



**TOBB Ekonomi ve Teknoloji
Üniversitesi**
Bilgisayar Mühendisliği Bölümü
BİL 361 – Bilgisayar Mimarisi ve Organizasyonu

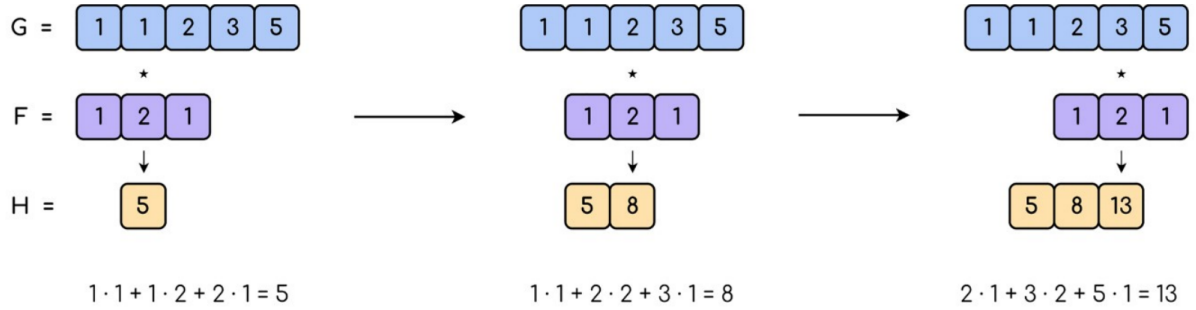
31 Ocak 2022
2021 – 2022 Öğretim Yılı
Bahar Dönemi
Ödev 1

Güncelleme - 01/02/2022

[100 puan] RISC-V İşlemci Tasarımı

Bu ödevde 1 boyutlu evrişim işlemlerini gerçekleştirebilecek RISC-V buyruk kümesi mimarisindeki buyruklarını yürütebilen tek vuruşlu 32-bit işlemciyi tasarlayacaksınız.

Aşağıda 1x5 boyutunda giriş (G) ve 1x3 boyutunda filtre (F) matrisinin 1 boyutlu evrişim işlem örneği ve sonucunda oluşan çıkış (H) matrisi bulunmaktadır.



Tasarlayacağınız işlemcide giriş ve filtre matrislerinin veri belleğinde bulunduğunu ve evrişim işlemlerinin döngüsel bir şekilde gerçekleştirileceğini varsayın. Buna göre aşağıdaki üç soruyu cevaplayın.

1. Evrişim işlemlerini gerçekleştirmek için gerekli olan RISC-V buyruklarını listeleyin. Listediğiniz her bir buyruğun evrişim işleminde ne için kullanıldığını açıklayın.
2. Listediğiniz buyrukları kullanarak verilen girdilere ve kısıtlara göre evrişim işlemini gerçekleştiren RISC-V BKM tabanlı Assembly kodunu yazın.
 $1 \times (\#G_BOYUT)$ 'luk G matrisinin veri belleğinin on altılık tabanda $0x(\#G_BASLANGIC)$ adresinden başladığını, $1 \times (\#F_BOYUT)$ 'luk F matrisinin de $0x(\#F_BASLANGIC)$ adresinden başladığını varsayın. $1 \times (\#H_BOYUT)$ 'luk sonuç matrisinin elemanlarını sırasıyla $0x(\#H_BASLANGIC)$ adresinden başlayarak belleğe yazın.
Varsayım: Bellekte bulunan üç ayrı matrisin adreslerinin hiçbir zaman çakışmadığını varsayabilirsiniz.
Kodunuzu yazarsan anlık değerleri ($\#G_BOYUT$, $G_BASLANGIC$, F_BOYUT , $F_BASLANGIC$, $H_BASLANGIC$) olarak verilen parametreleri yazın.
3. 1. soruda listediğiniz RISC-V buyruklarını yürütebilen tek vuruşlu 32-bit işlemciyi Verilog dilinde tasarlayın. İşlemcinizin tasarımını barındıran en üst düzey dosyayı "islemci.v" olarak adlandırın. (İstediğiniz kadar alt modül kullanabilirsiniz.)
islemci modülünüzün giriş çıkışları aşağıdaki gibidir.
 - **saat:** 1-bitlik saat girişi
 - **reset:** 1-bitlik devrenizi başlangıç durumuna geri çeviren sinyal girişi.
 - **buyruk:** İşlemcinin şu anki saat vuruşunda yürüteceği buyruk
 - **y_on:** Yazmaç öbeğindeki on numaralı yazmacın değeri

İşlemciniz her saat vuruşunda buyruk sinyali ile gelen buyruğu çözmeli, yazmaç öbeğini, veri belleğini ve program sayacını yapılacak hesaplamaya göre güncellemelidir. Devreniz, resetlendikten sonraki ilk çevrim on altılık tabanda “0000_0000” program sayacı değerinden ilk buyruğu getirmelidir. Bir diğer deyişle işlemciniz çalışmaya başladığında buyruk belleğindeki 0. buyruğu alarak programı çalıştırmaya başlayacak. İşlemcinizin içinde bulunduğu sistem bayt adresleme kullanmaktadır. Program sayacı değerini doğru bir şekilde güncellemeyi unutmayın. Yazdığınız tasarımın doğruluğunu ikinci sorudaki işlemi gerçekleştirerek test edebilirsiniz.

4. **[Çalışma Sorusu - notlandırılmayacak.]** 2.sorudaki işlemlerde F_BASLANGIC adresinde bulunan F matrisinin ilk elemanının (F[0]) en anlamlı 16-biti, bu adresin en anlamsız 16-bitinde; F[0]’ın en anlamsız 16-bitinin F_BASLANGIC+4 adresindeki en anlamlı 16-bitinde tutulduğunu ve diğer elemanların da bu şekilde saklandığı bir durumda evrişim işlemini gerçekleştiren RISC-V assembly kodunu yazın.

Örnek: İlk üç elemanın veri belleğinde nasıl tutulduğunu gösteren tablo

| DEĞER | - | F[0]’ın En anlamlı 16-biti | F[0]’ın En anlamsız 16-biti | F[1]’ın En anlamlı 16-biti | F[1]’ın En anlamsız 16-biti | F[2]’ın En anlamlı 16-biti |
|-------|-------------|----------------------------|-----------------------------|----------------------------|-----------------------------|----------------------------|
| ADRES | F_BASLANGIC | | F_BASLANGIC + 4 | | F_BASLANGIC + 8 | |

İşlemcinizin çalıştıracağı buyruklar RISC-V buyruk kümesi mimarisi tanımlamasındaki gibi kodlanacaktır. Buyrukları doğru bir şekilde çözmek için [bu dokümanı](https://github.com/riscv/riscv-isa-manual/releases/download/Ratified-IMAFDQC/riscv-spec-20191213.pdf) (https://github.com/riscv/riscv-isa-manual/releases/download/Ratified-IMAFDQC/riscv-spec-20191213.pdf) referans almalısınız. Kural dışı durumları yok sayabilirsiniz

Ödev Gönderimi ve Formatı

Ödevinizde aşağıdaki soruları cevapladığınız ve yazdığınız kodları açıkladığınız “odev1_raporu.pdf” isimli bir rapor, işlemci tasarımı için yazdığınız Verilog dosyaları ve Assembly dilinde yazılmış kodu içeren “asm_kod.txt” adlı kod dosyası bulunmalıdır. Ödevinizi uzak platformuna **sıkıştırmadan** yükleyeceksiniz.

Son Teslim Tarihi: 13 Şubat 2022, 23:59