

# پردازنده های ۸۰۸۶/۸۸

## مقدمه

در فصل قبل با مباحث اساسی طراحی سیستم های مبتنی بر پردازنده ها همچون اتصال حافظه، اتصال وسایل ورودی/خروجی و رمز گشایی آدرس آشنا شدید.

در این فصل می خواهیم پردازنده های ۸۰۸۶ و ۸۰۸۸ اینتل را به عنوان پردازنده های نمونه بررسی کنیم.

پردازنده های ۸۰۸۶/۸۸ در اواخر دهه ۱۹۷۰ به بازار عرضه شدند. ویژگی های پیشرفته این دو پردازنده از جمله معماری ۱۶ بیتی، وجود دستور عملهای ۸ بیتی و ۱۶ بیتی، فضای آدرس دهی یک مگابایتی، فضای آدرس دهی ورودی/خروجی ۶۴ کیلو بیتی، فرکانس کاری مناسب در آن زمان (حداکثر 8 MHz) و ... باعث شد این پردازنده ها محبوبیت زیادی کسب کنند و سرانجام در اوایل دهه ۱۹۸۰، اولین کامپیوترهای شخصی بر پایه پردازنده ۸۰۸۸ به بازار آمدند.

این موضوع تحول عظیمی در صنعت پردازنده ها ایجاد کرد که مهمترین آن سوق دادن راه تکامل پردازنده ها به سمت استفاده بهتر از آنها در کامپیوترهای شخصی بود. امکانات حافظه مجازی، چند کاربری بودن، مدیریت حافظه و ... که در نسلهای بعدی این پردازنده یعنی ۸۰۲۸۶ گنجانده شده بود، در حقیقت پاسخی به نیازهای کاربران کامپیوترهای شخصی بود. در کنار تحولات سخت افزاری پردازنده ها، سیستم های عامل نیز به سرعت رشد کردند و اینکه امروز شاهد وجود چند صد میلیون کامپیوتر شخصی در دنیا هستیم، جز با پیشرفت موازی پردازنده ها و نرم افزارهای مناسب برای استفاده از آنها نمی توانست محقق شود.

علیرغم اینکه پردازنده های پنتیوم که در کامپیوترهای امروزی به صورت گسترده مورد استفاده قرار می گیرند، دارای امکانات فوق العاده ای هستند که به بخشی از آنها در فصل قبل اشاره شد، اما پایه ای اساسی آنها همان پردازنده های ۸۰۸۶/۸۸ است؛ بنابراین برای آشنایی با پردازنده های پیشرفته و امکانات آنها آشنایی با ۸۰۸۶/۸۸ ضروری است. به علاوه معماری کامپیوترها که در ابتدا بر اساس ۸۰۸۸ بنا نهاده شد، تاکنون به صورت اساسی تغییر نکرده و آشنایی با پردازنده ۸۰۸۸ و کامپیوتر PC-XT که اولین نسل کامپیوترهای شخصی مبتنی بر ۸۰۸۸ است، برای شناسایی ساختار داخلی کامپیوتر و نحوه استفاده از امکانات آن مفید است.

هدف این فصل معرفی سخت افزار و نرم افزار ۸۰۸۶/۸۸ است؛ البته به جای توضیح مفصل در رابطه با کلیه امکانات این پردازنده ها که موضوعی پیچیده، وقتگیر و تاحدی غیرضروری است، کوشیده ایم با پررنگ کردن ویژگیهایی از این پردازنده ها که در نسلهای بعدی پردازنده های اینتل گسترش یافته

و به تغییرات مهمی در معماری پردازنده‌ها منجر شد، به تشریح ساختار کاری ۸۰۸۶/۸۸ بپردازیم؛ به صورتی که پس از مطالعه این فصل نکات لازم جهت طراحی یک سیستم کنترلی مبتنی بر ۸۰۸۸ را آموخته باشید.

