جمهورية العراق
وزارة التربية
المديرية العامة للتعليم المهني

المعالجات الدقيقة

الحاسوب وتقنية المعلومات

تجميع وصيانة الحاسوب وشبكات الحاسوب

الصف الثالث

المؤلفون

الدكتور محمود زكي عبد الله

مصطفى صباح مصطفى　　سميرة ماجد مناتي　　سعاد حميد علي

اسراء عادل حيدر

إعداد

لجنة من المديرية العامة للتعليم المهني

الطبعة الثامنة　　1446هـ - 2024م

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

تمهيد

يعد الحاسوب الآلي السمة المميزة لعصرنا الحديث، نظراً للأهمية التي احتلها في شتى التطبيقات العلمية والتجارية، وما حققه من تقدم وتطور بسرعة مذهلة. لذا أصبحت الحاجة الى إستحداث أقسام وفروع علمية جديدة تواكب هذا التطور ورفدها بالمصادر العلمية والكتب المنهجية الحديثة أمراً ضرورياً يتماشى مع التطور الذي يشهده البلد في هذه المرحلة.

ومن هذا المنطلق شرعت المديرية العامة للتعليم المهني في وزارة التربية في العراق بإستحداث فروع وأقسام علمية جديدة مثل فرع الحاسوب وتقنية المعلومات بجميع أقسامه و تشكيل اللجان العلمية المعنية بوضع المناهج العلمية الحديثة لهذه الأقسام لتواكب التطور العلمي الحاصل في هذا المجال، ولتدريب ملاكات وطنية قادرة على شغل الوظائف التقانية والفنية والمهنية المتوافرة في سوق العمل في بلدنا.

يهدف هذا الكتاب الى تزويد الطالب المفاهيم الأساسية للتعرف على المعالج الدقيق. يتألف الكتاب من ستة فصول، يتناول الفصل الأول نبذة تعريفية عن المعالج الدقيق والحاسب الدقيق. أما الفصول من الثاني وحتى الخامس فهي تقدم شرحاً وافياً وميسرا عن أساسيات تنفيذ التعليمات في المعالج وطرائق انتقال المعلومات والأوامر وتركيب المعالج الدقيق 8080 و 8085، في حين يركز الفصل السادس على التعرف على أجيال المعالج الدقيق وتكنولوجيا التطور التي ساهمت في تحسين وتطوير المعالجات الدقيقة.

وفي الختام نرجو أن نكون قد وفقنا في عرض مضامين هذا الكتاب بالأسلوب السهل والميسر، ونتقدم بالشكر إلى جميع من أسهم في إنجاز هذا الكتاب، ومن الله التوفيق.

المؤلفون

المحتويات

الموضوع	رقم الصفحة
الفصل الأول: المعالج الدقيق والحاسب الدقيق	6
1-1 المقدمة	6
2-1 المخطط الكتلي للمعالج الدقيق	7
3-1 أجزاء المعالج الدقيق	8
أسئلة الفصل الأول	16
الفصل الثاني: تنفيذ التعليمات في المعالج	17
1-2 المقدمة	17
2-2 المعالج وتعليمات البرمجة	18
3-2 البنية الداخلية للمعالج وطريقة عمله	23
4-2 دورتا الآلــة والتعليمة	27
5-2 دورة التعليمة	28
6-2 العوامل المؤثرة في سرعــة المعالــج	32
أسئلة الفصل الثاني	39
الفصل الثالث: طرائق انتقال المعلومات	41
1-3 المقدمة	41
2-3 النواقل في الحاسوب	43
3-3 التخزين المؤقت في خطوط المعالج	48
4-3 طرائق انتقال المعلومات بين الأجزاء الداخلية للحاسوب	58

5-3 طرائق انتقال المعلومات بين الحاسوب والأجهزة الطرفية	65
أسئلة الفصل الثالث	72
الفصل الرابع: الأوامر	73
1-4 المقدمة	73
2-4 تصنيف الأوامر	74
3-4 العنونة	79
أسئلة الفصل الرابع	83
الفصل الخامس: المعالجان الدقيقان 8080 و 8085	84
1-5 المقدمة	84
2-5 المعالج الدقيق 8080	84
3-5 المعالج الدقيق 8085	90
أسئلة الفصل الخامس	94
الفصل السادس: أجيال المعالج الدقيق	95
1-6 المقدمة	95
2-6 المعالج الدقيق بنتيوم I	95
3-6 المعالج الدقيق بانتيوم II	96
4-6 المعالج الدقيق بنتيوم III	97
5-6 المعالج الدقيق بانتيوم IIII	98
6-6 تطور المعالجات	99
أسئلة الفصل السادس	106

الفصل الأول

المعـالج الدقـيـق

1-1 المقدمة

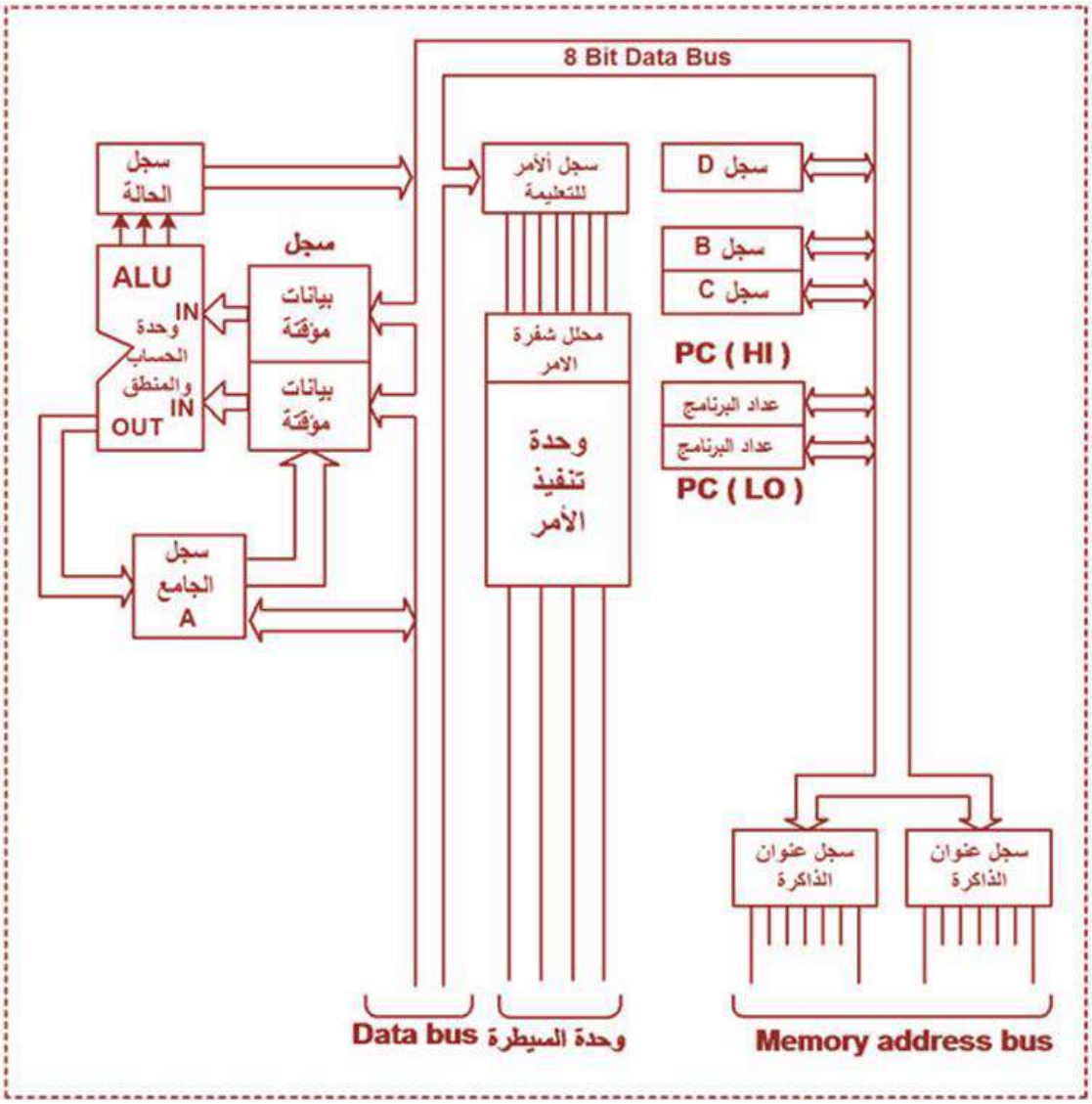
تعتبر الحواسيب من أهم مستجدات التكنولوجيا والتي أصبحت تأثيراتها تمتد إلى مختلف ميادين الحياة ومجالاتها، وتأتي أهميتها بالدرجة الأولى من قدرتها الكبيرة على تخزين المعلومات وإسترجاعها وكذلك السرعة الهائلة في معالجة البيانات والدقة المتناهية في تنفيذ الأوامر والعمليات. برزت أهمية الحواسيب بعد الزيادة الهائلة في حجم المعلومات مما أدى إلى ضرورة معالجة هذا الكم الضخم من البيانات ومتابعته وحفظه، لذلك كان لابد من إنتاج أجهزة متفوقة يكون بمقدورها القيام بهذه المهمات.

