

## بسمه تعالی

مهرماه 97

پروژه درس معماری

هدف: آشنایی با نحوه طراحی یک CPU ساده

می‌خواهیم یک CPU ساده 16 بیتی با مجموعه دستورات زیر را پیاده سازی کنیم:

Opcode	Mnemonic	Description	Function
0	LDA src	Load A, don't clear carry	$A := [src]$
1	STA dest	Store A	$[dest] := A$
2	XOR src	Exclusive or	$A := A \text{ xor } [src]$
3	SBC src	Subtract	$A := A - [src] - C$
4	ROR	Rotate right through carry	$A, C := A, C \text{ ror } 1$
5	HLT	Halt	CPU stops operation
6	JMP vec	Jump	$PC := vec$
7	ADC src	Addition with Carry	$A := A + [src] + C$
8	TAT	Transfer to T	$T := A$
9	LDCC src	Load A, clear carry	$A := [src]; C := 0$
A	IBCC vec	Indirect jump if carry clear	IF $C = 1$ THEN $PC := [vec]$
B	IBNE vec	Indirect jump if not zero	IF $Z = 0$ THEN $PC := [vec]$
C	LDI	Load A indirect	$A := [A]$
D	STTI	Store T indirect	$[A] := T$
E	LOR src	Logical or	$A := A \text{ or } [src]$
F	LAND src	Logical and	$A := A \text{ and } [src]$

کلید دستورات 16 بیتی است که 4 بیت اول (بیت‌های 15 تا 12) opcode و 12 بیت بعدی (بیت‌های 11 تا 0) آدرس می‌باشند.

CPU دارای حداقل سه ثبات و دو flag زیر می‌باشد:

ثبات A (Accumulator)

ثبات PC (Program Counter)

ثبات T (Temporary Register)

C (Carry Flag)

Z (Zero Flag)

در صورت نیاز می‌توان ثبات‌های دیگری (مانند DR، AR و ...) به CPU اضافه نمود، ولی در هر صورت کاربر به طور مستقیم فقط با سه ثبات فوق در ارتباط می‌باشد و قادر به دستکاری سایر ثبات‌ها نیست.

الف) برای این CPU، مسیر داده (data path) مورد نیاز را طراحی کنید. این مسیر داده باید شامل ALU، ثبات‌ها، سیستم گذرگاه مشترک و حافظه باشد.

ب) با توجه به مجموعه دستورات و ریزعمل‌های مورد نیاز برای هر اجرای هر دستور، واحد کنترل این CPU را طراحی کنید. (واحد کنترل می‌تواند سخت افزاری یا نرم افزاری باشد)

ج) با استفاده از مجموعه دستورات، برنامه‌ای بنویسید که با دریافت یک عدد 16 بیتی ( $n$ )، مجموع اعداد 1 تا  $n$  را محاسبه کند. این برنامه را اسمبل کرده و با CPU اجرا کنید.

توجه:

برای پروژه‌های مشابه در هر سطحی نمره صفر منظور خواهد شد.

موفق باشید

جراحی