## ساختار سخت افزاری پردازنده‌های ۸۰۸۶/۸۸

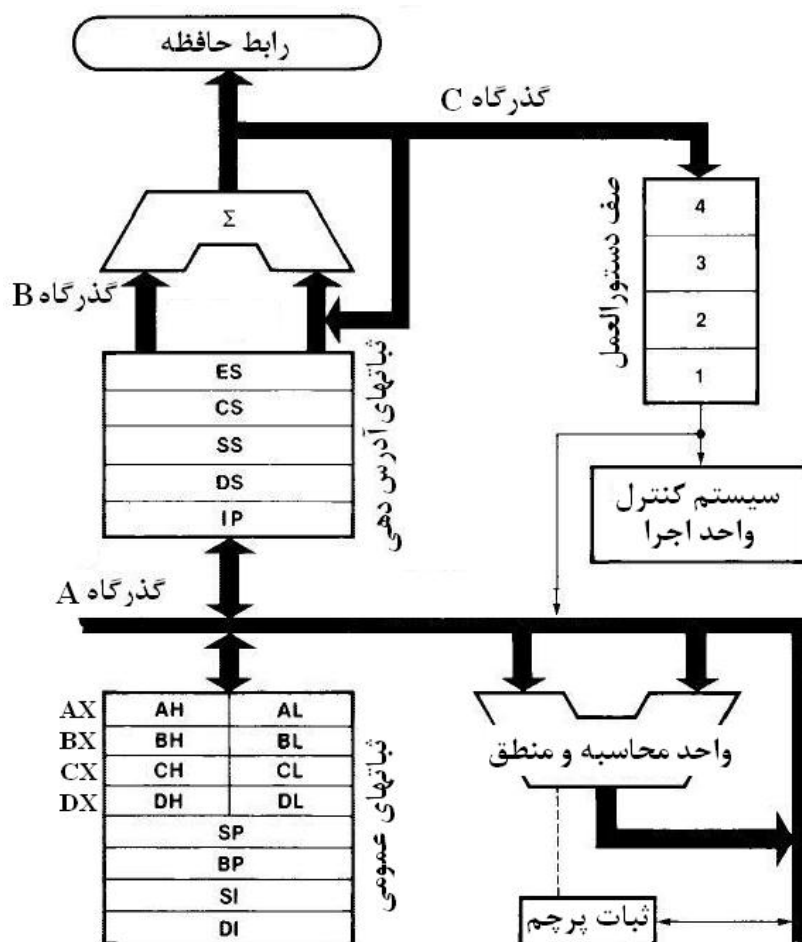
همانطور که گفته شد، پردازنده ۸۰۸۵، پردازنده‌ای با معماری ۸ بیتی بود. اینتل طی یک تحول مهم در سال ۱۹۷۸ پردازنده ۸۰۸۶ را معرفی کرد که چه از نظر ساختار داخلی (گذرگاه داده داخلی) و چه از نظر تعداد پینهای داده (گذرگاه داده خارجی)، یک پردازنده ۱۶ بیتی بود. به رغم انتظار اینتل، از پردازنده ۸۰۸۶ استقبال چندانی نشد؛ دلیل اصلی این امر، وجود هزاران دستگاه کنترلی مبتنی بر گذرگاههای داده ۸ بیتی منطبق بر ۸۰۸۵ بود که تغییر ناگهانی آنها به سیستم‌های ۱۶ بیتی هیچ توجیه اقتصادی نداشت.

فروش ۸۰۸۶ در سال ۱۹۷۸ یک شکست مطلق بود؛ اما اینتل به جای عقب نشینی در مقابل این مشکل، دست به کاری جالب زد که در نوع خود در آن زمان بی نظیر بود. اینتل در سال ۱۹۷۹ پردازنده ۸۰۸۸ را معرفی کرد که از نظر معماری داخلی تطابق کامل با ۸۰۸۶ داشت؛ بنابراین می‌توانست از ویژگیهای مفید معماری آن مثل صف دستورالعمل، ثباتها و دستورات ۱۶ بیتی و ... استفاده نماید. اما ۸۰۸۸ بر خلاف ۸۰۸۶ دارای گذرگاه داده خارجی ۸ بیتی (۸ پین برای گذرگاه داده) بود تا بتواند به راحتی جایگزین پردازنده سیستم‌های ۸ بیتی آن زمان شود. استقبال فوق‌العاده از ۸۰۸۸ و استفاده از آن به عنوان پردازنده اولین نسل کامپیوترهای شخصی، پاسخی به ایده جالب اینتل در مورد تغییر ساختار ۸۰۸۶ بود.

به مرور با ایجاد سیستم‌های ۱۶ بیتی، استفاده از پردازنده ۸۰۸۶ و نسلهای بعدی آن متداول شد و جایگاه ۸۰۸۸ به عنوان پلی بین پردازنده‌های ۱۶ بیتی و ۸ بیتی در عرصه پردازنده‌ها محفوظ ماند. بنابر آنچه گفته شد، پایه اساسی بحث ما پردازنده ۸۰۸۸ خواهد بود و در قسمت آخر این فصل گذری مختصر بر ۸۰۸۶ نیز خواهیم داشت.

## معماری پردازنده ۸۰۸۸

معماری داخلی پردازنده ۸۰۸۸ را در شکل زیر می بینید.