يتكون الحاسوب الدقيق (المايكروي) Microcomputer من عدة أجزاء مترابطة فيما بينها ليؤدي عمله بالشكل الصحيح والتي تتمثل بوحدة المعالجة المركزية والذاكرة الرئيسة وما يرتبط بها من وحدات الإدخال والإخراج. وتتألف وحدة المعالجة المركزية كذلك من جزئين رئيسيين هما المعالج الدقيق الذي يعد الجزء الرئيس والمهم من هذه الوحدة والذاكرة الرئيسية, وأن التطور الحاصل في مجال الحواسيب تركز بشكل رئيسي حول تطور المعالج الدقيق على مدى أجيال عديدة لما له من تأثير كبير في تطور الحواسيب بشكل عام. و يطلق أحيانا على وحدة المعالجة المركزية بالمعالج الدقيق والذي يرتبط مع الذاكرة الرئيسية وبقية الأجزاء على اللوحة الأم بطريقة تسهل عملية تناقل البيانات وإجراء العمليات عليها.

إن المعالج الدقيق عبارة عن شريحة أو رقاقة من السليكون مغلفة وموصلة باللوحة الأم بطريقة خاصة تقوم باستقبال البيانات من أجزاء الحاسوب الأخرى ومعالجتها، ثم إرسال النتائج إلى الأجزاء الأخرى لإخراجها أو تخزينها, ويتكون المعالج الدقيق بدوره من وحدتين رئيسيتين هما وحدة الحساب والمنطق ووحدة السيطرة المنطقية والتي سيأتي شرحها لاحقاً.

2-1 المخطط المعماري للمعالج الدقيق

يوضح الشكل (1-1) مخططاً تفصيلياً لمعالج دقيق، وهو عبارة عن وسيلة توضيحية تفسر كيفية عمل المعالج الدقيق وإرتباطه بالنواقل وتظهر أنواع السجلات (Registers) وكيفية ارتباطها بالأجزاء الأخرى داخل المعالج الدقيق، هذه السجلات تقوم بخزن البيانات داخل المعالج الدقيق بنحوٍ مؤقت، كما تقوم بالمساعدة في تنفيذ الأوامر.



الشكل 1-1 المخطط المعماري للمعالج الدقيق

3-1 أجزاء المعالج الدقيق

يتكون المعالج الدقيق بمختلف أنواعه من وحدتين رئيستين هما:-

1. وحدة الحساب والمنطق Arithmetic Logic Unit (ALU).
2. وحدة السيطرة المنطقية Control Unit (CU).

ويحتوي المعالج الدقيق في تركيبه المادي على مجموعة من السجلات والعدادات (Registers & Counters) التي تساعد في خزن البيانات داخل المعالج الدقيق بنحوٍ مؤقت.

يؤدي المعالج الدقيق وظيفتين أساسيتين هما:

- تنفيذ البرامج المخزونة في الذاكرة الرئيسة على وفق سياق أوامر وتعليمات البرامج وضبط المعدات لتؤدي الوظائف المطلوبة.
- إجراء العمليات الحسابية والمنطقية.

وترتبط بالمعالج الدقيق العديد من النواقل (Buses) التي تسهم في نقل البيانات والأوامر من المعالج وإليه, التي سيأتي شرحها لاحقا في الفصل الثالث بالتفصيل.

وتمر أغلب عمليات المعالجة التي يقوم بها المعالج الدقيق بعدة مراحل تسلسلية، تتضمن:

