به نام خدا

سیستم‌های دیجیتال ۲ ـ ریزپردازنده و زبان اسمبلی

پاسخ تمرین سری دوم

دانشگاه صنعتی خواجه نصیرالدین طوسی

مدرس: رسول دلیرروی فرد

پاسخ سؤال ۱

دستورات LDI هیچ پرچمی را تغییر نمی‌دهند؛ بنابراین فقط وضعیت پس از ADD، SUB، INC و ADC در جدول آمده است. (با توجه به عدم ذکر علامت‌دار بودن در صورت سؤال، اعداد در تحلیل‌ها مثبت و بدون علامت در نظر گرفته شده‌اند و بیت‌های علامت بررسی نمی‌شوند).

دستور	نتیجه	C	Z	N	V	S	H
ADD R16, R17	0x80	۰	۰	۰	۰	۰	۱
SUB R16, R17	0x7F	۰	۰	۰	۰	۰	۱
INC R18	0x00	۰ (بدون تغییر)	۱	۰	۰	۰	۱ (بدون تغییر)
ADC R18, R17	0x01	۰	۰	۰	۰	۰	۰

دلیل هر مورد

- **ADD R16, R17:** حاصل برابر $127 + 1 = 128$ (0x80) است. چون حاصل در بازه‌ی 0 تا 255 قرار دارد، رقم نقلی خروجی نداریم و $C = 0$ است. نتیجه غیرصفر است، پس $Z = 0$. در ۴ بیت پایین نیز حاصل $\mathtt{F} + 1 = 16$ خارج از بازه‌ی 0 تا 15 است (رقم نقلی نیم‌بایت تولید می‌شود)، پس $H = 1$ است.
- **SUB R16, R17:** حاصل برابر $128 - 1 = 127$ (0x7F) است. چون $\mathtt{0x80} \geq \mathtt{0x01}$، نیازی به قرض نیست و $C = 0$ است. نتیجه غیرصفر است، پس $Z = 0$. در ۴ بیت پایین نیز برای تفریق $0 - 1$ نیاز به قرض است (حاصل خارج از بازه‌ی 0 تا 15 است)، پس $H = 1$ است.
- **INC R18:** مقدار 255 (0xFF) با افزایش یک واحدی به 0 (0x00) می‌رود. این دستور روی C و H اثری ندارد؛ پس این دو مقدار قبلی خود (حاصل از SUB) را حفظ می‌کنند. چون نتیجه برابر صفر است، $Z = 1$ می‌شود.
- **ADC R18, R17:** این دستور $R18 \leftarrow R18 + R17 + C$ را محاسبه می‌کند. با توجه به $C = 0$ داریم $0 + 1 + 0 = 1$ (0x01). حاصل در بازه‌ی 0 تا 255 است ($C = 0$) و غیرصفر است ($Z = 0$). در ۴ بیت پایین نیز حاصل برابر 1 (بین 0 تا 15) است؛ پس $H = 0$ می‌شود.

نکته‌ی پایانی

مقدار نهایی R18 برابر 0x01 است و به پرچم C در لحظه‌ی اجرای ADC بستگی دارد. دستور INC روی C اثر ندارد؛ پس C همان مقداری است که آخرین دستور مؤثر بر آن، یعنی SUB، ساخته است. اگر INC در گذر از 255 (0xFF) به 0 (0x00) پرچم C را یک می‌کرد، جواب 0x02 می‌شد که اشتباه است.

پاسخ سؤال ۲

دستور	معتبر؟	پاسخ
LDI R12, 0x30	خیر	دستور LDI فقط روی R16 تا R31 کار می‌کند. شکل صحیح: `LDI R16, 0x30` و سپس `MOV R12, R16`
LDS R5, 0x0014	بله	آدرس 0x0014 در فایل رجیسترها (0x0000 تا 0x001F) است و محتوای R20 در R5 نوشته می‌شود
STS 0x10FF, R3	بله	SRAM داخلی (آخرین خانه‌ی بازه‌ی 0x0100 تا 0x10FF)
STS 0x2400, R17	بله	حافظه‌ی خارجی (آدرس از 0x1100 به بعد)
MULS R3, R25	خیر	هر دو رجیستر باید بین R16 تا R31 باشند. شکل صحیح: `MOV R16, R3` و سپس `MULS R16, R25`
SUBI R18, 300	خیر	ثابت k باید ۸ بیتی (۰ تا ۲۵۵) باشد. چون $300 = 256+44$، در رجیستر ۸ بیتی معادل آن `SUBI R18, 44` است
ADDI R20, 5	خیر	دستور جمع با ثابت وجود ندارد. شکل صحیح: `LDI R17, 5` و سپس `ADD R20, R17`
OUT 0x45, R16	خیر	آدرس OUT فقط بین 0x00 تا 0x3F مجاز است. شکل صحیح: `STS 0x0065, R16` (آدرس حافظه برابر آدرس I/O به‌علاوه‌ی 0x20) که در بخش Extended I/O قرار دارد
(Y = 0x0080) LD R20, Y	بله	آدرس 0x0080 در بخش Extended I/O (0x0060 تا 0x00FF) است
IN R7, 0x3F	بله	بخش SFR؛ رجیستر SREG (آدرس حافظه‌ی 0x005F) را می‌خواند

موفق باشید