metin, grafik tasarım, poster, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

**ÇİP TASARIM YARIŞMASI MİKRODENETLEYİCİ TASARIM KATEGORİSİ**

**ÖN TASARIM RAPORU**

**TAKIM ADI: ArchOzu**

**BAŞVURU ID: 2007098**

**2024**

İçindekiler

[1. Giriş 2](#_Toc161341433)

[2. Sistem Mimarisi 3](#_Toc161341434)

[2.1 İşlemci Çekirdeği 3](#_Toc161341435)

[2.2 Çevre Birimleri 4](#_Toc161341436)

[2.3 Sistemin Doğrulanması 5](#_Toc161341437)

[3. Tasarım Detayları 5](#_Toc161341438)

[3.1 İşlemci Çekirdeği 6](#_Toc161341439)

[3.1.1 Getir: Bellekten İşlemciye 6](#_Toc161341440)

[3.1.2 Çöz: Anlamlandırma 6](#_Toc161341441)

[3.1.3 Yürüt: Sonucu bul ve kaydet 6](#_Toc161341442)

[3.1.4 Geri Yaz: LOAD Sonucunu Kaydet 7](#_Toc161341443)

[3.2 Doğrulama 7](#_Toc161341444)

[3.2.1 UVM 7](#_Toc161341445)

[4. Takım Organizasyonu ve İş Planı 7](#_Toc161341446)

[5. Kaynakça 8](#_Toc161341447)

# Giriş

Bu proje; açık-kaynak CV32E40P RISC-V çekirdeğine aşağıda belirtilen çevre birimlerini ekleyerek bir Mikrodenetleyici yapmayı amaçlamaktadır:

✔ UART

✔ I2C Master

✔ QSPI Master

✔ Timer

✔ GPIO

✔ USB Full-Speed Device (12 Mbps)

✔ JTAG

Bu çevre birimlerinin; takım üyeleri tarafından yazılması, açık-kaynak olarak alınmasına tercih edilecektir. Çevre birimlerinin her biri tarafımızca kapsamlı testlere tutulacaktır. Elde edilen Mikrodenetleyicinin FPGA implementasyonunun yapılması ve Fiziksel Tasarım çıktılarının elde edilmesi hedeflenmektedir.

Sistem Mimarisi bölümünde, mikrodenetleyicinin tanımından ve içerdiği birimlerden bahsedilmektedir. Ayrıca, sistemin doğrulamasının nasıl olacağından da bahsedilmektedir.

Tasarım Detayları bölümünde, mikrodenetleyicinin tasarımı ve doğrulanması detaylıca anlatılmıştır.