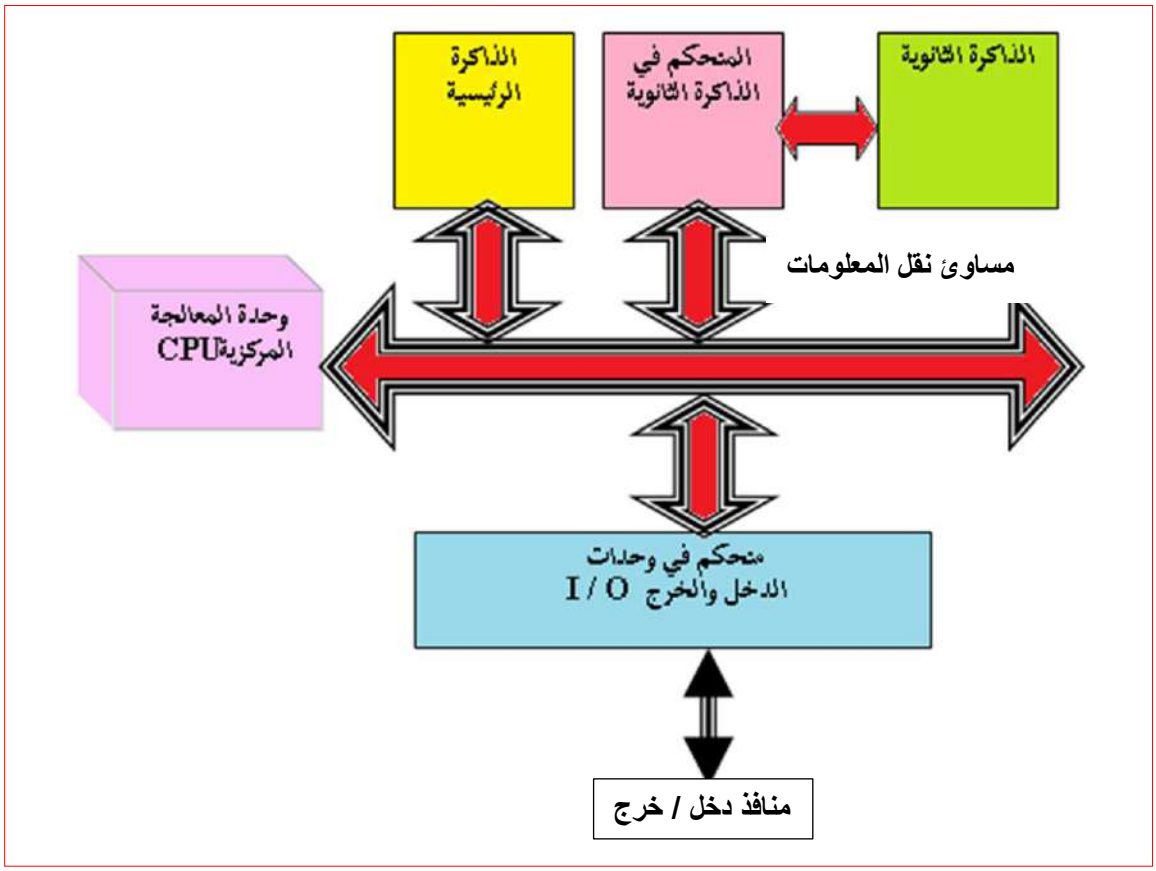
- قراءة (جلب Fetching) البيانات من ذاكرة البرنامج الرئيسة.
- تفسير البيانات (Decoding).
- عملية التنفيذ والإظهار (Executing).
- كتابة النتائج في الذاكرة (Saving or Storing) أو بقاؤها مرحلياً في وحدة المعالجة المركزية. كما لا بد من الذكر أن عملية السيطرة والتحكم في هذه العمليات تتم عن طريق وحدة السيطرة المنطقية الموجودة في المعالج الدقيق.

4-1 العلاقة بين وحدة المعالجة المركزية والذاكرة

يتم إنتقال البيانات بين الذاكرة ووحدة المعالجة المركزية عبر ناقل البيانات والعناوين، فطبع نسخة أو قراءتها من محتوى معين من البيانات مخزون في الذاكرة يتطلب جلب هذه البيانات المخزونة ونقلها إلى السجلات المناسبة في وحدة المعالجة عبر ناقل البيانات.

إن وحدة المعالجة المركزية تعمل على استخلاص البيانات أو أيعازات البرامج وقراءتها من الذاكرة بإرسال إشارة قراءة من وحدة السيطرة عبر ناقل التحكم تشمل إرسال عنوان خلية الذاكرة المطلوبة عبر ناقل العنوان من وحدة المعالجة المركزية إلى الذاكرة، وعلى المنوال نفسه

يمكن لوحدة المعالجة المركزية كتابة بيانات في خلايا الذاكرة. يمكن توضيح العلاقة بين وحدة المعالجة المركزية والذاكرة عن طريق المخطط المعماري الميسر الموضح بالشكل (1-2).



الشكل 1-2 توضيح العلاقة بين وحدة المعالجة المركزية والذاكرة

يمكن تعريف الذاكرة الرئيسية العشوائية RAM، بأنها عبارة عن شريحة أو رقاقة تقوم بتخزين البيانات أو البرامج المراد تنفيذها أو معالجتها عشوائيا بنحوٍ مؤقت، أما الذاكرة الثانوية أو وحدات الخزن الثانوية فهي وحدة تخزين مساعدة دائمة التخزين تستعمل ALU لخزن البرامج والبيانات بنحوٍ دائم لحين الحاجة اليها، حيث تستمر بالاحتفاظ بالبيانات والبرامج حتى في حالة إغلاق الحاسوب ، ومن الأمثلة على هذه الوحدات هي الأقراص الصلبة والمرنة والمدمجة المضغوطة.

1-3-1 وحدة الحساب والمنطق

هي أحدى المكونات الرئيسة لوحدة المعالجة المركزية، ويرمز لها بالرمز (ALU)، وهي مختصر لكلمات (Arithmetic and Logic Unit) ونعني بها وحدة الحساب والمنطق، وهذه الوحدة مسؤولة عن تنفيذ العمليات الحسابية والمنطقية التي يقوم بها المعالج الدقيق، إذ تتألف هذه الوحدة من مجموعة من الدوائر المنطقية الخاصة بذلك ومنها:

- دائرة الجامع التام التي تقوم بجمع ثلاث خانات ثنائية، ودائرة الجامع النصفي التي تقوم بجمع خانتين ثنائيتين.
- دائرة العكس (Invertors) المستعملة للحصول على المكمل الواحد أو الاثنين للرقم الثنائي.
- دائرة المركم (Accumulator) وهو مجموعة خلايا ثنائية تسمى بالسجل، وتستعمل عادة للاحتفاظ مؤقتاً بنتائج العمليات المنفذة إلى حين نقلها إلى الذاكرة أو إلى الوحدات الأخرى.
- سجل الحالة (Status Register) ويسمى بسجل المؤشرات وهو مجموعة من الخلايا الثنائية التي تبين حالة العملية المنفذة إذ تخصص كل خلية لمتابعة حالة معينة.

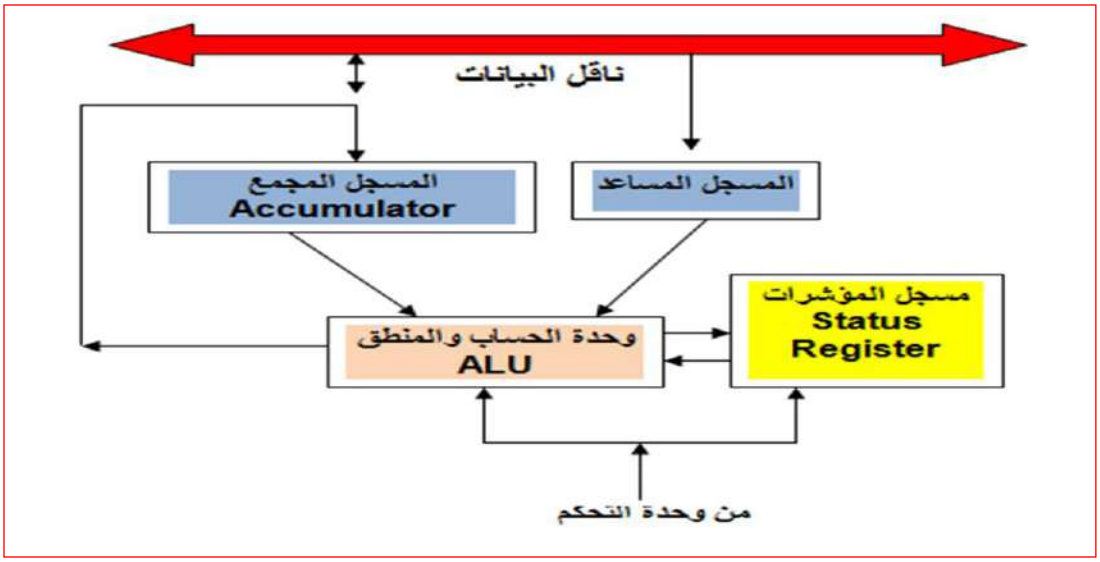
تحتوي وحدة الحساب والمنطق على وحدتين خاصة بالأعداد هما:-

