Computer System Architecture

سازمان و طراحی یک کامپیوتر پایه

کدهای دستورالعمل ها

سازمان داخلی یک سیستم دیجیتال با رشته ریزعمل هایی که روی داده های ذخیره شده در ثباتهایش انجام می دهد تعریف می شود.

دستورالعمل کامپیوتر یک کد دودویی است که رشته ای از ریزاعمال را برای کامپیوتر مشخص می کند.

کدهای دستورات همراه با داده هایشان در حافظه ذخیره می شوند.

یک کد دستورالعمل مجموعه ای از بیت هاست که انجام یک عمل خاص را به کامپیوتر فرمان می دهد.

poorya.kh@gmail.com

کدهای دستورالعمل ها

کد دستورالعمل به دو قسمت تقسیم می شود:

- **کد عمل (اعمال مانند جمع، تفریق، ضرب، شیفت و ...)**

تعداد بیت های مربوط به عمل یک دستور به تعداد کل اعمال موجود درکامپیوتر بستگی دارد.

- **آدرس**

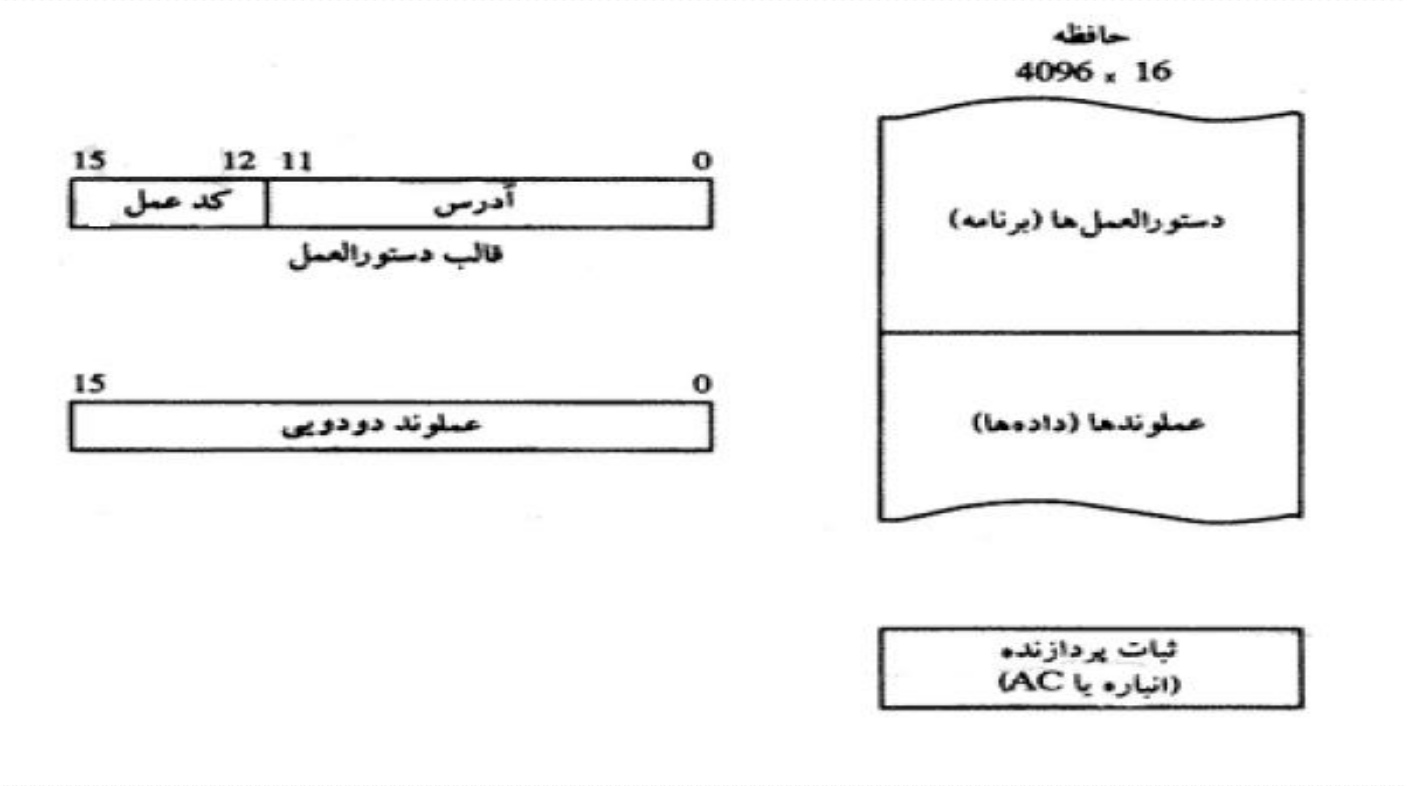
ثبات یا حافظه هایی که عملوندها در آنها می توانند یافت شوند و یا در آنها ذخیره می شود.

poorya.kh@gmail.com

کدهای دستورالعمل ها

$2^{12} = 4096$

$2^4 = 16$ ⟵ 16 عمل مختلف



بخش آدرس

بلافصل (Immediate)

در بخش آدرس از عملوند واقعی استفاده می شود

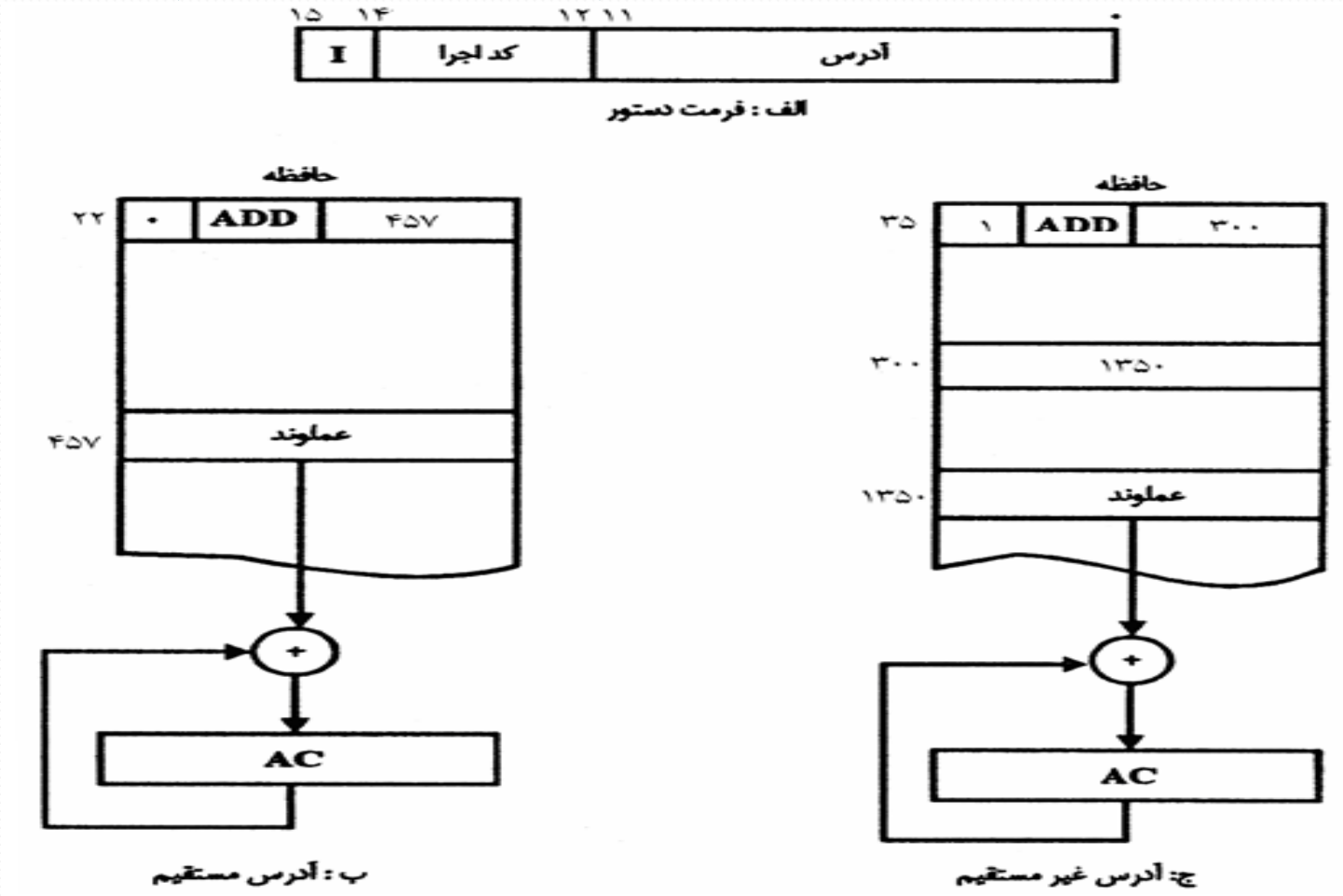
آدرس مستقیم (Direct Address)