قسمتهای مهم این ساختار عبارتند از :

### - واحد محاسبه آدرس

در ۸۰۸۸ یک واحد محاسبه آدرس در نظر گرفته شده که با استفاده از ثباتهای آدرس دهی IP، DS، SS، CS و ES عمل استخراج آدرس و تبدیل آدرس منطقی به آدرس واقعی (فیزیکی) را انجام دهد. در بخش آینده به معنای این عمل پی خواهیم برد.

### - رابط حافظه

آدرس محاسبه شده در واحد محاسبه آدرس به رابط حافظه داده می شود تا با استفاده از آن تبادل داده مناسب با خارج پردازنده را انجام دهد.

## - صف دستورالعمل

یکی از مهمترین قسمت‌های ساختار داخلی ۸۰۸۸، بخش صف دستورالعمل است. این بخش مبنای مباحث مهمی در علم پردازنده‌ها مانند Cache، چند وظیفگی، موازی سازی و ... است. در فصل قبل خواندید که پردازنده برای اجرای یک دستورالعمل باید سه مرحله واکنشی یا دریافت دستور از حافظه (Fetch)، رمزگشایی دستور (Decode) و اجرای دستور (Execution) را انجام دهد. در پردازنده نسل قبل از ۸۰۸۸، یعنی ۸۰۸۵، این عمل به صورت پشت سرهم انجام می شد؛ یعنی ۸۰۸۵ برای اجرای یک دستورالعمل، آن را از حافظه دریافت، رمزگشایی و سپس اجرا می کرد. سپس این سه مرحله برای دستور بعدی دنبال می شد. این روش ساده است، اما با استفاده از امکانات ۸۰۸۸ می توان با نوعی موازی سازی این روش را بهبود بخشید.

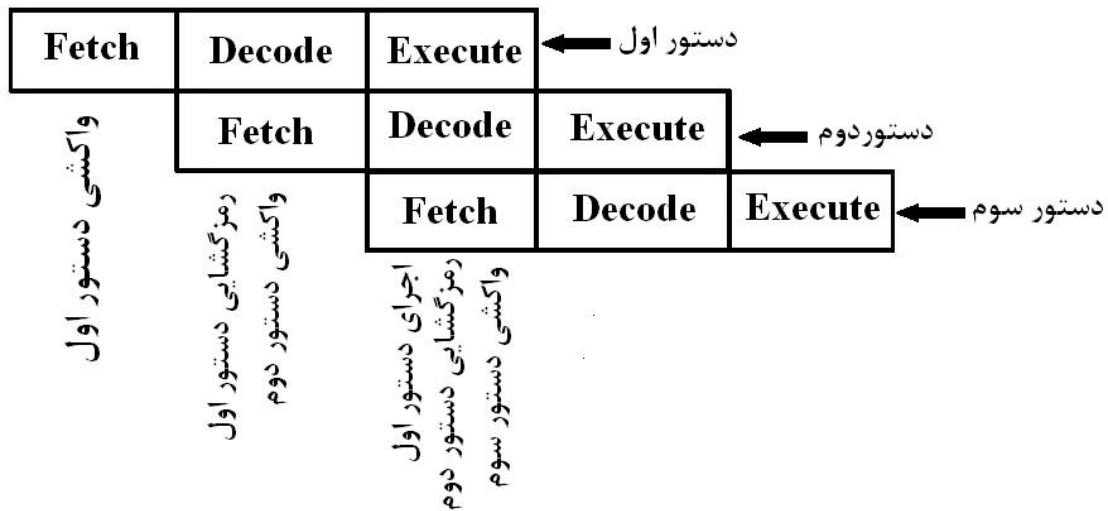
همانطور که در شکل معماری داخلی ۸۰۸۸ می بینید، واحد محاسبه آدرس از واحد اجرای دستور مجزاست. این موضوع به ۸۰۸۸ این امکان را می دهد که در حین اجرای یک دستور (موازی با اجرای یک دستور) تا ۴ دستور بعد را از حافظه دریافت کرده و در یک صف ذخیره کند تا به موقع رمزگشایی و اجرا شوند. این کار باعث کاهش زمان کلی اجرای برنامه می شود.

طول صف دستورالعمل در ۸۰۸۶/۸۸، ۴ دستور است که در پردازنده های بعدی افزایش یافت و در نهایت به ایجاد Cache که در پردازنده های فعلی یکی از معیارهای قدرت آنها به شمار می رود، منجر گردید؛ بعلاوه ایده صف دستورالعمل آغاز بحث موازی سازی در پردازنده ها نیز به شمار می رود. ایده جدا کردن مرحله واکنشی از مراحل رمزگشایی و اجرا، در پردازنده های بعدی اینتل منجر به ایده خط لوله<sup>۱</sup> شد که یکی از موضوعات بسیار مهم معماری پردازنده های فعلی به شمار می رود.