أ- وحدة الأعداد الصحيحة: تقوم بمعالجة العمليات الحسابية التي تتكون من أعداد صحيحة لا تحتوي على فاصلة عشرية، تستعمل هذه العمليات في التطبيقيات البرمجية ثنائية الإبعاد مثل (Word و Power Point) ومعظم البرامج التي نستعملها، وبما أن هذه الوحدة تستعمل في تنفيذ التطبيقات البرمجية ثنائية الأبعاد، لذلك تعد مهمة جداً ووجودها ضروري في تطبيق هذه البرامجيات.

ب - وحدة الفاصلة العائمة (Floating Point Unit): تقوم بمعالجة العمليات الحسابية التي تحتوي الفاصلة العشرية، تستعمل هذه الوحدة في البرامج التي تعتمد على هذا النوع من العمليات الحسابية، مثل الألعاب، وبرامج التصميم الهندسي مثل (Auto CAD)، إذ أصبحت هذه الوحدة مهمة جداً نظراً لأن الألعاب الحديثة تعتمد في سرعتها على هذه الوحدة.

1-3-1-1 مبدأ عمل وحدة الحساب والمنطق

يبين الشكل (1- 3) وجود سجلين على مدخلي وحدة الحساب والمنطق يسمى أحدهما بالمركم أو المجمع (Accumulator)، والآخر بالسجل المساعد، تخزن فيهما القيم الواجب إجراء العمليات عليها ضمن تلك الوحدة، ونلاحظ أن خرج الوحدة يمكن أن يعود ويخزن في المركم أو يوضع على ناقل البيانات، كما يبين الشكل (1-3) وجود سجل المؤشرات (الحالة) الذي توضع فيه مؤشرات تصف خرج وحدة الحساب والمنطق، فمثلاً عند جمع قيمتين يوضع في أحد مواقع سجل الحالة قيمة تدل على كون الناتج صفراً، وفي موقع آخر للدلالة على إشارة الناتج واحد، أي كونه موجباً أو سالباً, كما سوف يتم شرحها لاحقاً بالتفصيل.



الشكل 1-3 عمل وحدة الحساب والمنطق في المعالج الدقيق

وتصنف عمليات الحساب والمنطق كالآتي:-

1- العمليات ذات المعامل الواحد، وتشمل:

- تصفير محتوى سجل ما (Clear).
- إيجاد المكمل (المعكوس) لمحتوى المركم.
- زيادة محتوى سجل ما بمقدار واحد (Increment).
- طرح واحد من محتوى سجل ما (Decrement).
- حركة محتوى السجل إلى اليسار أو اليمين.

2- العمليات ذات المعاملين، ومن أهمها:

- الجمع: جمع محتوى المركم مع محتوى سجل ما.
- الطرح: طرح محتوى سجل ما من محتوى المركم.
- المقارنة : إن النتيجة الحاصلة من العملية الحسابية لا تخزن في المركم، بل يخزن 1 أو إعتماداً على نتيجة المقارنة في خانة المقارنة.
- العملية المنطقية OR: إجراء عملية الجمع المنطقي بين محتوى المركم ومحتوى سجل ما إذ تخزن في المركم.
- العملية المنطقية AND: إجراء عملية الضرب المنطقي لمحتوى المركم وسجل ما وتخزين النتيجة في المركم.

2-3-1 وحدة السيطرة المنطقية Control Unit (CU):

يمكن تعريف وحدة السيطرة المنطقية بأنها وحدة الكترونية مؤلفة من مجموعة من الدوائر التي تتحكم بجميع العمليات المنفذة وتشرف على تسلسل تنفيذ التعليمات وتبادل البيانات بين وحدة الحساب والمنطق والذاكرة الرئيسية.

2-3-1-1 مكونات وحدة السيطرة المنطقية

تتكون وحدة السيطرة المنطقية بدورها من ثلاثة أجزاء هي: لاحظ الشكل (1-1).

1- **سجل (الأمر) التعليمة:** وهو أحد سجلات المعالج الدقيق تخزن فيه التعليمة التي سيقوم المعالج بتنفيذها ويكون طول هذا السجل بطول التعليمة.

2- **مفكك شفرة الأمر (التعليمة):** هو عبارة عن دائرة خاصة مهمتها ترجمة التعليمة التي تم جلبها الى المعالج عن طريق سجل التعليمة ثم تفسيرها.

3- **دائرة تنفيذ الأمر (التعليمة):** هي الوحدة المسؤولة عن تنفيذ الآمر بعد أن تم تشفيره وإعطاء الإشارات اللازمة الى كافة الوحدات لإكمال عملية تنفيذ التعليمة.

2-3-1-2 وظائف وحدة السيطرة المنطقية

1- قراءة تعليمات البرامج الموجودة في الذاكرة الأساسية وتفسيرها.

2- توجيه العمليات داخل المعالج.

3- التحكم في تدفق التعليمات والبيانات ومرورها من الذاكرة الرئيسية ومتحكمات ووحدات الإدخال والإخراج واليها.

3-3-1 السجلات

السجلات عبارة عن ذاكرة سريعة جداً ضمن المعالج الدقيق تستعمل لتوليد نتائج عمليات وحدة المعالجة المركزية والحسابات الأخرى وخزنها بصورة مؤقتة.

تعد السجلات بمنزلة أسرع وحدة في جهاز الحاسوب على الإطلاق، والسبب في ذلك إنها توجد على شريحة المعالج نفسها، وبذلك فانها تبادل البيانات بسرعته، وتختلف مجموعة السجلات من حاسوب إلى حاسوب آخر، وذلك بحسب بنية المعالج الدقيق، وتقسم السجلات على عدة أنواع بحسب وظيفتها الى:-

1- المركم Accumulator (A):-

وهو من أكثر سجلات المعالج عملاً، حيث تخزن فيه نواتج العمليات الحسابية والمنطقية, ويسمى أيضاً بسجل التراكم وذلك لتراكم نواتج العمليات فيه, وإن عدد الخانات الموجودة في المركم دائما تساوي عدد خطوط ناقل البيانات وقد تحتوي بعض المعالجات على أكثر من مركم واحد وذلك لزيادة سرعة تنفيذ التعليمات داخل المعالج.