در بخش آدرس، آدرس عملوند مشخص می شود.

آدرس غیرمستقیم (Indirect Address)

آدرس کلمه ای از حافظه را مشخص می کند که آدرس عملوند در آن قرار دارد

نمایش آدرس مستقیم و غیرمستقیم



شکل (۵-۲) نمایش آدرس مستقیم و غیرمستقیم برای دستور جمع ADD

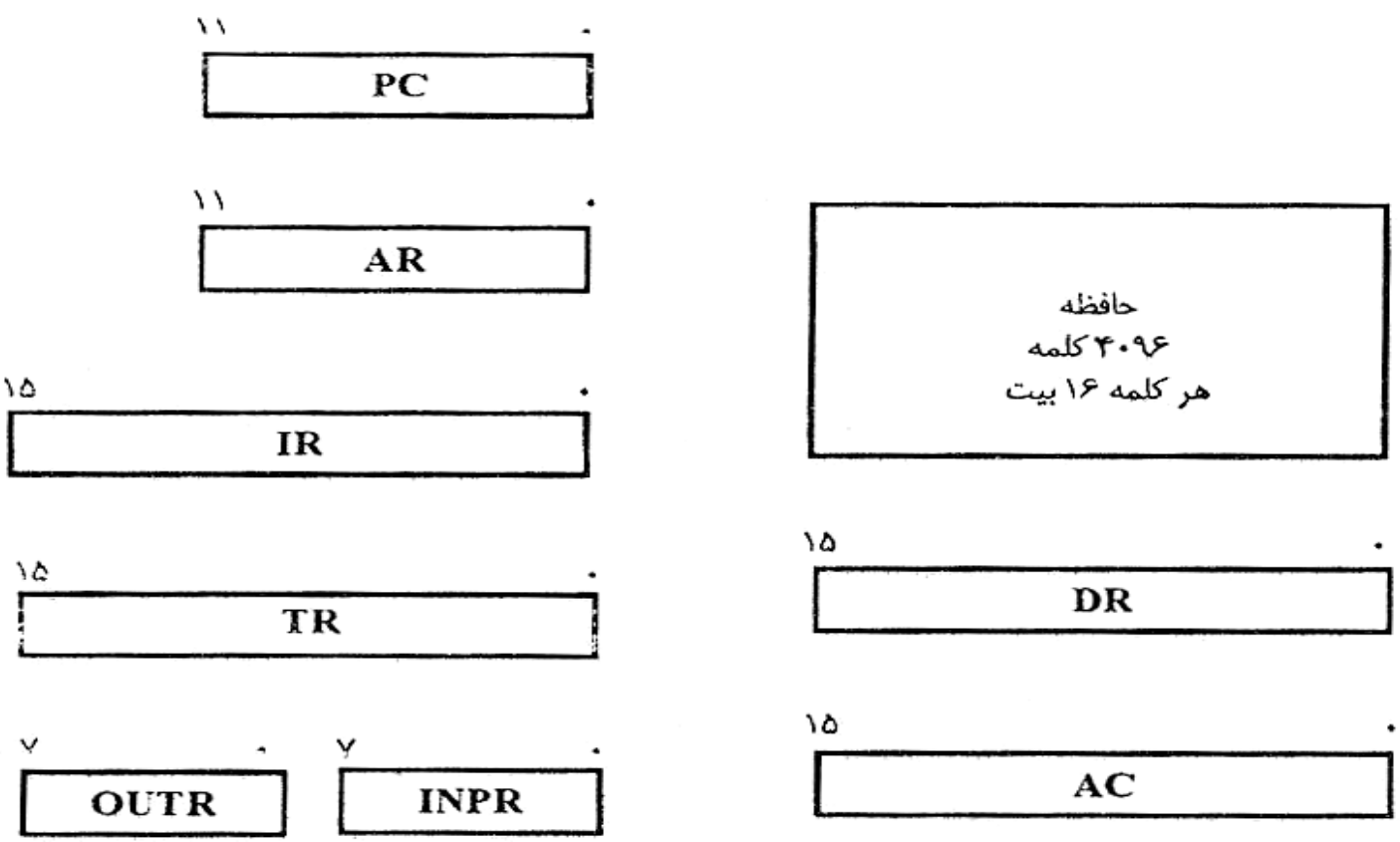
poorya.kh@gmail.com

ثبات ها و حافظه ها در کامپیوتر پایه

جدول (۵-۱) فهرست ثبات‌ها برای یک کامپیوتر مبنا

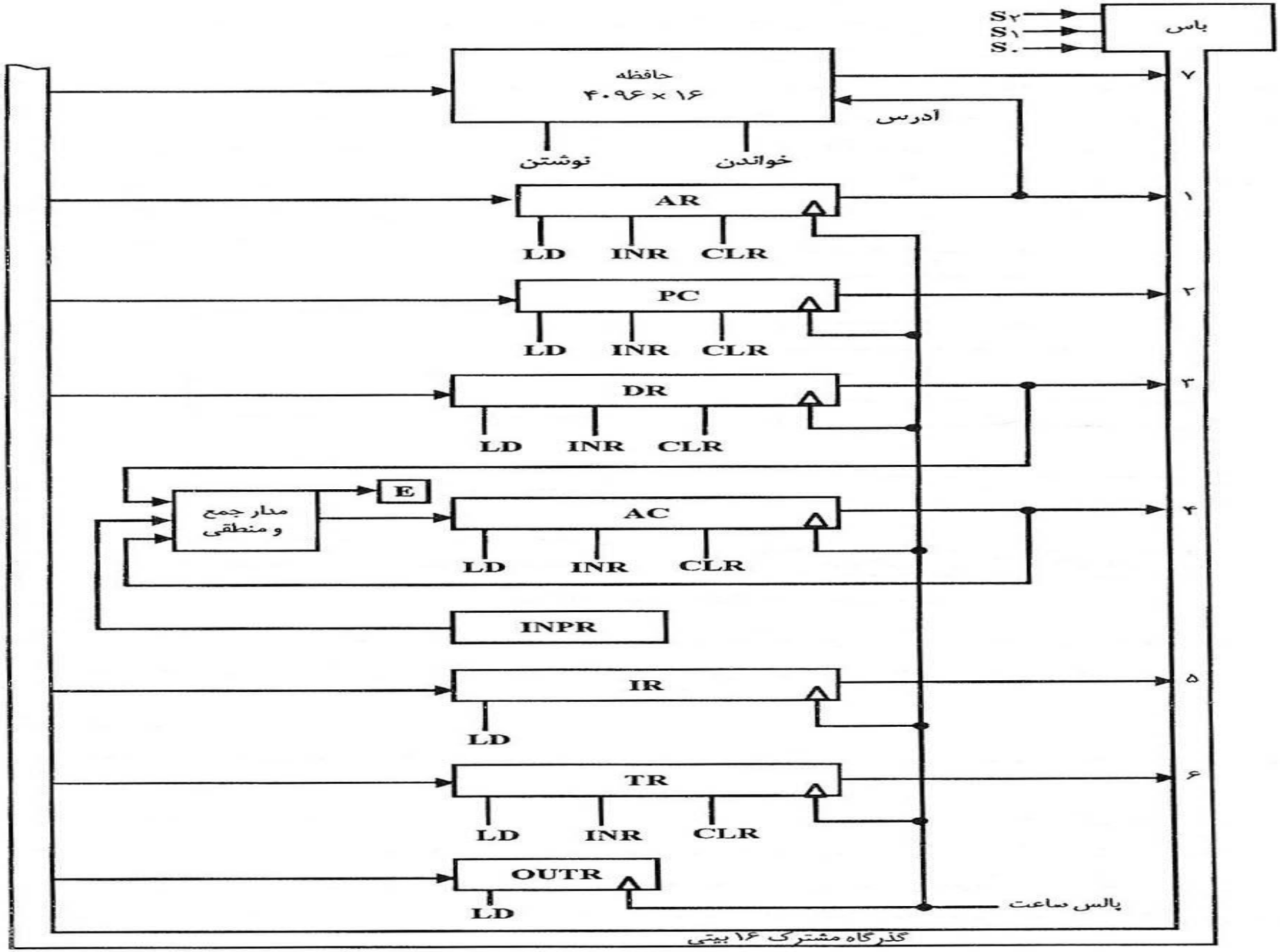
کار ثبات	نام ثبات	تعداد بیت	سمبول ثبات
اطلاعات خروجی حافظه را نگه می‌دارد	ثبات داده	۱۶	DR
آدرس حافظه را نگه می‌دارد	ثبات آدرس	۱۲	AR
ثبات پروسسور	اکومولیتور	۱۶	AC
کُد دستورالعمل را نگه‌می‌دارد	ثبات دستور	۱۶	IR
آدرس دستورالعمل را نگه‌می‌دارد	کنتور برنامه	۱۲	PC
اطلاعات موقتی را در خود نگه‌می‌دارد	ثبات موقت	۱۶	TR
داده‌های ورودی را در خود نگه‌می‌دارد	ثبات ورودی	۸	INPR
داده‌های خروجی را در خود نگه می‌دارد	ثبات خروجی	۸	OUTR

ثبات ها و حافظه ها در کامپیوتر پایه



شکل (۵-۳) ثبات‌ها و حافظه یک کامپیوتر مبنا

poorya.kh@gmail.com

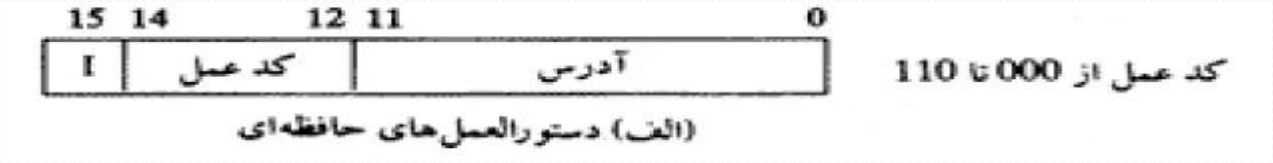


شکل (۵-۴) اتصال ثبات‌های کامپیوتر مبنا با گذرگاه مشترک

دستورالعمل های کامپیوتر

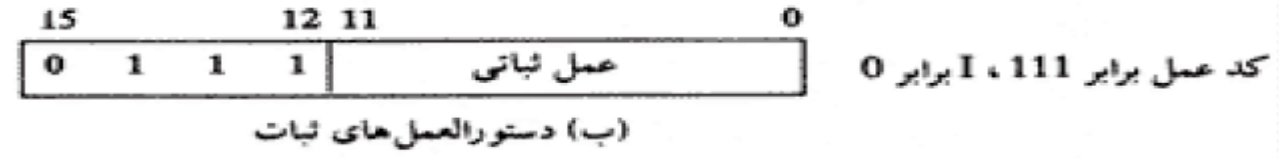
دستور العمل های ارجاع به حافظه

۱۲ بیت برای تعیین آدرس و یک بیت برای مشخص کردن روش آدرس دهی