### خط لوله

همانطور که گفته شد پردازنده ۸۰۸۵ سه مرحله واکنشی، رمزگشایی و اجرا را برای هر دستور به طور مجزا و مستقل از دستورات قبلی و بعدی انجام می دهد. این موضوع باعث می شود که نتوانیم از امکانات پردازنده به درستی استفاده کنیم. چون وقتی ۸۰۸۵ دستور اول را واکنشی می کند و سپس آن را به واحد رمزگشایی می سپارد، در حین رمزگشایی دستور اول واحد واکنشی بیکار است، در حالی که واکنشی دستور دوم به رمزگشایی دستور اول هیچ ارتباطی ندارد و این دو عمل وابسته به هم نیستند و می توانند موازی هم اجرا شوند. به نحو مشابه وقتی ۸۰۸۵ دستور اول را پس از رمزگشایی به واحد اجرا می فرستد، واحدهای واکنشی و رمزگشایی بیکارند؛ در حالی که در همین زمان می توانند به ترتیب به واکنشی دستور سوم و رمزگشایی دستور دوم پردازند.

<sup>1</sup> Pipelining



اینتل در پردازنده‌های بعدی خود، با بهره‌گیری از ایده صف دستورالعمل ۸۰۸۸ به موضوع استفاده از واحدهای واکشی، رمزگشایی و اجرای دستور در زمانهای بیکاری جامه عمل پوشاند. این تکنیک به نام **خط لوله** معروف است که شکل آن را در بالا مشاهده می‌کنید. این نام از تفاوت روشهای آوردن آب از جایی به جای دیگر گرفته شده است؛ به جای آوردن آب به صورت سطل سطل از چاه (مانند آنچه ۸۰۸۵ برای اجرای دستورات انجام می‌دهد) می‌توان یک خط لوله از چاه به منزل کشید تا آب به صورت جاری و پشت سرهم منتقل شود (مانند آنچه در پردازنده‌های دارای خط لوله می‌گذرد).

### - سیستم کنترل واحد اجرا

این واحد پس از دریافت دستورالعمل بعدی، دستورات کنترلی لازم برای اجرای آن را به قسمت‌های مختلف پردازنده ارسال می‌کند.

### - واحد محاسبه و منطق (ALU)

این واحد علاوه بر انجام اعمال اساسی محاسباتی و منطقی، در ۸۰۸۸ قادر به انجام اعمال ضرب و تقسیم نیز می‌باشد.

### - ثبات‌های داخلی

۸۰۸۸ به جز ثبات‌های آدرس دهی، دارای ۸ ثبات عمومی ۱۶ بیتی به نام‌های AX, BX, CX, DX (۴ ثبات فوق، به ثبات‌های همه منظوره معروفند)، SI, BP, SP و PI است که با کاربردهای آنها در بخش‌های بعد آشنا خواهید شد.

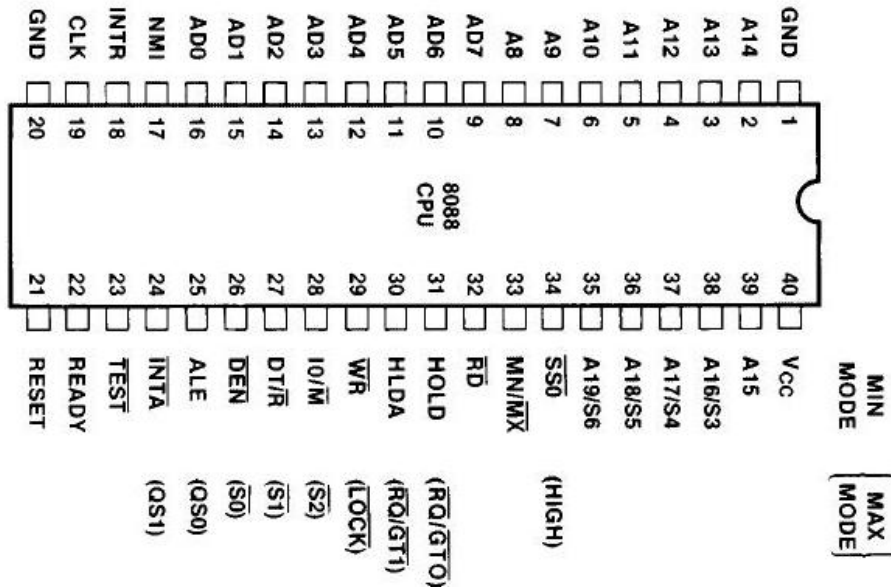
نکته قابل ذکر این است که چون ۸۰۸۸ قابلیت اجرای دستورات ۸ بیتی و ۱۶ بیتی را دارد، ثبات‌های همه منظوره آن به صورت دو ثبات ۸ بیتی مجزا نیز قابل دسترسی اند؛ مثلاً ثبات ۱۶ بیتی AX خود شامل ۲ ثبات ۸ بیتی AH (۸ بیت بالای AX) و AL (۸ بیت پایین AX) است.