2- عداد البرامج Program Counter (PC):-

وهو نوع من السجلات له وظيفة محددة, وهي تحديد عنوان التعليمة التالية الجاهزة للتنفيذ, فعند تنفيذ التعليمة تزداد قيمة هذا العداد ليؤشر إلى عنوان بداية التعليمة التالية, إذ توضع قيمته على ناقل العناوين لإيجاد التعليمة المطلوبة وجلبها, ويسمى أحيانا بمؤشر التعليمة.

3- سجل التعليمة Instruction Register (IR):-

هو أحد السجلات التابعة للمعالج الدقيق وتخزن فيه التعليمة التي يقوم المعالج بتنفيذها حالياً, ويكون طول هذا السجل بطول التعليمة.

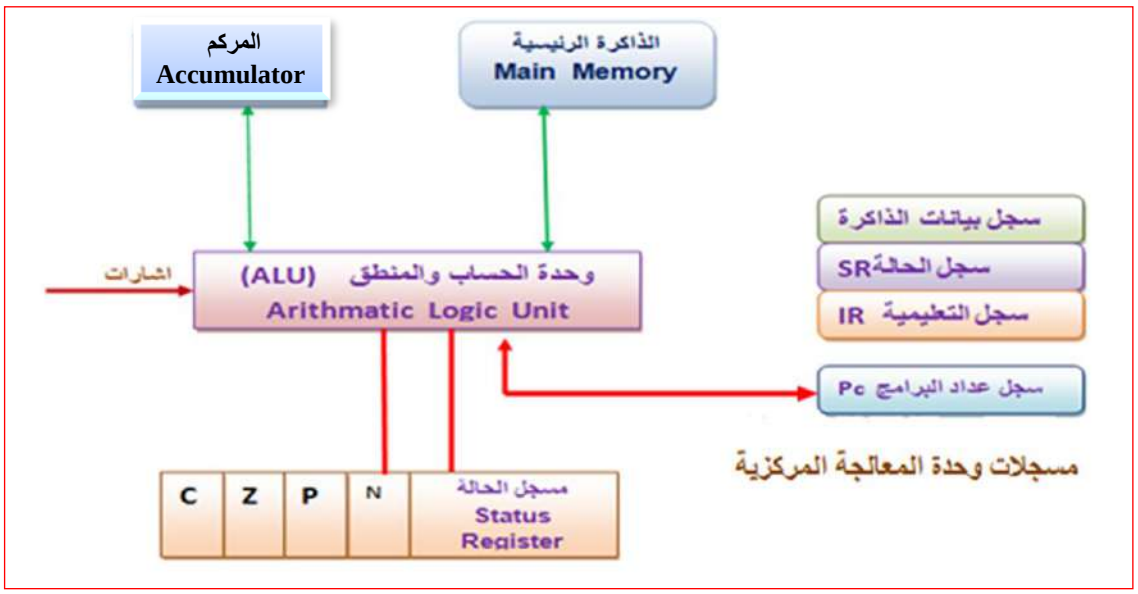
4- سجل الحالة Status Register (SR):-

يطلق عليه أيضاً إسم سجل الأعلام (Flag Register FR)، هذا السجل يحتوي على عدد من (bits)، وكل واحدة منها تعد علماً (Flag) تعكس أو تدل على حالة معينة من نتيجة العملية الحسابية أو المنطقية التي تم تنفيذها.

ومن هذه الأعلام:

- **علم الصفر Zero Flag ZF:-** هذه الخانة (بت) تكون واحدا إذا كانت نتيجة آخر عملية حسابية أو منطقية تساوي صفراً، وتكون هذه الخانة صفراً إذا كانت النتيجة مختلفة عن الصفر سواء موجبة أم سالبة.
- **علم الإشارة Sign Flag SF:-** هذه الخانة تكون واحدا إذا كانت نتيجة آخر عملية حسابية أو منطقية نفذها المعالج سالبة، أما إذا كانت هذه النتيجة موجبة فإن هذا العلم يكون صفراً.
- **علم الحمل Carry Flag CF:-** هذا العلم يكون واحداً إذا حصل حمل Carry من آخر خانة (بت) في أي عملية جمع أو حصل إستلاف Borrow لآخر خانة (بت).

لاحظ الشكل (1-4) الذي يبين آلية عمل سجل الحالة.



الشكل 1-4 آلية عمل سجل الحالة

5- سجلات عامة الأغراض General Purpose Registers:-

تم تجهيز المعالج ببعض السجلات التي تستعمل لتخزين النتائج المرحلية الخاصة بالعمليات الحسابية لحين الحاجة إليها دون الرجوع الى الذاكرة وذلك لزيادة سرعة التنفيذ للعمليات الحسابية، و يكون عدد الخانات (bits) في هذه السجلات عادة مساوياً لعدد خانات ناقل البيانات، تسمى بالسجلات (B,C,D,E,H,L) ويمكن استخدامها بصورة مفردة بطول 8 بت أو بصورة مزدوجة بطول 16 بت (BC, DE ,HL) وعدد هذه السجلات يختلف من معالج إلى معالج آخر.

6- سجل البيانات المؤقتة Temporary Register:-

هو أحد السجلات الخاصة بالمعالج الدقيق ويقوم بخزن البيانات مؤقتاً قبل الدخول الى وحدة الحساب والمنطق وذلك لعدم وجود سجلات داخل هذه الوحدة, و يكون عدد الخانات (bits) في هذه السجلات عادة مساوياً لعدد خانات ناقل البيانات.

7- سجل عنوان الذاكرة Memory Address Register (MAR):-

عندما نريد كتابة معلومات في موقع محدد من الذاكرة أو قراءتها منه ينتقل عنوان الموقع المحدد إلى سجل عنوان الذاكرة MAR, لينتقل منه بعد ذلك إلى مفكك ترميز (ناخب) الذي يقوم عندئذٍ بانتخاب خط العنوان المناسب، وبانتخاب هذا الخط تفعل كل عناصر التخزين الموجودة على هذا الخط, وعدد الخانات في هذا السجل مساوياً لعدد خانات ناقل العناوين.

أسئلة الفصل الأول

س1: إملأ الفراغات الآتية :-

1- .....تتحكم بتيسير المعلومات من المعالج الدقيق وإليه.

2- يتألف المعالج الدقيق من وحدتين أساسيتين هما 1 .....2 .....

3- .....هي ذاكرة صغيرة جداً تستعمل لتخزين نتائج عمليات وحدة المعالجة المركزية مؤقتاً.

4- .....هو عبارة عن دائرة خاصة مهمتها فك شفرة التعليمة وتحويلها الى إشارات مفهومة داخل المعالج ليتم تنفيذها.

5- الذاكرة هي .....

6- يسمى عداد البرنامج في بعض أنواع الحواسيب بـ .....