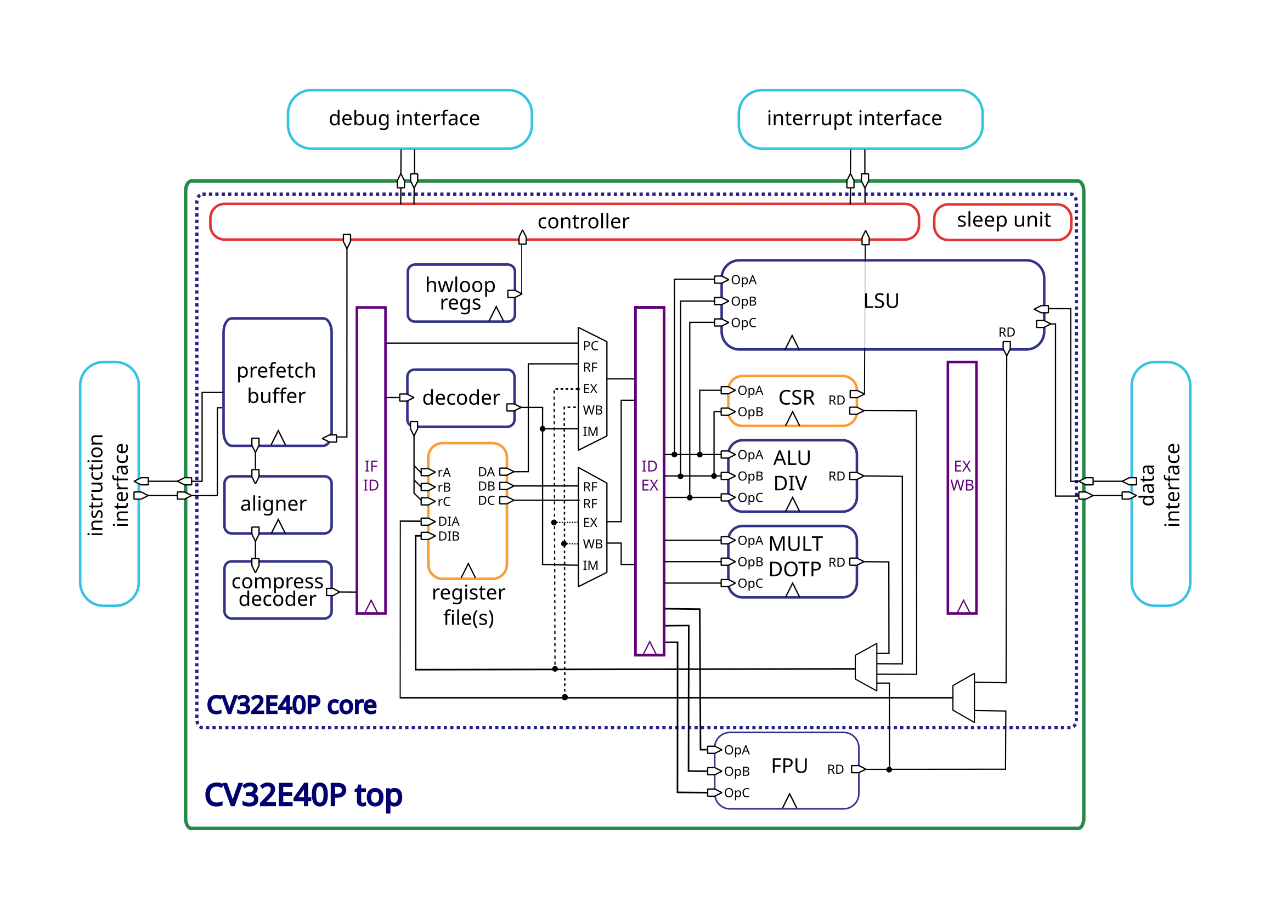
Takım Organizasyonu ve İş Planı bölümünde, öngördüğümüz iş planını detaylıca bulabilirsiniz. Planın seyir akışını bozacağını düşündüğümüz bir sorunla karşılaşırsak planda küçük çapta değişikliğe gidebiliriz.

Kaynakça ve Ekler bölümünde, raporla ilgili merak ettiğiniz kısımlar hakkında daha detaylı bilgi edinmek için ilgili adresleri IEEE formatında bulabilirsiniz.

# Sistem Mimarisi

Mikrodenetleyiciler analog ve dijital giriş/çıkışlara sahip olan ve içerisinde mikroişlemci, bellek vb. birçok birim bulunduran elektronik cihazlardır.

# İşlemci Çekirdeği

Bu projede, OpenHW Group tarafından tasarlanan CV32E40P işlemci çekirdeğini kullanmaktayız. Bu işlemci çekirdeği, 32-bit RISC-V standardında olup birçok uzantıyı desteklemektedir. İçerisinde bulunan boruhattı 4 aşamalıdır. Mikrodenetleyicilerin öngörülebilir olması istenildiği için boruhattında komutları sırasıyla yürütür.

Şekil 1 İşlemci Çekirdeği Blok Diyagramı [1]

# Çevre Birimleri

Çevre birimleri, Mikrodenetleyicinin dış dünya ile iletişim kurabilmesini sağlar.

RESİM KOY

UART, evrensel asenkron alıcı/verici demektir.

I2C Master, …

QSPI Master, …

Timer, …

GPIO, …

USB, …

JTAG, …

# Sistemin Doğrulanması

Sistemin doğrulanması, Donanım Tasarımında en çok dikkat gerektiren kısımlardan biridir. İşlemci çekirdeği, OpenHW Group tarafından test edildiği için ayrıca test edilmeyecektir. UART modülü UVM standardında, geri kalanlar ise genel Verilog test ortamlarında test edilecektir. En son sistem genel olarak test edilecektir.

# Tasarım Detayları

* Şartnamede belirtilen mikrodenetleyici tasarımının alt blokları (Bus yapısı, bellek yapısı, boot yapısı, çevre birimler, USB device birimi vb.) belirtilmeli, blok diyagramı çizilmeli ve açıklanmalıdır.
* RISC-V çekirdeğin doğrulaması hakkında bilgi verilmelidir.
* Mikrodenetleyici donanım tasarımı bittikten sonra, hazırlanan yazılım uygulamalarının (C/C++) mikrodenetleyici üzerinde nasıl çalıştırılacağı hakkında bilgi verilmelidir.
* Çip tasarım akışının her aşamasına kısaca değinilmeli, RTL tasarımından PnR (Place and Route) kısmına kadar öngörülen aşamalar kısa ve anlaşılır ifadeler ile açıklanmalı ve nasıl gerçekleştirileceği belirtilmelidir (açık-kaynak araçlar (Örn. Openlane) üzerinden veya ticari kaynaklar üzerinden (Örn. Synopsys, Cadence) örnekler verilerek anlatılabilir). Bu noktada sadece akış hakkında bilgi verilmesi istenilmiştir.