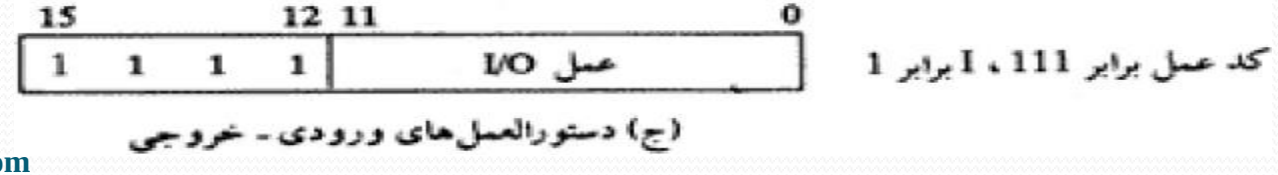


دستور العمل های ارجاع به ثبات

عمل را بر روی AC و یا هر تستی بر روی آن را مشخص می نماید. (عملوندی از حافظه نیاز نیست)



دستور العمل های ورودی خروجی



جدول (۵-۲) دستورات کامپیوتر مبنا

سمبول	کُد هگزا دسیمال I = ۰	کُد هگزا دسیمال I = ۱	توضیح
AND	۰xxx	۸xxx	AND کردن کلمه حافظه با اکومولیتور AC
ADD	۱xxx	۹xxx	جمع کردن کلمه حافظه با اکومولیتور AC
LDA	۲xxx	Axxx	بارکردن کلمه حافظه در اکومولیتور AC
STA	۳xxx	Bxxx	ذخیره نمودن محتوای اکومولیتور AC در حافظه
BUN	۴xxx	Cxxx	انشعاب غیرشرطی
BSA	۵xxx	Dxxx	انشعاب و ذخیره آدرس برگشت
ISZ	۶xxx	Exxx	یک اضافه کن و جابگذار[1] اگر صفر است
CLA	۷۸۰۰		صفرکردن AC
CLE	۷۴۰۰		صفرکردن E
CMA	۷۲۰۰		مکمل کردن AC
CME	۷۱۰۰		مکمل کردن E
CIR	۷۰۸۰		چرخش به‌راست E ,AC
CIL	۷۰۴۰		چرخش به‌چپ E ,AC
INC	۷۰۲۰		اضافه کن یکی به AC
SPA	۷۰۱۰		دستور بعدی را جابگذار، اگر AC مثبت است
SNA	۷۰۰۸		دستور بعدی را جابگذار، اگر AC منفی است
SZA	۷۰۰۴		دستور بعدی را جابگذار، اگر AC صفر است
SZE	۷۰۰۲		دستور بعدی را جابگذار، اگر E صفر است
HLT	۷۰۰۱		کامپیوتر می‌ایستد
INP	F۸۰۰		یک کارکتر به AC وارد کن
OUT	F۴۰۰		یک کارکتر از AC خارج کن
SKI	F۲۰۰		یکی جابگذار اگر پرچم ورودی فعال است
SKO	F۱۰۰		یکی جابگذار اگر پرچم خروجی فعال است
ION	F۰۸۰		وقفه را فعال کن
IOF	F۰۴۰		وقفه را غیرفعال کن