## گذرگاههای A و B و C

این گذرگاهها، مسیرهای داخلی پردازنده برای جابجایی اطلاعات مختلف مانند کدهای عملیاتی یا داده هاست.

## پینه‌های پردازنده ۸۰۸۸

پس از بررسی معماری داخلی ۸۰۸۸، به بررسی پینه‌های تراشه ۴۰ پایه ۸۰۸۸ که در شکل زیر نشان داده شده است، می پردازیم.



## پین ورودی MN/MX (33):

چون تعداد پینه‌های ۸۰۸۸ محدود است، برای گنجاندن کاربردهای زیادتر در همین تعداد پین محدود، دو مدکاری برای ۸۰۸۸ به نام مدهای مینی‌م (Min) و ماکزیم (Max) تعریف شده است. اگر این پین به ولتاژ +۵ ولت («یک» منطقی) متصل باشد، ۸۰۸۸ در مد مینی‌م کار می کند و پینه‌های آن طبق نامهایی که در جلو آنها ذکر شده است، نامیده شده و عمل می کنند؛ اما اگر این پین به زمین (صفر منطقی) متصل باشد، معانی تعدادی از پینه‌ها عوض شده و به جای معانی اصلی، طبق نامهایی که در پرانتز جلوی آنها ذکر شده است، نامیده شده و عمل می کنند. در این فصل به شرح پینه‌ها در مد مینی‌م می پردازیم.

برای شناخت پینه‌های مد مینی‌م، در ابتدا به جستجوی پینه‌هایی می پردازیم که در پردازنده فرضی فصل قبل به آنها برخورد کرده ایم.

## گذرگاه داده

پینه‌های AD0 تا AD7 (پینه‌های ۱۶ تا ۹) ، پینه‌های مربوط به گذرگاه داده خارجی ۸۰۸۸ هستند (به یاد می‌آورید که گذرگاه داده داخلی ۸۰۸۸ مانند ۸۰۸۶، ۱۶ بیتی است و گذرگاه خارجی آن برای تطابق با دستگاههایی که گذرگاه داده ۸ بیتی دارند، به صورت ۸ بیتی در نظر گرفته شده است). همانطور که در فصل قبل دیدید، چون پردازنده برای ارسال و دریافت اطلاعات از گذرگاه داده خود استفاده می‌کند، این ۸ پین به صورت دوطرفه<sup>۱</sup> طراحی شده‌اند.

## گذرگاه آدرس

پینه‌های A15 تا A19 (۳۹ تا ۳۵) و A8 تا A14 (۸ تا ۲) ، پینه‌های مربوط به گذرگاه آدرس ۸۰۸۸ هستند. پس A0 تا A7 کجاست؟! باز هم پینه‌های ۱۶ تا ۹ را ببینید. نام AD (Address/Data) بیانگر این نکته است که این پینه‌ها هم به عنوان گذرگاه داده و هم به عنوان ۸ بیت ابتدایی گذرگاه آدرس به کار رفته‌اند. در واقع به دلیل صرفه جویی در تعداد پینه‌های تراشه، در پردازنده‌های ۸۰۸۶/۸۸ پینه‌های مربوط به گذرگاه داده با تعدادی از پینه‌های گذرگاه آدرس مشترکند؛ در واقع تعدادی از پینه‌ها هم نقش گذرگاه داده و هم نقش قسمتی از گذرگاه آدرس را بازی می‌کنند. چگونه چنین چیزی ممکن است؟!

طبیعی است که در یک لحظه یک پین باید یا محتوی داده باشد یا محتوی آدرس؛ بنابراین در لحظه T، ۸ پین AD0 تا AD7 محتوی آدرس (A0 تا A7) هستند که به همراه پینه‌های A8 تا A19، گذرگاه آدرس ۸۰۸۸ را تشکیل می‌دهند و در لحظه T' این ۸ پین محتوی داده (D0 تا D7) می‌باشند.