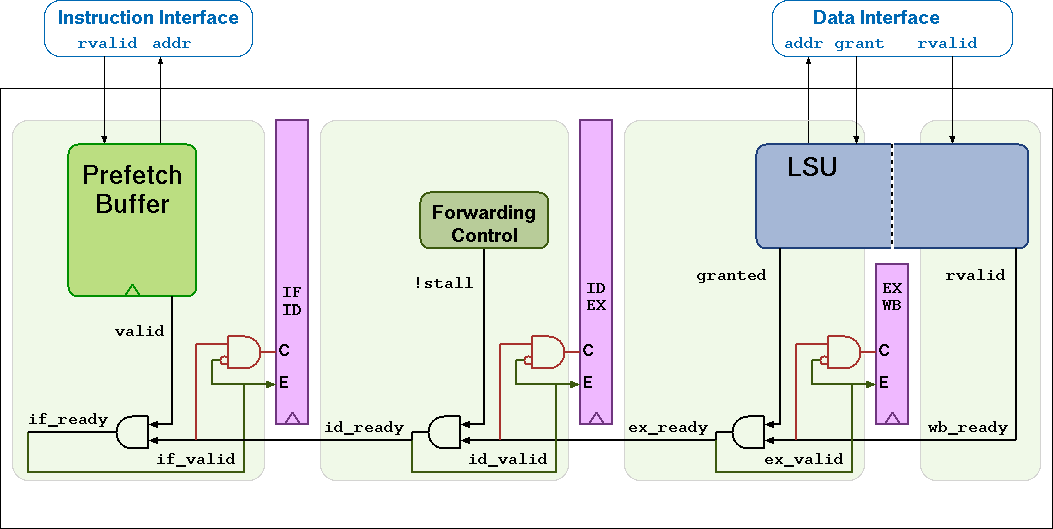
# İşlemci Çekirdeği

CV32E40P, temel komutlara ek olarak başka komutları da desteklemektedir. Bunlardan bazıları:

* Sıkıştırılmış
* Çarpma ve bölme
* Performans Sayaçları
* Virgüllü sayı işlemleri

CV32E40P, 4 aşamaya sahiptir. Sırasıyla:

1. Getir (IF)
2. Çöz (ID)
3. Yürüt (EX)
4. Geri Yaz (WB)



Şekil 2 Boruhattı [2]

# Getir: Bellekten İşlemciye

Komut belleğinden, oldukça küçük büyüklükteki İGİÇ (İng. FIFO) prensibine sahip arabelleğe (İng. Buffer) komut hizalı bir şekilde getirilir. Daha sonra getirilen bu komut, arabellekten Çöz aşamasına aktarılır.

# Çöz: Anlamlandırma

Bu aşamada komut, kategorize edilir. Bu kategorizasyon sırasında, ilgili komutun türü, hangi birimlere ihtiyacı olduğu, hangi yazmaçları okumak/yazmak istediği vb. kriterler belirlenir. Böylece benzer komutların aynı alt görevleri aynı devrelerden farklı parametrelerle işlenebilir. Devrelerin birden fazla komut için kullanılabilir olması alan tasarrufu, yerleştirme kolaylığı, maliyet vb. birçok konuda olumludur.

# Yürüt: Sonucu bul ve kaydet

Aritmetik işlemler, virgüllü sayı işlemleri ve daha birçok işlem hesaplanır. LOAD dışındaki komutların sonucu bu aşamada kaydedilir. Dallanma koşullarının sağlanıp sağlanmadığına bakılır. Bazı komutlar 1’den fazla döngü (İng. Cycle) gerektirebilir. Böyle durumlarda, boruhattı bloke edilir.