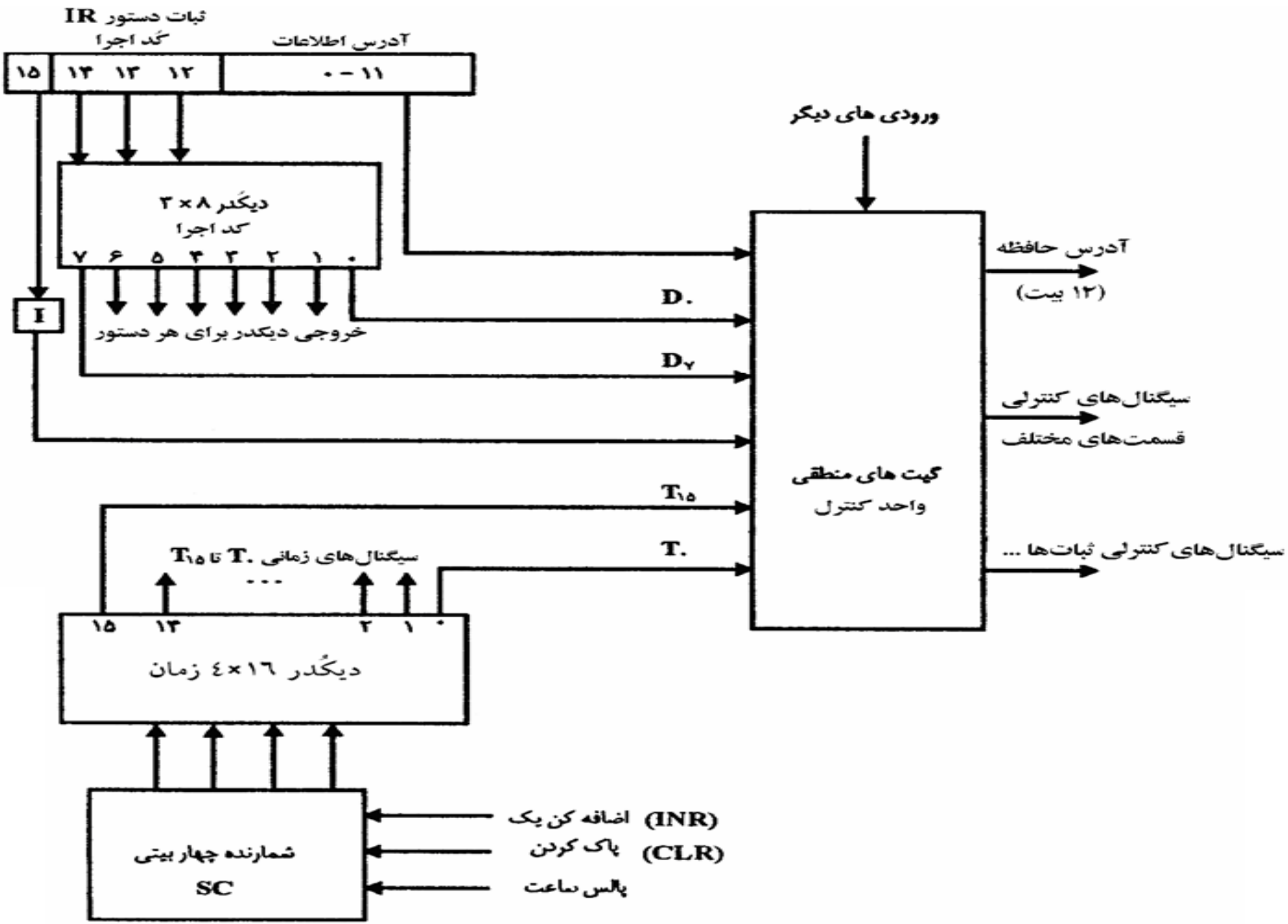
poorya.kh@gmail.com

زمانبندی وکنترل

زمانبندی همه ثبات ها در کامپیوتر پایه به وسیله یک مولد پالس ساعت اصلی کنترل می شود.

سازمان کنترل:

1. کنترل سخت افزاری
2. کنترل ریز برنامه نویسی شده



شکل (۵-۶) واحد کنترل سخت‌افزاری (*Hardwired*) کامپیوتر مبنا

سیکل دستورالعمل

برنامه ای که در واحد حافظه کامپیوتر ذخیره شده از دنباله ای از دستورالعمل ها تشکیل شده است.

در کامپیوتر پایه هر سیکل دستور از فازهای زیر تشکیل شده است:

1. برداشت یک دستور از حافظه
2. دیکد کردن دستور
3. خواندن آدرس موثر
4. اجرای دستورالعمل

برداشت و دیکد

شمارنده برنامه، PC، با آدرس اولین دستورالعمل برنامه بار می شود.

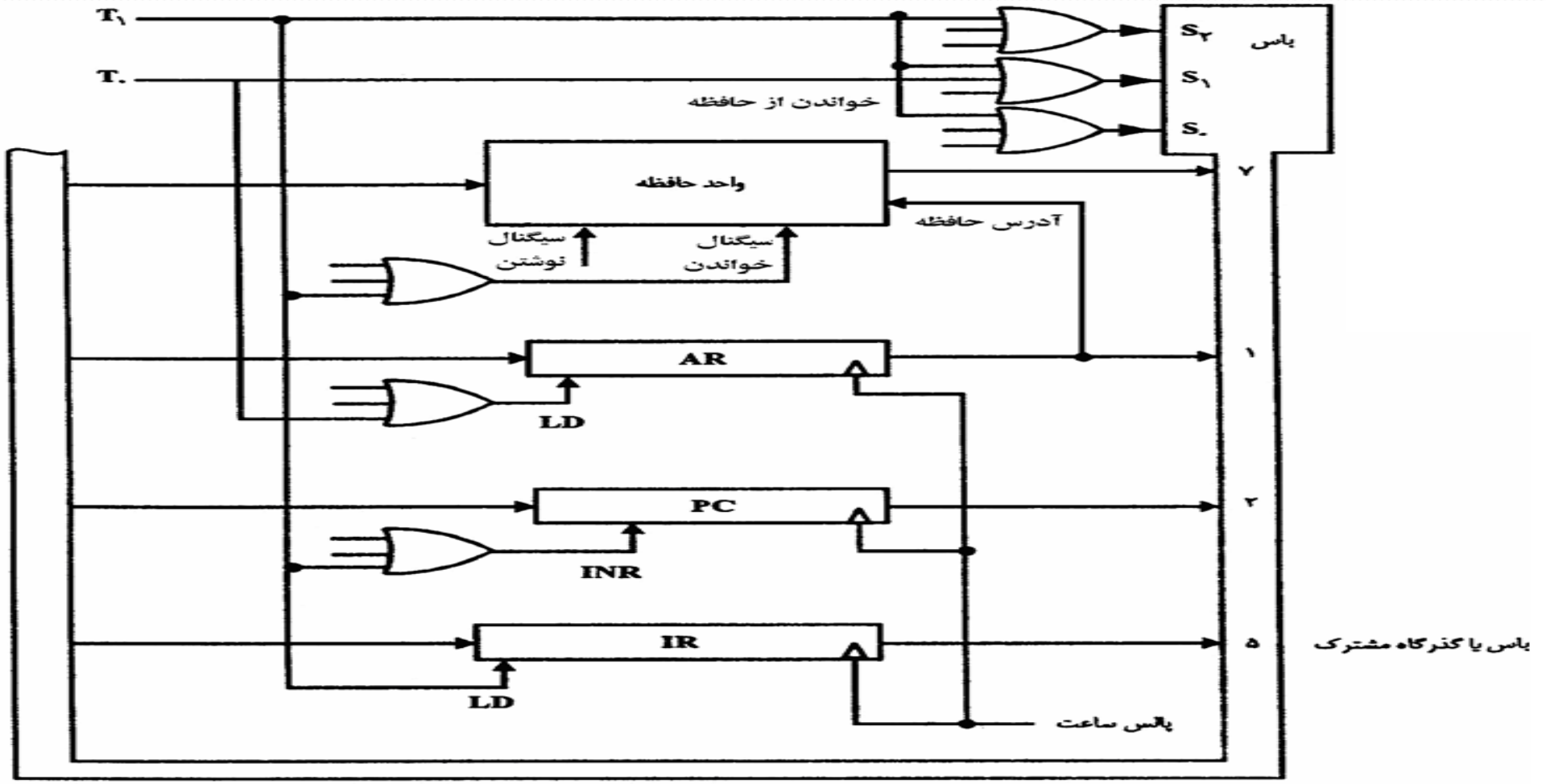
شمارنده SC پاک شده و در نتیجه سیگنال To را تولید می کند.