اولین سؤالی که به ذهن می‌رسد این است که آیا ابتدا لحظه T فرا می‌رسد یا T'؟ به بیان دیگر برای اجرای یک دستور آیا لازم است که پینه‌های AD0 تا AD7 ابتدا برای آدرس دهی به کار روند یا برای ارسال و دریافت داده‌ها؟

اگر مطالب فصل قبل را به دقت خوانده باشید، پاسخ شما " آدرس " خواهد بود. همانگونه که گفته شد، شماره ای (آدرسی) که پردازنده روی گذرگاه آدرس خود می‌گذارد، بیانگر هویت واحد جانبی است که مجاز به استفاده از گذرگاه داده می‌باشد. بنابراین طبیعی است که ابتدا باید آدرس به صورت کامل مشخص شود تا هویت واحدی که مجاز به استفاده از گذرگاه داده است تعیین گردد و سپس گذرگاه داده در اختیار آن قرار گیرد. بنابراین پینه‌های AD0 تا AD7 در ابتدا حاوی ۸ بیت ابتدایی آدرس (A0 تا A7) هستند تا بعلاوه پینه‌های دیگر آدرس (A8 تا A19) شماره واحدی که

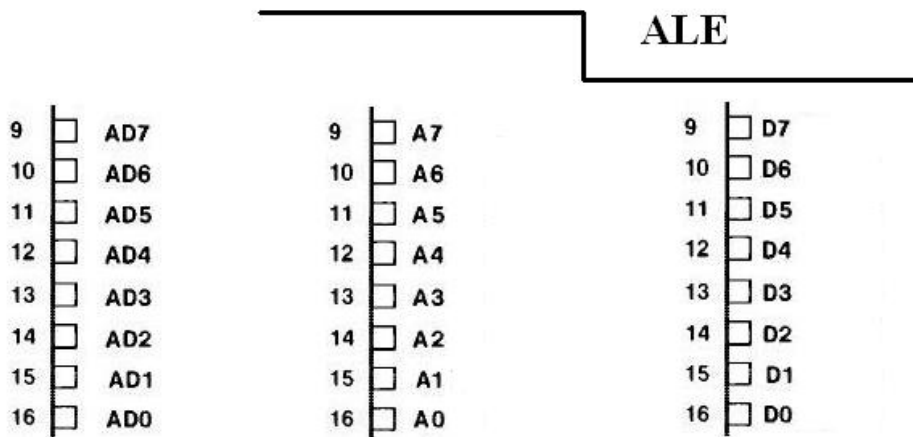
<sup>1</sup> Bidirectional

<sup>2</sup> Multiplexed

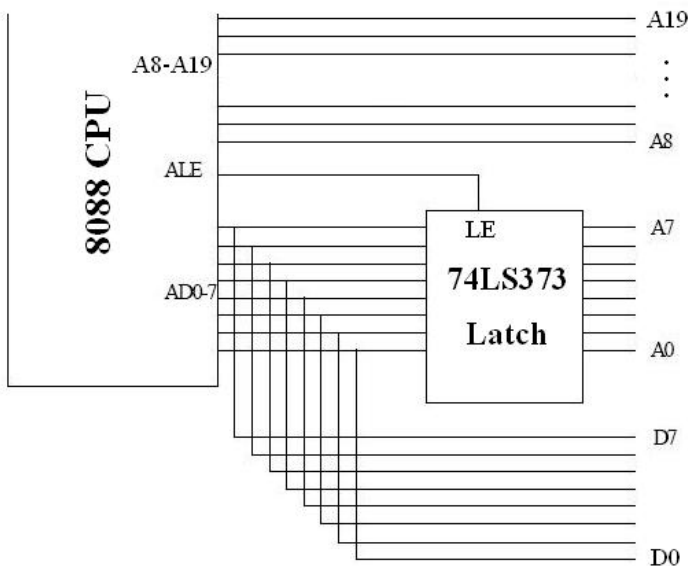
می‌تواند از گذرگاه داده استفاده کند را معلوم کنند و سپس این ۸ بیت به صورت گذرگاه داده (D0 تا D7) عمل می‌کنند تا برای ارسال و دریافت داده‌ها مورد استفاده قرار گیرند. پرسش دوم این است که سیستم باید به چه نحوی طراحی شود تا در زمان مناسب با این پین‌ها به صورت پین آدرس و در زمان مناسب به صورت پین داده رفتار کند؟

### پین خروجی ALE (Address Latch Enable) (۲۵)

این پین توسط پردازنده برای اعلان وجود آدرس روی پین‌های AD0 تا AD7 به کار می‌رود. بنا بر آنچه گفته شد، در ابتدای اجرای دستور این پین به نشانه وجود آدرس، «یک» می‌شود و پس از مدت زمانی، به نشانه اختصاص پین‌های AD0 تا AD7 به گذرگاه داده صفر می‌شود.