س2: ما المقصود بالمعالج الدقيق؟

س3: وضح عمل المعالج والخطوات لتنفيذ الأوامر؟

س4: إذكر أجزاء المعالج الدقيق؟ وما وظيفة كل جزء؟

س5: ما العلاقة بين وحدة المعالجة المركزية والذاكرة؟

س 6: عدد وظائف وحدة السيطرة (CU)؟

س 7: ما مكونات وحدة السيطرة؟

س 8: وضح عمل كل من (المركم، سجل عداد البرامج، سجل التعليمة IR)؟

س9: ما سجل الحالة SR؟ وما الحالات التي يسجلها؟

س10: ما هي السجلات عامة الأغراض؟

س11: ما الغرض من استعمال كل من :

1. مفكك شفرة التعليمة.
2. السجلات.
3. سجل البيانات المؤقتة.

س12: عدد خمسة أنواع من السجلات وإشرح واحداً منها؟

الفصل الثاني

تنفيـذ التعليمـات في المعالـج

1-2 المقدمة

بعد دراسة هيكلية المعالج الدقيق وبنيته في الفصل الأول، سندرس في هذا الفصل كيف يقوم المعالج بتنفيذ البرامج وما تتضمنه من تعليمات مما يؤدي إلى تشغيل الحاسوب بسلاسة وكفاءة، وللقيام بذلك يجب علينا التركيز على المعالج وفي الوقت نفسه إلقاء نظرة عامة على وحدات الحاسوب كاملة في أثناء عملية تنفيذ البرامج.

فعلى سبيل المثال عندما يتم تثبيت البرمجيات على جهاز الحاسوب الشخصي الحديث وعادة يتم ذلك عن طريق الوسائل الأكثر شيوعًا مثل الأقراص الليزرية CD-ROM، أو تحميله عن طريق الإنترنت، فيتم تخزين الشفرة (Code) التي يتألف منها البرنامج وأية ملفات مرتبطة على القرص الصلب، وهذه الشفرة تتألف من:

سلسلة تعليمات تقوم بأداء مهام معينة، وكذلك البيانات المرتبطة بهذه التعليمات، تبقى الشفرة هناك في القرص الصلب حتى يقوم المستعمل بطلب تنفيذ البرنامج وحينئذ جزء من هذه الشفرة يتم تحميله إلى ذاكرة الحاسوب.

بعد ذلك يقوم المعالج بتنفيذ البرنامج من الذاكرة وذلك بمعالجة كل تعليمة (إيعاز) بدورها، ولتنفيذ التعليمات من الضروري للمعالج أن يفهم ما التعليمة وما عملها, لذلك فإن هناك حاجة إلى تمييز التعليمات التي سيتعامل معها المعالج، ويتم ذلك عن طريق برمجتها إلى المعالج، إنَّ التعليمات التي يمكن للمعالج تمييزها تُعرف بمجموعة التعليمات (Instruction Set) التي سندرسها تفصيليًا في الفصل الرابع وعندما يتم التعرف على التعليمة وتقرير الإجراءات التي ينبغي تنفيذها عليها حينئذ سيقوم المعالج بتنفيذ هذه الإجراءات قبل الانتقال إلى التعليمة التالية في الذاكرة، وتسمى هذه العملية بـ "دورة تنفيذ التعليمات" (Instruction Execution Cycle) التي سنتناولها بالتفصيل في هذا الفصل، إن النتائج التي يتم الحصول عليها مما سبق سيتم تخزينها مرة أخرى في الذاكرة ثم حفظها على القرص الصلب في وقت لاحق، وربما احتياطيا على وسائط الخزن المتنقلة أو في مواقع منفصلة، وهذا هو تدفق المعلومات نفسه كما هي الحال عندما يتم تنفيذ البرنامج الوحيد ولكن بالاتجاه المعاكس، كما موضح في الشكل (2-1).



شكل 2-1 سير البيانات في أثناء عملية تنفيذ البرنامج وخزن البيانات

إن تفصيلات ما يحدث في أثناء عملية تنفيذ البرنامج هو ما سندرسه في هذا الفصل وهذا يتطلب معرفة ما (التعليمة أو الإيعاز) التي يتكون منها البرنامج، وأي لغة برمجة سيتم اعتمادها لكتابة التعليمات التي يتعامل معها المعالج، ومن ثم ما المراحل التي ستمر بها التعليمة ليتم تنفيذها، وفي الختام نتطرق لذكر أهم العوامل المؤثرة في سرعة المعالج لأداء مهامه.

2-2 المعالج وتعليمات البرمجة

من دراستنا السابقة في الفصل الأول علمنا أن الوظيفة الرئيسية للمعالج الدقيق هو معالجة وتنفيذ مجموعة من التعليمات المحددة بحيث يتم تنفيذ كل تعليمة عند إعطاء الشفرة الخاصة بها،

وكلما تعددت هذه التعليمات وزادت سرعة المعالج في تنفيذها وكذلك سهولة عمليات المواجهة مع الدوائر المحيطة بالمعالج كلما كان ذلك مؤشرًا على جودة المعالج، وقبل التطرق إلى المفاهيم الأساسية لتنفيذ التعليمات في المعالج لا بُدَّ من بيان أنَّ التعامل مع المعالج يتم عادة بوسيلة من اثنتين لا غنى لواحدة منهما عن الأخرى، وهما:

1. **المكونات البرمجية Software** وعادة ما يتم ذلك باعتماد البرمجة بلغة الآلة Machine Language الخاصة بالمعالج الذي يتم التعامل معه، إذ إن كل معالج له لغة آلـة خاصة به، مع التنبيه إلى أنَّ كل معالج يستطيع تنفيذ تعليمات المعالج الذي قبله ويملك مجموعة تعليمات موسعة وإضافية لا تستطيع المعالجات التي قبله تنفيذها، وأما المعالج الذي يأتي بعد هذا المعالج فأنه يدعم التعليمات الموسعة للمعالج الذي قبله في حين يحتوي هو أيضاً على تعليمات جديدة وموسعة.
2. **المكونات المادية Hardware** وتشتمل هذه الوسيلة على كيفية توصيل المعالج على الأطراف المحيطة به، مثل: (الذاكرة، ووحدات الإدخال والإخراج) ، وإستعمال المعالج في التطبيقات المختلفة، مثل (دوائر التحكم).

والمتعامل مع المعالج لا بُدَّ أن يكون ملمًا بكلتا الوسيلتين السابقتين: البرمجة، والمكونات المادية، وإن إختلفت نسبة إلمامه بأي واحدة منهما فهذا يؤدي إلى تحديد إستعمالاته للمعالج بطريقة أو أخرى، فمثلا الشخص الذي يرغب في تصميم وإعداد برامج الصيانة أو تصميم برامج التعريب أو إعداد برامج التعريف الخاصة بأجهزة الحاسوب مثل (كروت الشاشة أو برامج تعريف الطابعات وغيرها) يحتاج أن يكون ملمًا بلغة تتعامل مباشرة مع الأجهزة المادية للحاسوب مثل لغة الآلـة ولغة الاسمبلي, أمَّا في حالة كون المتعامل مع المعالج غرضه البرمجة فيمكنه تحقيق هذا الغرض بواسطة أي واحدة من لغات البرمجة المعروفة التي يكون تعلمها أسهل بكثير من تعلم برمجة المعالج بلغة الآلة أو لغة الاسمبلي التي تتميز بأنها لغات يمكن تفسيرها مباشرة بواسطة الأجهزة (Hardware) نفسها.