$$T_0 : AR \leftarrow PC$$

$$T_1 : IR \leftarrow M\,[AR]\,,\ PC \leftarrow PC + 1$$

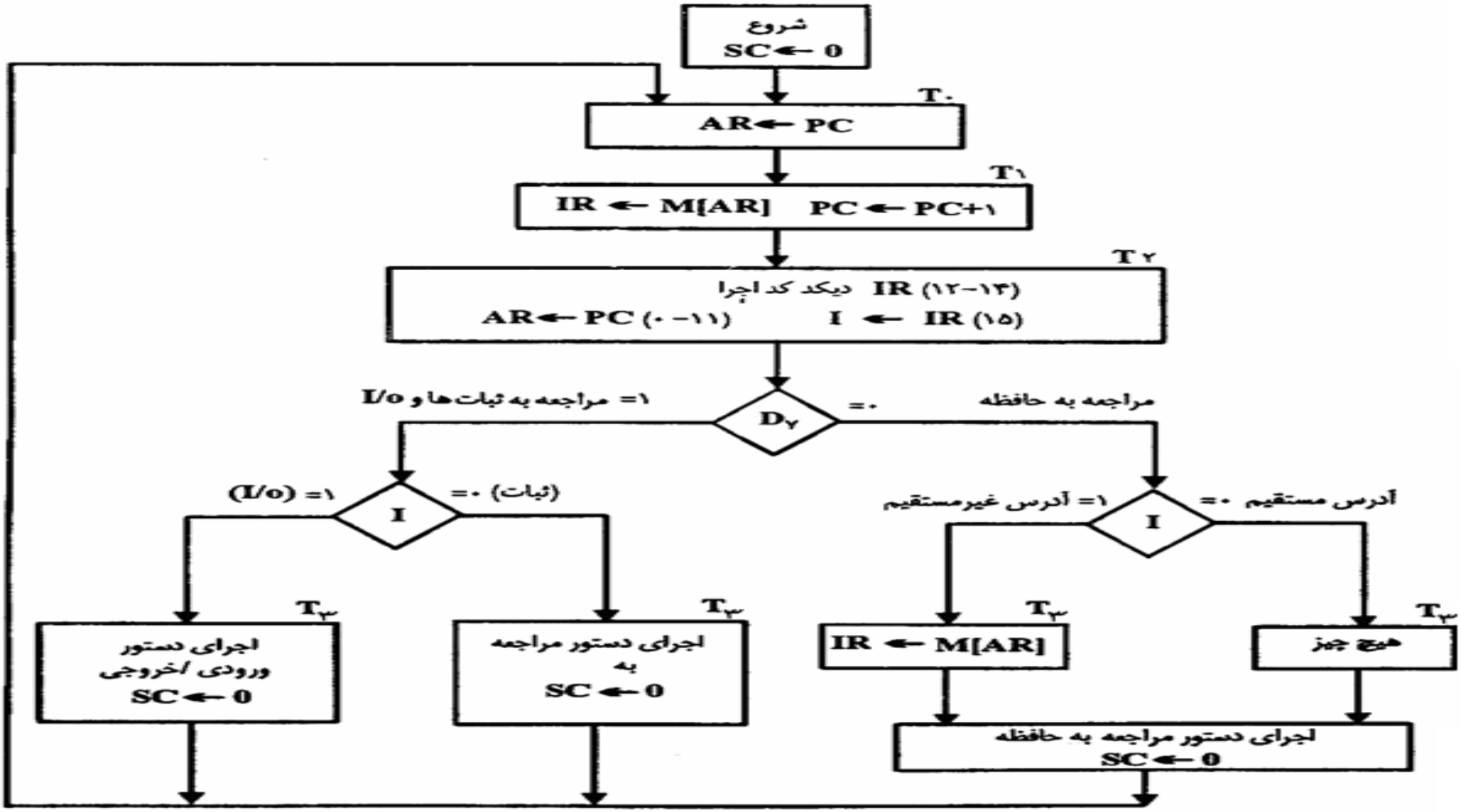
$$T_2 : D_0,\ldots D_7 \leftarrow \text{دیکد}\,,\ AR \leftarrow IR\ (0\text{-}11),I \leftarrow IR(15)$$

انتقال ثبات مربوط به فاز برداشت



شکل (۵–۸) عملیات انتقال ثبات‌ها برای فاز واکشی دستور

تعیین نوع دستورالعمل



شکل (۵-۹) نمودار سیکل دستور (آرایش اولیه)

دستورالعمل های ارجاع به حافظه

جدول (۵-۴) دستورات مراجعه به حافظه

نماد	عمل دیکدر	شرح نمادها
AND	D_0	AC ← AC ∧ M[AR]
ADD	D_1	AC ← AC + M[AR], E ← C_{out}
LDA	D_2	AC ← M[AR]
STA	D_3	M[AR] ← AC
BUN	D_4	PC ← AR
BSA	D_5	M[AR] ← PC, PC ← AR + ۱
ISZ	D_6	M[AR] ← M[AR] + ۱
		اگر ۰ = ۱ + M[AR] پس PC ← PC + ۱

AND با AC

عمل منطقی AND بر روی جفت بیت های AC و حافظه ای که توسط آدرس موثر معین می شود را انجام می دهد.

$D_0\ T_4$: DR ← M[AR]

$D_0\ T_5$: AC ← AC ∧ DR , SC ← 0

ADD با AC

محتوای حافظه مشخص شده توسط آدرس موثر را با مقدار AC جمع می کند.

$$D_1\ T_4: DR \leftarrow M[AR]$$

$$D_1\ T_5: AC \leftarrow AC + DR,\quad E \leftarrow C_{out},\quad SC \leftarrow 0$$

LDA (بار کردن AC)

محتوای حافظه ای که توسط آدرس موثرش مشخص شده را به AC منتقل می کند.

$D_2\ T_4$: DR ← M[AR]

$D_2\ T_5$: AC ← DR , SC ← 0

STA (ذخیره کردن AC در حافظه)

محتوای AC را در حافظه ای که با آدرس موثرش مشخص شده ذخیره می نماید.

$$D_3\ T_4:\ M[AR] \leftarrow AC\ ,\ SC \leftarrow 0$$

BUN(انشعاب بدون شرط)

این دستور برنامه را به دستورالعملی که توسط آدرس موثرش مشخص شده منتقل می نماید.