توجه کنید که سیستم کنترلی برای رمزگشایی آدرس به طور دائم به آدرس نیاز دارد. پین‌های A8 تا A19 که در طول اجرای دستور حاوی بیت‌های آدرس هستند؛ بنابراین باید به نحوی ۸ بیت پایینی آدرس در مدت زمان کوتاهی که در ابتدای اجرای دستور روی پین‌های AD0 تا AD7 قرار می‌گیرند،



جایی قفل یا نگهداشته شوند تا در ادامه کار که این پین‌ها به گذرگاه داده اختصاص می‌یابند، بتوانند در کنار A8 تا A19 برای رمزگشایی آدرس به کار روند. برای این کار از سیستم زیر استفاده می‌کنیم.

وقتی پردازنده A0 تا A7 را روی پین‌های AD0 تا AD7 قرار می‌دهد، پین ALE را «یک» می‌کند. بنابراین نگهدار فعال شده و A0 تا A7 را در خروجی خود می‌گذارد. پس از مدت زمانی که پردازنده می‌خواهد AD0 تا



AD7 را به D0 تا D7 اختصاص دهد، ALE را صفر می‌کند. پس نگهدار غیرفعال می‌شود و A0 تا A7 را در خروجی خود نگه می‌دارد که در کنار A8 تا A19 آدرس ۲۰ بیتی مورد نظر ۸۰۸۸ را تشکیل دهد. از این پس پینهای AD0 تا AD7 به D0 تا D7 اختصاص خواهد داشت.

**پرسش:** با توجه به اینکه گذرگاه آدرس یکطرفه و گذرگاه داده دو طرفه است، در چه زمانی AD0 تا AD7 یکطرفه و در چه زمانی دو طرفه است؟

**پرسش:** با توجه به ۲۰ بیتی بودن گذرگاه آدرس ۸۰۸۸، فضای آدرس دهی این پردازنده چقدر است؟

**پرسش:** پردازنده ۸۰۸۸ برای مراجعه به ورودی/خروجی تنها از ۱۶ بیت گذرگاه آدرس ۲۰ بیتی خود استفاده می‌کند. این پردازنده مجموعاً می‌تواند چند وسیله ورودی/خروجی را بشناسد؟ (این فضای آدرس دهی ورودی/خروجی در پردازنده های بعدی اینتل نیز ثابت ماند).

### پین ورودی CLK (Clock) (۱۹)

پالس هماهنگ کننده فعالیت‌های داخلی ۸۰۸۸ (پالس ساعت) از طریق این پین به پردازنده داده می‌شود. در فصل بعد با نحوه تولید این پالس ساعت توسط تراشه 8284 آشنا خواهید شد. فرکانس این پالس ساعت در ۸۰۸۸ پایه برابر 5MHZ و در ۸۰۸۸-۲ که نسخه پیشرفته تر ۸۰۸۸ است، برابر 8 MHZ می‌باشد.

### پینهای خروجی RD و WR (۳۲ و ۲۹)

پردازنده از این دو پین برای اعلام اینکه در حال خواندن یا نوشتن داده است استفاده می‌کند.

### پین خروجی IO/M (۲۸)

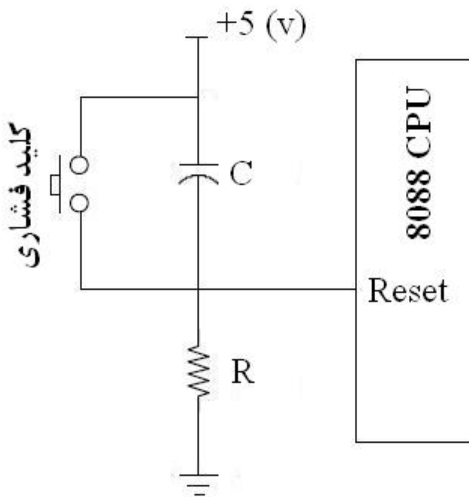
پردازنده با قرار دادن «یک» روی این پین اعلام می‌کند که مشغول تبادل داده با وسایل ورودی و خروجی است و وجود «صفر» روی این پین بیانگر تبادل داده پردازنده با حافظه است. معمولاً از ترکیب ۳ سیگنال اخیر، ۴ سیگنال مهم در پردازنده ها می‌سازیم که رده تبادل داده پردازنده با خارج را نشان می‌دهد.



