimi 5V ve gücü ise 65mW'tır.

PIC18F4520 mikro denetleyicisinin 20MHZ aktif modda çektiği akım 12mA besleme gerilimi 5 V ve gücü ise 60mW'tır.

Bu verilerin ışığında güç baz alınarak 8MHZ ve 20MHZ osilatör frekansında testleri yapılan üç mikro denetleyicinin tükettikleri güç bakımından FPGA üzerinde tasarlanan mikro denetleyici Atmega 328P için çok daha verimli ancak PIC18F4520 için neredeyse aynıdır. Bu karşılaştırmanın bir diğer parametresi olan hız(transistor teknolojisinden kaynaklanan hız her bir mikro denetleyicide farklı olacağından komut işleme zamanı baz alınmıştır) olarak değerlendirildiğinde PIC18F4520 tasarımı ve güç optimizasyonu amacıyla bir komut dört farklı saat darbesinde (Osilatör darbesi) işlenmektedir. Yani 20MHZ hızında çalıştığını varsayarak bir komutu işlemesi için, her bir saat darbesinin gelmesi için geçen süre T(periyot)=1/f(frekans) denkleminden 0.5 mikro saniye ve dört adet saat darbesinin gelmesi 2 mikro saniye zaman almaktadır. Ancak FPGA üzerinde tasarlanan işlemci her bir tek bir saat darbesinde(birden fazla bit kaydırma komutu dahil) gerçekleştirmektedir.

Bu noktada FPGA üzerinde tasarlanan mikro denetleyici komut işleme anlamında 4 kat daha hızlıdır. Diğer yandan AVR mimarisine sahip Atmega 328P FPGA

üzerinde tasarlanan mikro denetleyiciye paralel olarak her bir komut bir saat darbesinde gerçekleşmekte ancak maksimum 20MHZ çalışma frekansına sahiptir. FPGA üzerinde tasarlanan mikro denetleyici ise 50MHZ frekansta çalışabilmektedir.

Bunun yanı sıra Atmega ve PIC mikro denetleyicileri sırasıyla sekiz ve on altı bitlik portlara sahip iken FPGA üzerinde tasarlanan mikro denetleyici 32 bitlik portlara sahiptir. Bu durum mikro denetleyicinin daha çok kontrol portuna sahip olmasına daha büyük sayılarla çalışabilmesine olanak sağlar. Ayrıca komut seti olarak değerlendirildiğinde 32 bitlik komut setine sahiptir. Buna paralel olarak yazmaçlar da (register) 32 bit ve 32 tanedir. Bunun anlamı herhangi bir yazmada tek seferde 16bit değerinde veriler (maksimum desimal 65536 sayısı) yazılabilmektedir. Atmega ve PIC mikro denetleyicilerinde ise 8 veya 16 bitlik 8 veya 16 tane yazmaç bulunur

Yani daha çok kontrol portuna sahip olduğu için paralel işlem yapma kabiliyeti daha yüksek olması IOT(Internet Of Things) uygulamalarda işlevselliğini arttırmaktadır.

Bütün bunların ışığında iki mikro denetleyiciden daha avantajlı ve daha spesifik alanlarda tercih edilebilecek bir mikro denetleyici tasarlanmış ve üretimi için RTL tasarımları tamamlanmıştır.

Ayrıca bu iki mikro denetleyiciye kıyasla büyük avantajlara sahip olmasının yanında daha kompleks alanlarda kullanılabilecek bir mikro işlemci tasarımı yapılmıştır. Bu sayede oldukça geniş bir yelpaze de çok çeşitli uygulama alanlarında, büyük problemlerin çözümü olabilecek niteliklere sahiptir. Bu bağlamda bu tasarımın üretilmesi ve ülkemize kazandırılması hedeflenmektedir. Bu sayede ilgili alanlarda

KAYNAKLAR

BAIS, K. And ALİ, Z.,2016. Comparison of Various Adder Designs in terms of Delay and Area. International Journal of Science and Research (IJSR).5:1292-1295.

BAKAOGLU, H.B., GROHOSKI, G.F. and MONTOYE, R.K.,1990. The IBM RISC System/6000 processor: Hardware overview. IBM Journal of Research and Development.1:12-22.

BEELITZ, H., MIILLER, H., LINDHARDT, R. And SIDNAM, R., 1967. Partitioning for large-scale integration. IEEE International Solid-State Circuits Conference. Digest of Technical Papers. :50-51.

COHEN, E. And SOIFFER, N.,1980. CS 292R Final Reports. University of California at Berkeley.101-140.

DAVID, A. And DAVID, R., 1980. The case for the reduced instruction set computer. SIGARCH Comput. Archit.6-8.

DAVIS, S.,1974. A fresh view of microcomputers. Computer Design. :67-78

MARTIN, A.J.,1997. The design of an asynchronous MIPS R3000 microprocessor. Proceedings Seventeenth Conference on Advanced Research in VLSI. 164-181

NOVKOVIC, T., LUKAC, Z., JAVONOVIC,P., And KASTELAN I.,2020. Graphic Library Optimization for MIPS Architecture. Elektronika ir Elektrotechnika.2:69-76.

OMRAN, S. and JUMMA, L.,2017. Design of Multithreading SHA-1 & SHA-2 MIPS Processor Using FPGA. International Conference on Information Technology.632-637.

PIERSON, D. DROBAC, S. And PARRY, D.,1992. An 80 Mhz Mips R6000 Cpu Using Multi Chip Module Technology. Proceedings of the IEEE Custom Integrated Circuits Conference. 23.7.1-23.7.4

SACK, E.A., LYMAN, R.C. and CHANG, G.Y., 1964. Evolution of the Concept of a Computer on a Slice. IEEE,12:1713-1720.