BUN، به برنامه نویس این امکان را می دهد تا دستوری را در خارج از رشته متوالی برنامه مشخص کند. (برنامه بدون شرط انشعاب می یابد)

$$D_4\ T_4: PC \leftarrow AR, SC \leftarrow 0$$

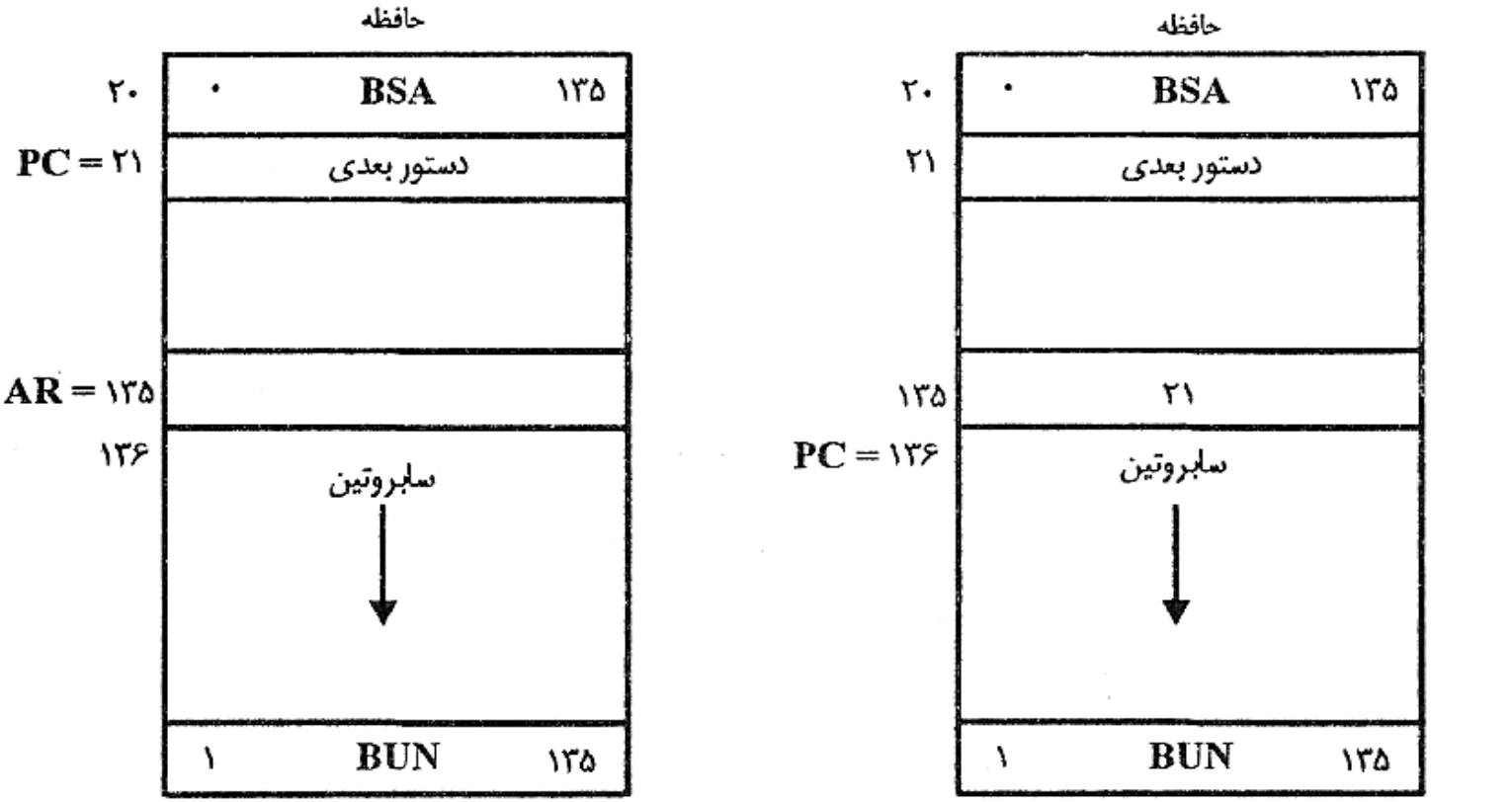
BSA(انشعاب با ذخیره آدرس برگشت)

این دستور برای انشعاب به بخشی از برنامه که زیرروال خوانده می شود، مفید است. آدرس دستور بعدی را که در PC موجود است را در حافظه ای که توسط آدرس موثر معین میشود ذخیره می نماید. سپس آدرس موثر بعلاوه ۱ به PC منتقل می شود تا بعنوان اولین دستور در زیرروال قرار گیرد.

$D_5\ T_4$: M[AR] ← PC , AR ← AR+1

$D_5\ T_5$: PC ← AR , SC ← 0

BSA(انشعاب با ذخیره آدرس برگشت)



الف : وضعیت حافظه و PC و AR در زمان T۳

ب : وضعیت حافظه و PC بعد از اجرای دستور

شکل (۵-۱۰) مثالی از اجرای دستور BSA

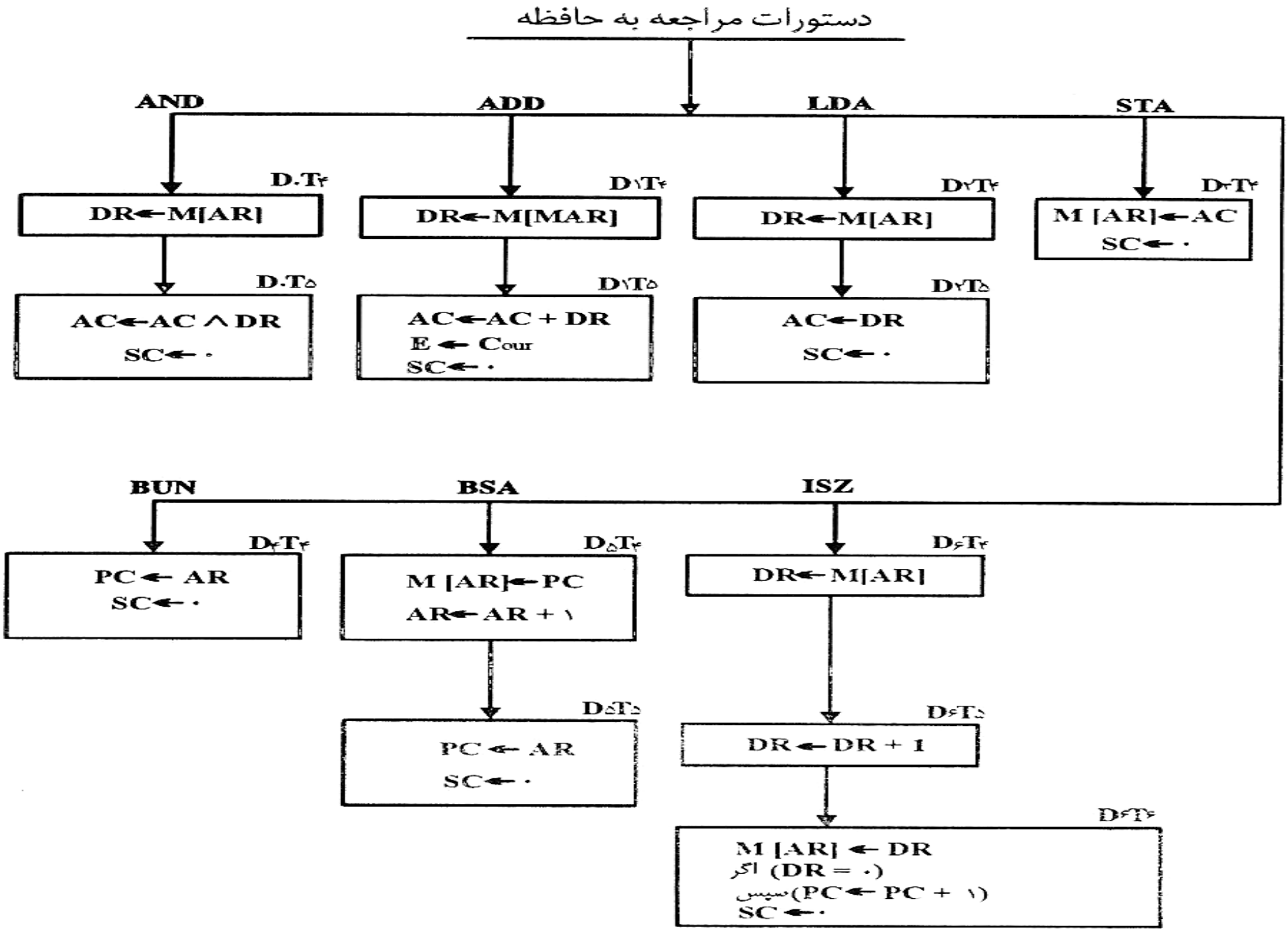
ISZ(افزایش و گذر اگر نتیجه صفر)