**پرسش:** با توجه به ۶۴ کیلو بایتی بودن فضای آدرس دهی ورودی/خروجی در ۸۰۸۸، هنگامی که IO/M برابر یک است، مقادیر موجود روی کدامیک از پینهای گذرگاه آدرس معتبر هستند؟

### پین ورودی Reset (۲۱)

با فعال شدن این پین، ۸۰۸۸ اجرای برنامه فعلی را قطع کرده و به وضعیت اولیه پردازنده هنگام



روشن شدن (که بعداً بررسی خواهیم کرد) باز می‌گردد و اجرای برنامه را از سر می‌گیرد. این پین حداقل برای ۴ سیکل ساعت باید (یک) باقی بماند تا عمل Reset به درستی انجام شود.

معمولاً برای اینکه علاوه بر هنگام نیاز، هنگام اتصال برق به پردازنده نیز عمل Reset به طور اتوماتیک انجام شود، از مداری به نام Auto-Power on-Reset استفاده می‌شود که شکل صفحه قبل این مدار را نشان می‌دهد.

هنگام روشن شدن سیستم (متصل شدن ولتاژ +۵ ولت)، خازن اتصال کوتاه است و پین Reset به +۵ ولت متصل و فعال

می‌شود. خازن شروع به شارژ شدن می‌کند و پس از مدتی که به مقادیر C و R بستگی دارد، شارژ شده و اتصال بین پین Reset و +۵ ولت را قطع می‌کند.

هرگاه نیاز به Reset مدار بود، با فشردن کلید فشاری پین Reset به +۵ ولت متصل شده و خازن تخلیه می‌شود. با رها کردن کلید، خازن مجدداً شارژ شده و ارتباط پین Reset با +۵ ولت را قطع می‌کند.

**پرسش:** مقادیر R و C چگونه انتخاب می‌شوند؟

معمولاً سیگنال Reset پردازنده ۸۰۸۸ توسط تراشه ۸۲۲۴ تأمین می‌شود که در فصل بعد مورد بررسی قرار خواهد گرفت.

### پینه‌های HOLD و HLDA (۳۰ و ۳۱)

این دو پین برای در اختیار گرفتن گذرگاه‌های سیستم توسط وسایلی غیر از پردازنده (مثلاً در DMA) به کار می‌روند

### پینه‌های خروجی DT/R و DEN (۲۶ و ۲۷)

این دو پین از سیگنال‌های لازم در طراحی سیستم هستند. به طور خلاصه بدانید که پین DEN برای فعال کردن بافرهای متصل به گذرگاه سیستم و پین DT/R برای مشخص کردن جهت جریان داده در این بافرها به کار می‌رود.

### پین ورودی Ready (۲۲)

اگر حافظه یا وسیله ورودی/خروجی قادر نباشد با سرعت پردازنده کار کند باید به نحوی پردازنده را معطل کند تا بتواند قبل از آنکه پردازنده به سراغ دستور بعدی برود، تبادل داده مورد نظر دستور فعلی با پردازنده را به پایان رساند. پین Ready برای وادار کردن پردازنده به طولانی تر کردن جریان کار عادی به کار می‌رود.

### پینه‌های INTR و NMI و INTA (۱۷ و ۱۸ و ۲۴)

این پینه‌ها مخصوص سرویس دادن به وقفه هستند. اجمالاً بدانید که واحدهای جانبی سیستم می‌توانند با ارسال سیگنال‌هایی به پینه‌های INTR و NMI پردازنده، باعث شوند پردازنده در کاری که مشغول انجام آن است وقفه ایجاد کرده و به واحد وقفه دهنده سرویس ارائه کند و سپس به ادامه کار قبلی خود بپردازد. پردازنده از طریق پین INTA، رسیدن سیگنال وقفه را تصدیق می‌کند.

### پین ورودی Test (23)

این پین برای هماهنگی سخت افزار و نرم افزار به کار می‌رود.

### پینه‌های خروجی آدرس/وضعیت (A19/S6 - A18/S5 - A17/S4 - A16/S3) (۳۵-۳۸)

این پینه‌ها در ابتدای اجرای دستور حاوی ۴ بیت بالای آدرس و در ادامه نشان دهنده وضعیت جاری سیستم از جمله سگمنت در حال استفاده و وضعیت پرچم وقفه می‌باشند. به کمک این پینه‌ها ارتباط مناسبی با تراشه‌های کمکی برقرار می‌شود. جزئیات را بعدها خواهید دید.

### پین خروجی SSO (۳۴) :

این پین به همراه پینه‌های IO/M و DT/R، برای نشان دادن حالت فعلی گذرگاه سیستم و تشخیص سیکلها (مانند سیکل خواندن حافظه، سیکل نوشتن در ورودی/خروجی و ...) به کار می‌رود.