# Geri Yaz: LOAD Sonucunu Kaydet

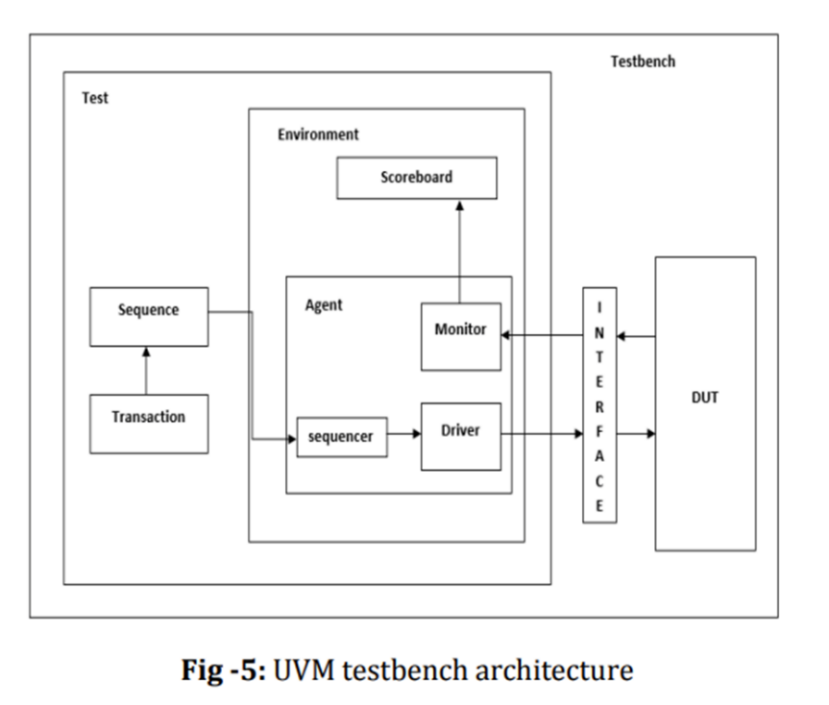
Veri belleğinden veri çekilmesini sağlayan LOAD komutunun sonucu kaydedilir. İstenirse virgüllü sayı komutlarının sonucu da bu aşamada kaydedilebilir.

# Doğrulama

Düzelt beni

# UVM

UVM veya Universal Verification Methodology, yarıiletken endüstrisinde kullanılan dijital tasarımların doğrulaması için standartlaştırılmış bir metodolojidir. Bu, modüler, ölçeklenebilir ve tekrar kullanılabilir doğrulama ortamları oluşturmak için bir çerçeve ve yönergeler sağlar. UVM, modern dijital tasarımların karmaşıklığını yönetmede ve doğrulama sürecini kolaylaştırmada etkinliği nedeniyle geniş bir şekilde benimsenmiştir.



Şekil 3 UVM Test Ortamı Mimarisi [3]

# Takım Organizasyonu ve İş Planı

Takımımız, danışmanımız da dahil olmak üzere toplamda 5 kişiden oluşmaktadır. Takımdaki herkes, Özyeğin Üniversitesi’nde okumaktadır. Danışmanımız Cengiz Emre Dedeağaç, Bilgisayar Mühendisliği bölümünde doktora yapmaktadır. Metin Arda Köker, Bilgisayar Mühendisliği bölümünde; Ahmet Utku Erşahin ise Elektrik-Elektronik Mühendisliği bölümünde yüksek lisans yapmaktadır. Kutay Bulun ve Taha Gemici, Elektrik ve Elektronik Mühendisliği bölümünde lisans öğrencileridir.

Ahmet Utku Erşahin; UART ve JTAG arayüzlerinin RTL tasarımlarından sorumludur.

Metin Arda Köker; USB arayüzünün RTL tasarımından sorumludur. Projenin Fiziksel Tasarım kısmını ikili beraber yapacaktır.

Taha Gemici; geriye kalan arayüzlerin RTL tasarımlarının yapılmasından, tüm sistemin FPGA üzerindeki implementasyonundan ve raporların hazırlanmasından sorumludur.

Kutay Bulun, RTL tasarımların doğrulanmasından sorumludur. İlk başta UART için UVM standardına uygun test modülü oluşturacaktır. Devamında, diğer arayüzler için test modülleri geliştirecektir.

CHART KOY

# Kaynakça

|  |  |
| --- | --- |
| [1] | «CORE-V CV32E40P User Manual - Introduction,» OpenHW, 2023. [Çevrimiçi]. Available: https://docs.openhwgroup.org/projects/cv32e40p-user-manual/en/latest/intro.html. [Erişildi: 12 Mart 2024]. |
| [2] | «CORE-V CV32E40P User Manual - Pipeline Details,» OpenHW, 2023. [Çevrimiçi]. Available: https://docs.openhwgroup.org/projects/cv32e40p-user-manual/en/latest/pipeline.html. [Erişildi: 12 Mart 2023]. |
| [3] | M. K. N. Vivekananda T, «Design and Verification of the UART and SPI protocol using UVM,» 2022. [Çevrimiçi]. Available: https://www.irjet.net/archives/V9/i9/IRJET-V9I918.pdf. |