کلمه ای که توسط آدرس موثر مشخص می شود را یک واحد افزایش می دهد و اگر مقدار افزایش یافته o شد، PC را یک واحد افزایش می دهد.

$D_6\ T_4$: DR ← M[AR]

$D_6\ T_5$: DR ← DR+1

$D_6\ T_6$: M[AR] ← DR if (DR=0) then (PC←PC+1),SC←0



شکل (۵-۱۱) شمای عملیاتی برای دستورات مراجعه به حافظه

ورودی/خروجی

اطلاعات سری را ارسال و دریافت می کند.

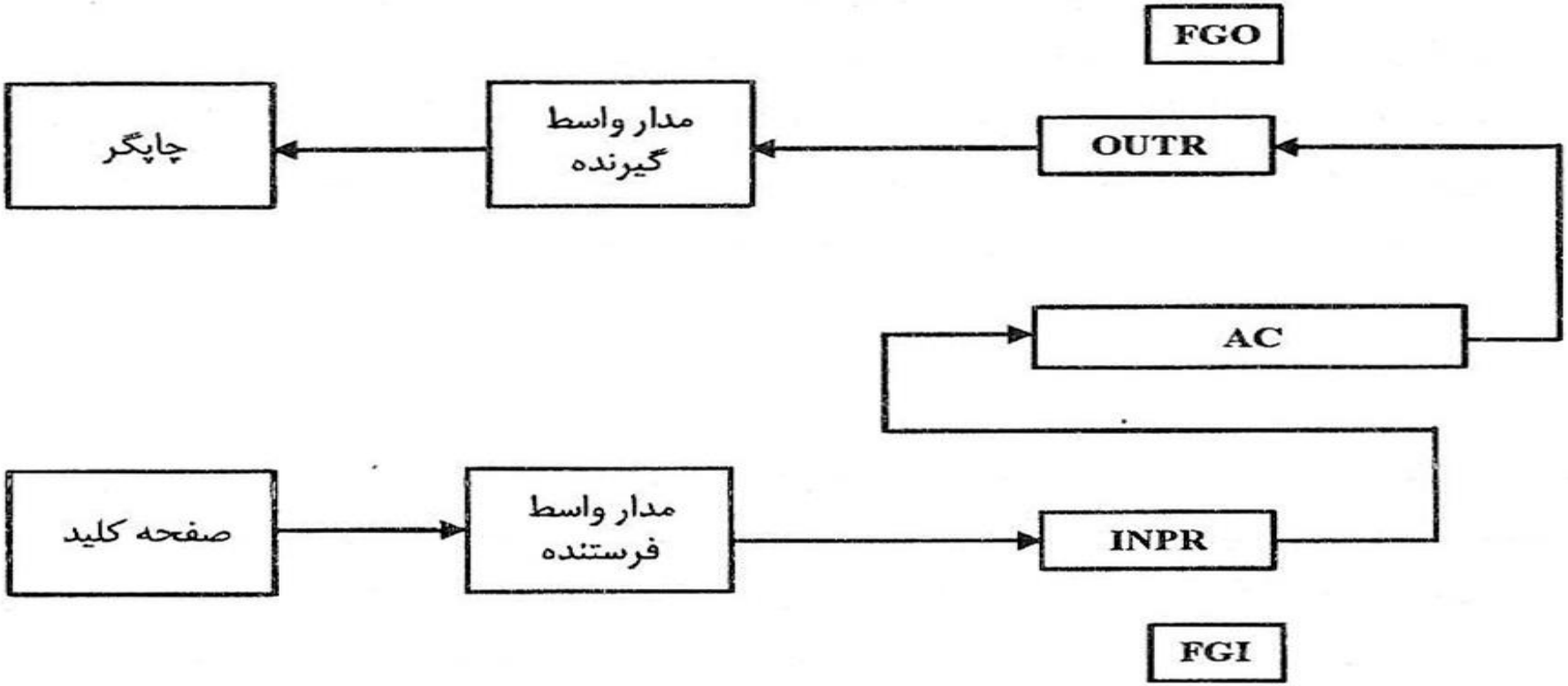
اطلاعات سری وارده از صفحه کلید به داخل INPR شیفت داده می شود.

اطلاعات سری چاپگر درOUTR ذخیره می شود.

این دو ثبات با یک مدار واسط وسیله جانبی به طور سری و با AC ارتباط موازی دارند.

ورودی/خروجی

ثبات‌ها و فلیپ فلاپ های کامپیوتر | مدارهای واسط مخابراتی سری | واحدهای ورودی و خروجی