وكما اسلفنا سابقاً فأنَّ من المهام الرئيسة للمعالج هي تنفيذه لبرنامج ما، والبرنامج هو عبارة عن مجموعة من التعليمات المتسلسلة المرتبة منطقياً مكتوبة بلغة برمجة معينة تقوم بتوجيه المعالج لأداء وظيفة ما أو عمل معين مثل: التحكم في متغير معين، أو التعرف على معلومة معينة من بين الكثير من المعلومات، يجب أن يحتوي البرنامج أو يحدد مصدر أي معلومة يتم اعتمادها بواسطته، فمثلًا لإجراء عملية جمع لرقمين فلا بد من أن يحدد البرنامج أين يوجد الرقمان وأين ستوضع

النتيجة، يمكن النظر لأي برنامج على أنَّه مجموعة من الشفرات الثنائية المخزنة في الذاكرة في انتظار أن يقوم المعالج بتنفيذها.

أما التعليمة (الإيعاز) المكونة للبرنامج فهي الشفرة الثنائية (Binary Code) التي تُعطى للمعالج وعلى أثرها يقوم المعالج بتنفيذ عمل معين، مثل جمع رقمين أو إحضار معلومة من الذاكرة أو غير ذلك من الأفعال التي يستطيع المعالج القيام بها.

ومن أمثلة هذه الشفرات الثنائية الشفرة الآتية: (10000000) التي معناها اجمع محتويات السجل B مع سجل التراكم A، وضع النتيجة في السجل A، كما نرى يستطيع المعالج التعرف على الشفرة السابقة، لأنها شفرة ثنائية في حين أنَّه لن يستطيع التعرف على أي نوع آخر من الشفرات سواء كانت حرفية أم ثمانية أم ست عشرية.

إنَّ لغة البرمجة التي توظف لكتابة التعليمات المكونة للبرنامج الذي سيتم تنفيذه بواسطة المعالج، هي لغة متكونة من الواحدات والأصفار التي يتم إرسالها بسهولة عبر إشارات كهربائية ليتعامل معها المعالج، وتسمى بلغة الآلــة Machine Language، وهي اللغة الوحيدة التي يفهمها الحاسوب ويوظفها للاتصال بين الوحدات الداخلية المختلفة وتعد لذلك اللغة الداخلية له، وإعتمدت لغة الآلة في كتابة برمجيات الجيل الأول من الحواسيب، ويتكون البرنامج بهذه اللغة من سلسلة طويلة من الأرقام المكونة من الصفر والواحد الأمر الذي يتطلب من المبرمج وقتًا وجهدًا كبيرين في كتابته فضلاً عن معرفته بدقائق المعالج المستعمل، إذ تختلف هذه اللغة من معالج إلى معالج آخر تبعاً لنوعه وطرازه وتركيبه الداخلي، والتعليمات والبرامج المكتوبة بهذه اللغة يمكن للمعالج أن ينفذها بسرعة عالية جدًا، إلا أنَّ برمجة الحواسيب بالنظام الثنائي مباشرة عملية شاقة يصعب على الإنسان القيام بها بسبب العيوب التي تتضمنها، ومنها:

1. البرامج المكتوبة بلغة الآلـة تأخذ وقتًا طويلًا في إدخالها للذاكرة, لأنها تكتب بتاً بعد بت.
2. صعوبة فهم أي خطأ أو متابعته أو تصحيحه في البرامج المكتوبة بلغة الآلـة، لأن الأعداد الثنائية عبارة عن نماذج من الواحدات والأصفار التي يصعب التفريق بينها ولاسيما بعد مدة عمل طويلة مع هذه الأرقام.
3. شكل البرنامج لا يعطي أي دلالة على الغرض منه، بخلاف البرامج المكتوبة بلغات البرمجة الأخرى سواء اللغات منخفضة المستوى (Low Level Language) مثل لغة التجميع المعروفة بالاسمبلي (Assembly) أم لغات عالية المستوى (High Level Language) مثل لغة البيسك (Basic)، لأنه بعد نظرة فاحصة على البرنامج يمكن معرفة الغرض منه.

4. من السهل أن يقع المبرمج في الكثير من الأخطاء في أثناء كتابة البرامج المكتوبة بلغة الآلة وفي الوقت نفسه من الصعب جدًا على المبرمج أن يقوم باستخراج هذه الأخطاء في ما بعد.

حاول المبرمجون التغلب على بعض هذه العيوب عن طريق اعتماد النظام الست عشري (Hexadecimal System) بدلًا عن النظام الثنائي (Binary System) كطريقة أكثر فاعلية في تمثيل لغة الآلة، فقد حل النظام الست عشري مشكلة صعوبة كتابة البرامج واستخراج الأخطاء منها إلى حد ما, ولكن بقيت المشكلة الأخرى وهي استمرار التعامل مع أرقام صماء كشفرات للأوامر لا تحمل أي دلالة عن ماذا يفعل هذا الأمر أو ذاك، فمثلًا الرقم (3A) هو شفرة لأمر معين؛ ولكن لا يمكن تمييز ماهية هذا الأمر، فقد يكون الرقم جزءًا من عنوان كالأرقام (60, 61, 62) وغيرها، وفي حالة معرفة أنَّه شفرة لأمر فستبقى إمكانية معرفة ماذا يفعل هذا الأمر معدومة إلا بالعودة إلى الدليل الخاص بذلك, ولأجل ذلك عمل المبرمجون على تطوير فكرة يتم عن طريقها فهم المقصود من كل أمر من الأوامر بحيث يتم إعطاء كل أمر شفرة مكونة من ثلاثة أو أربعة أحرف في الأكثر على أن تكون هذه الأحرف مختصرة من الحروف الأبجدية التي تدل تقريبًا على ما يقوم به المعالج عند تنفيذ الأمر، فمثلًا أمر الجمع يكون ADD التي هي اختصار لكلمة جمع (Addition)، وأمر الطرح يكون SUB الذي يمثل اختصارًا لكلمة طرح (Subtraction)، وهكذا مع بقية الأوامر.

إنَّ هذه الاختصارات هي ما يسمى بلغة التجميع (Assembly Language) أو أحيانا تسمى بالشفرات الحرفية (Mnemonics Codes)، وهي تعني الشفرات التي يمكن تذكرها، وتسمى أيضًا بلغة الترميز، فهي لغة تعتمد شفرات رمزية سهلة التذكر بدلًا من الشفرات الرقمية التي كانت معتمدة في لغة الآلة، ومن هذه الشفرات التي تعتمدها اللغة: (MOV, ADD) وبوضع التعليمات بهذه الصورة صار من السهل تذكرها ومعرفة وظيفتها بمجرد النظر إليها، ومن الجدير بالذكر بأنَّ كل نوع من أنواع المعالجات يتم تزويده بدليل يحتوي على الاختصارات الحرفية (Mnemonics) وغالبًا ما تكون اختصارات كل شركة منتجة للمعالجات تختلف عن اختصارات الشركات الأخرى.