SEIM, T.A., 1975. Automation of a Brazing Process with an Intel 8008 Microprocessor. IEEE Transactions on Industrial Electronics and Control Instrumentation.3:303-307.

SEIM, T.A.,1973. A Microprocessor Sampled Data Process Controller. Richland:Battelle-Northwest

YEAGER, K.C.,1996. The Mips R10000 superscalar microprocessor. IEEE Micro.2:28-41

URL-1:https://en.wikipedia.org/wiki/MIPS_Technologies Erişim Tarihi: 10.09.2020

URL-2:https://makecode.microbit.org/courses/logic-lab/logic-gates Erişim Tarihi: 10.09.2020

URL-3:https://microcontrollerslab.com/cd4008-4-bit-full-adder-ic-pinout-features-example-datasheet/ Erişim Tarihi: 10.09.2020

URL-4:https://www.electronicshub.org/binary-multiplication/ Erişim Tarihi: 10.09.2020

URL-5:https://www.electronics-tutorial.net/combinational-logic-circuits/binaryadder-subtractor/ Erişim Tarihi: 10.09.2020

URL-6:https://www.electronics-tutorials.ws/logic/logic_2.html Erişim Tarihi: 10.09.2020

URL-7:https://binaryterms.com/risc-processor.html Erişim Tarihi: 10.11.2020

URL-8:https://binaryterms.com/cisc-processors.html Erişim Tarihi: 10.11.2020

URL-9:https://www.intel.com/content/www/us/en/programmable/products/boards_and_kits/dev-kits/partners/max-10-neek.html Erişim Tarihi 10.11.220

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Sovadı **:** Yusuf EREN

EĞİTİM

Derece		**Adı, İlçe, İl**	**Bitirme Yılı**
Lise	:	GAP Anadolu Lisesi	2014
Üniversite	:	Harran Üniversitesi Elektrik Elektronik Mühendisliği	2018
Yüksek Lisans	:	Harran Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik Mühendisliği Ana Bilim Dalı,	2021

TEKNİK BİLGİLER

Bilgisayar ve Yazılım Bilgileri

C/C++, Verilog, Python, Altium PCB, Matlab, PICC, AutoCAD, DSPic,

Elektrik Elektronik Bilgileri

Baskı devre kartı tasarımı (PCB)
Tümleşik devre tasarımı
Derin Öğrenme
Makine Öğrenmesi
Görüntü işleme
PLC Scada
FPGA tabanlı DSP uygulamaları
Elektrik makineleri ve tesisleri

YABANCI DİLLER

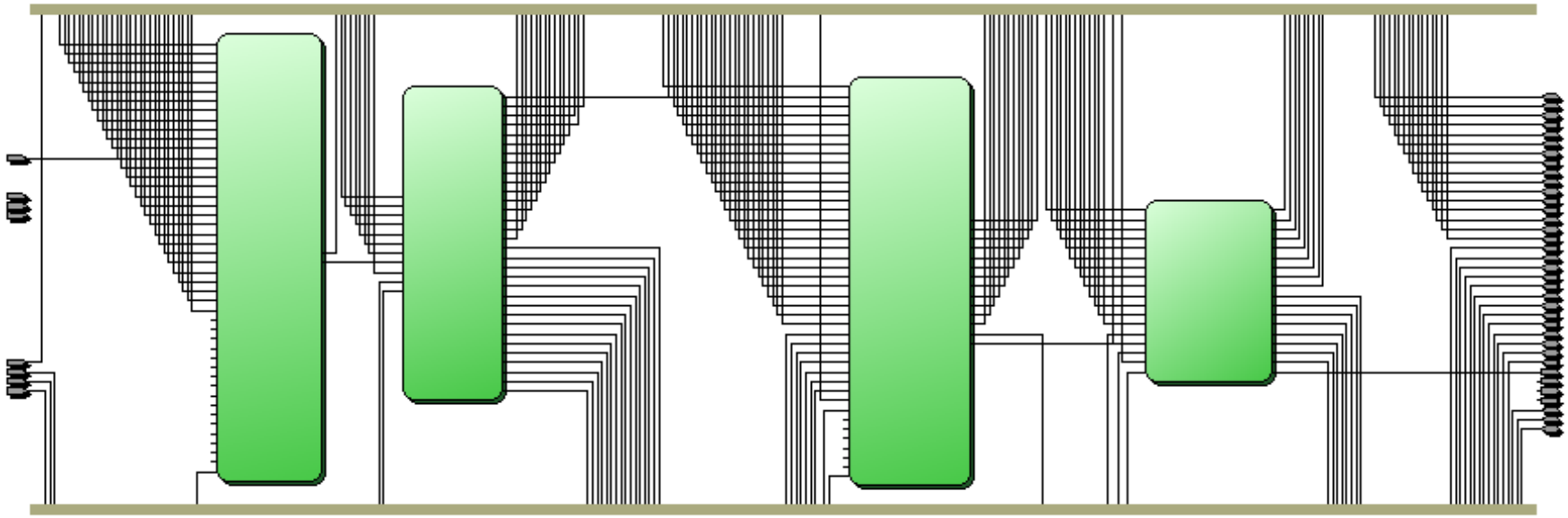
İngilizce (Orta)

YAYINLAR

EREN, Y., ve GÖKTAŞ,H.,2020. FPGA tabanlı multi-clock cycle, piplined MIPS-32 mimarisinde mikroişlemci tasarımı. 5. International Conference on Material Science and Technology (IMSTEC 2020).5:23

EKLER

Ek-1 Çevresel Birimlerin RTL gösterimi



Ek-2 CPU Mimarisi RTL gösterimi