شکل (۵-۱۲) ساختار واحدهای ورودی/خروجی

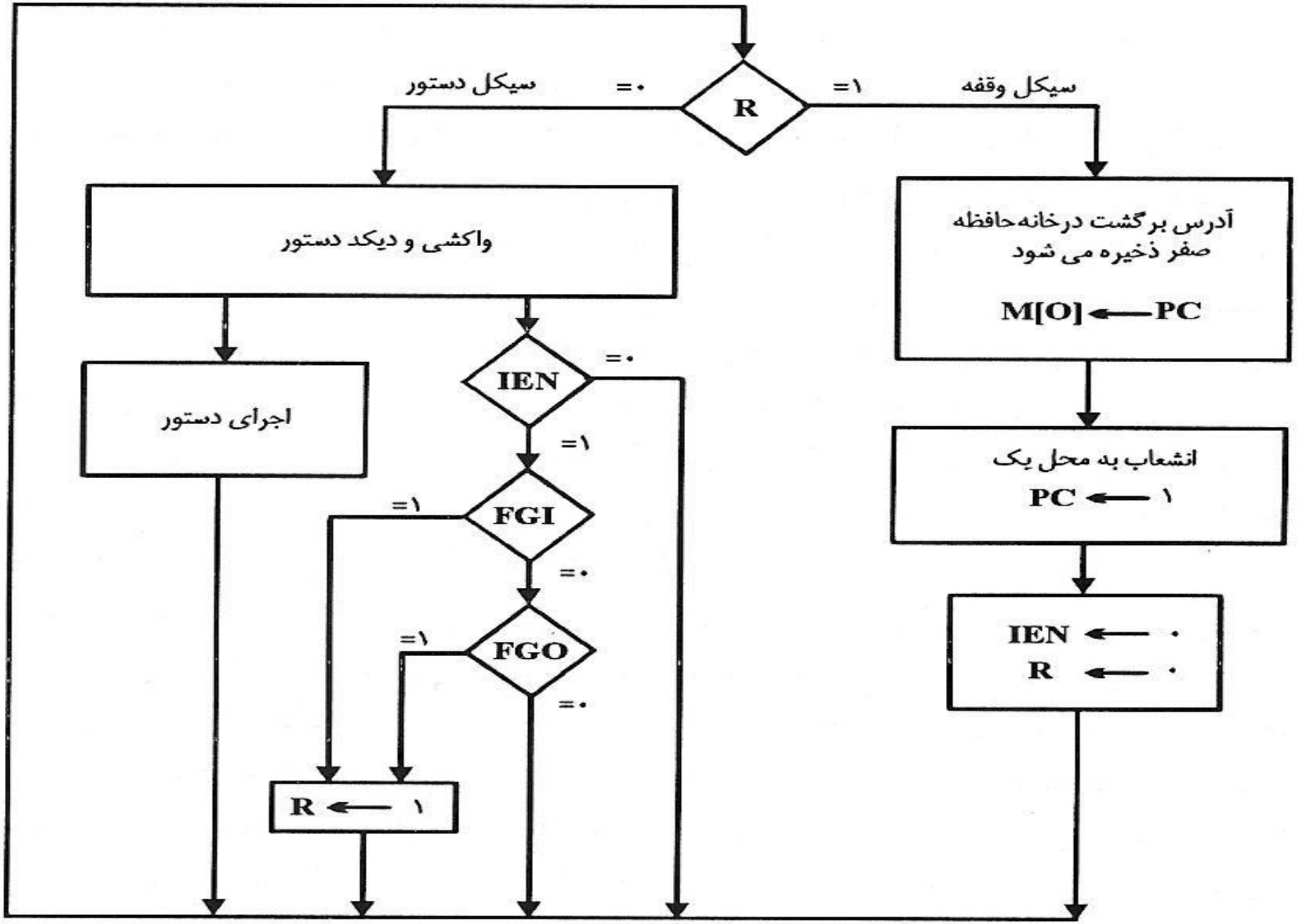
دستورات ورودی/خروجی

جدول (۵-۵) دستورات ورودی/خروجی

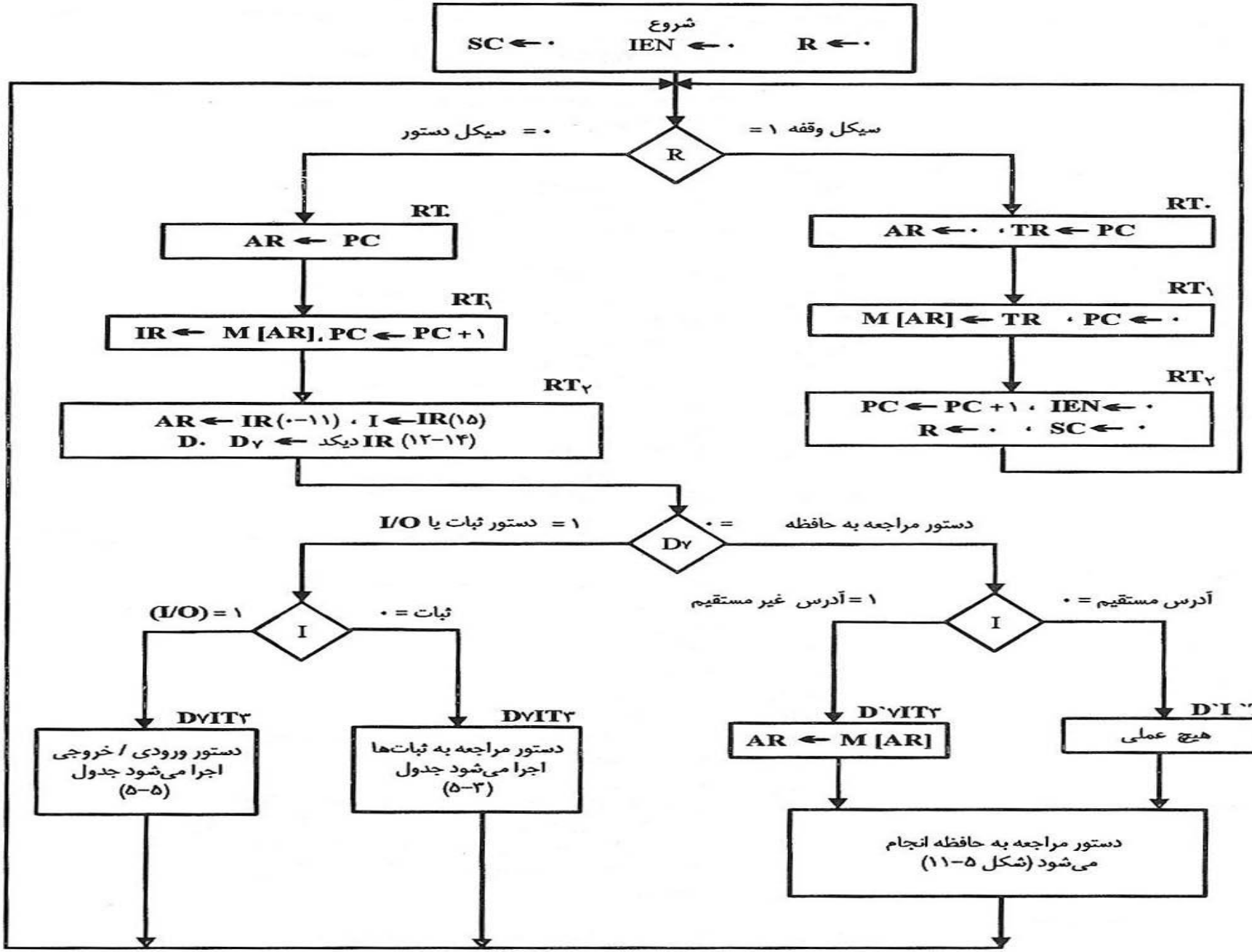
برای تمام دستورات ورودی‌خروجی مشترک است $D_7IT_3 = P$

بیت ۶ تا ۱۱ ثبات IR، که نوع دستور را مشخص می‌نماید. $IR(i) = B_i$

	P :	SC ← ۰	SC صفر می‌شود
INP	PB_{11} :	AC(۰-۷) ← INPR , FGI ← ۰	یک حرف وارد می‌شود
OUT	PB_{10} :	OUTR ← AC(۰-۷) , FGO ← ۰	یک حرف خارج می‌شود
SKI	PB_9 :	(PC ← PC + ۱) پس (FGI = ۱)	یک دستور را رهامی‌کند اگر پرچم ورودی فعال باشد اگر
SKO	PB_8 :	(PC ← PC + ۱) پس (FGO = ۱)	یک دستور را رهامی‌کند اگر پرچم خروجی فعال باشد اگر
ION	PB_7 :	IEN ← ۱	فعال ساز وقفه فعال می‌شود
IOF	PB_6 :	IEN ← ۰	فعال‌ساز وقفه غیرفعال می‌شود



شکل (۵-۱۳) فلوچارت یک سیکل وقفه



شکل (۵-۱۵) شمای عملیاتی کامپیوتر