



## BLG 322E – Computer Architecture Assignment 2

**Due Date:** 15.03.2017, **Wednesday**, 16.30.

### QUESTION:

A RISC CPU has an instruction pipeline with 5 stages:

F: Fetch instruction

D: Decode instruction

O: Operand (register) read

A: ALU operation, and write the result to the register

M: Memory access if necessary (from register to memory or from memory to register)

...		
ADD	R6, R7, R8	; $R8 \leftarrow R6 + R7$
LD	O(R8), R1	; $R1 \leftarrow M[R8]$
LD	O(R11), R2	; $R2 \leftarrow M[R11]$
ADD	R1, R2, R3	; $R3 \leftarrow R1 + R2$
ADD	R0, R3, R4	; $R4 \leftarrow R3$
ST	O(R12), R3	; $M[R12] \leftarrow R3$
BA	L2	; Branch Always
ADD	R0, 0, R4	; $R4 \leftarrow 0$
...		
...		
L2:	ADD R0, 0, R3	; $R3 \leftarrow 0$

- Draw the space-time diagram for the given program, if it runs on a CPU with the pipeline given above. Solve all data and branch conflicts using NOP instructions. What is the total amount of penalty in clock cycles caused by conflicts for the given piece of code?
- To minimize the amount of penalty, apply the optimized software-based solutions to the conflicts, if it is possible. Remember; the results generated by the program cannot be changed. What is the total amount of penalty in clock cycles

with the new solutions?

### INSTRUCTION SET:

LD	X(Rs), Rd	$Rd \leftarrow M[Rs + X]$	Load
ST	X(Rs), Rd	$M[Rs + X] \leftarrow Rd$	Store
ADD	Ri, Rj, Rd	$Rd \leftarrow Ri + Rj$	
BA	Y	$PC \leftarrow PC + Y$	Branch Always (relative)

**Submission:** Use a single A4 paper to present your solution. Draw the diagrams using a computer program. If your solution is longer than a page, it means you are on the wrong way. You should type your name and student ID at the top of the paper. You must submit your homework through the Ninova system before the due date.

Late submissions are not accepted.

Assignments have to be made individually. If any plagiarism issue is detected, disciplinary regulations of the university are applied.

**Note:** If you have a problem about the homework, you may make contact with the research assistants of the course ([hakangunduz@itu.edu.tr](mailto:hakangunduz@itu.edu.tr), [muluyagmur@itu.edu.tr](mailto:muluyagmur@itu.edu.tr), [yildirimez@itu.edu.tr](mailto:yildirimez@itu.edu.tr)).



## BLG 322 – Bilgisayar Mimarisi Ödev 2

**Teslim Tarihi:** 15.03.2017, Çarşamba, 16.30

### SORU:

Aşağıda verilen program 5 katlı bir komut iş hattına (pipeline) sahip olan RISC işlemcisinde çalıştırılacaktır.

F: Komut alma

D: Komut çözme

O: Operand (saklayıcı) okuma

A: MİB işlemi ve MİB sonuçlarının saklayıcılara yazılması

M: Gerekli durumlarda bellek erişimi (saklayıcıdan belleğe veya bellekten saklayıcıya)

...		
ADD	R6, R7, R8	; $R8 \leftarrow R6 + R7$
LD	O(R8), R1	; $R1 \leftarrow M[R8]$
LD	O(R11), R2	; $R2 \leftarrow M[R11]$
ADD	R1, R2, R3	; $R3 \leftarrow R1 + R2$
ADD	R0, R3, R4	; $R4 \leftarrow R3$
ST	O(R12), R3	; $M[R12] \leftarrow R3$
BA	L2	; Branch Always
ADD	R0, 0, R4	; $R4 \leftarrow 0$
...		
...		
L2:	ADD	R0, 0, R3 ; $R3 \leftarrow 0$

saat çevrimi cezaya neden olmaktadır?

### KOMUT SETİ:

LD	X(Rs), Rd	$Rd \leftarrow M[Rs + X]$	Load
ST	X(Rs), Rd	$M[Rs + X] \leftarrow Rd$	Store
ADD	Ri, Rj, Rd	$Rd \leftarrow Ri + Rj$	
BA	Y	$PC \leftarrow PC + Y$	Branch Always (relative)

**Ödevin Teslimi:** Soruyu tek bir A4 kağıdına çözünüz. Diyagramları bilgisayarla anlaşılır biçimde çizin. Eğer çözümünüz bir sayfadan daha uzunsa bu yanlış yolda olduğunuz anlamına gelir. Adınızı ve öğrenci numaranızı kağıdın üst kısmına yazmalısınız. Ödevinizi teslim tarihinden önce Ninova sistemi aracılığıyla teslim etmelisiniz.

Geç teslim edilen ödevler kabul edilmeyecektir.

Ödevler tek kişiliktir. Kopya belirlenmesi durumunda kopyaya karışan tüm öğrenciler hakkında üniversitenin yönetmelikleri uyarınca disiplin işlemi uygulanır.

**Not:** Ödev hakkında bir sorunuz varsa dersin yardımcıları ile iletişime geçebilirsiniz ([hakangunduz@itu.edu.tr](mailto:hakangunduz@itu.edu.tr), [muluyagmur@itu.edu.tr](mailto:muluyagmur@itu.edu.tr), [yildirimez@itu.edu.tr](mailto:yildirimez@itu.edu.tr)).