إنَّ لغة التجميع تشابه لغة الآلة، لأن اللغتين من اللغات منخفضة المستوى (Low Level Language)، وهذا لا يعني أنها لغات ضعيفة، وإنما يعني أنها مرتبطة بالمعالج مباشرة، فالمتعامل مع هذه اللغات لا بُد من أن يكون ملمًا بمكونات المعالج ليستطيع استخدام هذه اللغة بكفاءة، فلغة التجميع لغة يحتاج إليها كل مبرمج محترف لبناء برامج قوية مثل برامج

التطبيقات الخدمية، وبرامج كشف الاختراقات والقيام بها، لذا نجد أن المخترقين (Hackers) يكونون عادة محترفين في التعامل مع هذه اللغة.

وبما أنَّ المعالج لا يتعامل إلا مع الشفرات الثنائية كان من الضروري ترجمة البرنامج المكتوب بلغة التجميع ذي الشفرات الحرفية إلى برنامج ذي شفرات ثنائية، ويقوم بهذا العمل برنامج خاص يسمى بالمجمع (Assembler)، وهو عبارة عن برنامج خاص يقوم بتحويل البرنامج المكتوب بلغة التجميع (الشفرات الحرفية) إلى برنامج بلغة الآلة (شفرات ثنائية)، يُطلق على البرنامج المكتوب بلغة التجميع برنامج المصدر (Source Program)، والبرنامج المكتوب بلغة الآلة يسمى برنامج الهدف (Object Program)، والفترة المستغرقة في التحويل إلى برنامج الهدف تعد من أهم العيوب لهذه اللغة مقارنة مع لغة الآلة، وهذه المدة تعتمد على زمن تنفيذ المعالج لمجموعة التعليمات المكتوبة التي يحتوي عليها البرنامج.

تتكون التعليمات في لغة التجميع من: الاسم الرمزي (Operation Code) الذي قد يكون متبوعًا بمعامل واحد أو معاملين (Operands)، والمعاملات هي المعطيات التي سيتم معالجتها، أمَّا الاسم الرمزي، فهو الأمر الذي سيوجَّه إلى المعالج لإخباره عن العملية التي ستُجرى على هذه المعطيات, وسيتم شرح التعليمات (الأوامر) ومكوناتها في التفصيل في الفصول القادمة.

Opcode (الشفرة)	Operand (s) (المعامل)

كل عملية لها رمز يميزها عن غيرها، أمَّا المعاملات، فهي تشير إلى البيانات المطلوبة لتنفيذ العملية أين يمكن إيجادها والوصول إليها، وتكون أما بيانات (Data)، وإما عناوين في الذاكرة (Memory Address).

ولتوضيح ذلك بنحوٍ عام سنستعرض تعليمتين مشهورتي الاستعمال في لغة التجميع على سبيل المثال:

التعليمة الأولى : تعليمة النقل MOV

تستعمل هذه التعليمة لنسخ المعطيات الموجودة في معامل المصدر(Source Operand) إلى معامل الهدف (Destination Operand) وتأخذ الصيغة الآتية:

MOV destination , source

تخبر هذه التعليمة المعالج بأن يقوم بنقل (وفي الحقيقة نسخ) معامل المصدر إلى معامل الهدف، فعلى سبيل المثال تنسخ التعليمة MOV D, C محتويات السجل C إلى السجل D بحيث

ستصبح محتويات السجل D مطابقة لمحتويات السجل C بعد تنفيذ العملية، يقوم البرنامج التالي بتحميل القيمة 55 H في السجل C ومن ثم ينسخها إلى سجلات مختلفة أخرى في المعالج.

```
MOV    C, 55 H        ;ضع القيمة 55 H في السجل C
MOV    D, C           ;(D=C=55 H) انسـخ محتويات السجل C إلى السجل D
MOV    A, D           ;(A=D=55 H) انسخ محتويات السجل D إلى السجل A
MOV    B, C           ;(B=C=55 H) انسخ محتويات السجل C إلى السجل B
```

التعليمة الثانيـة: تعليمة الجمع ADD

تستعمل هذه التعليمة لجمع البيانات الموجـــودة في معامـــل الهـــدف (Destination Operand) مع البيانات الموجودة في معامل المصدر (Source Operand) ومن ثم تخزن في معامل الهدف وتأخذ الصيغة الآتية:

ADD destination, source

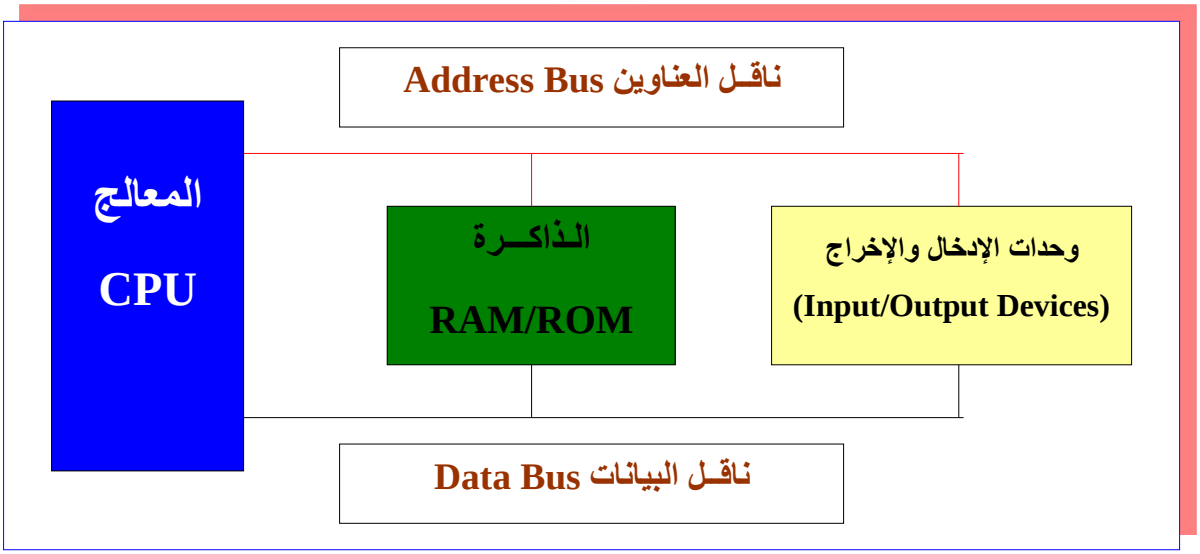
تخبر هذه التعليمة المعالج بأن يقوم بجمع معاملي المصدر والهدف ووضع النتيجة في معامل الهدف، فعلى سبيل المثال للقيام بجمع عددين مثل 25 H و 34 H بإمكاننا أن نضع هذه القيم في السجلات، ومن ثم نجمعها بعضها مع بعض كالآتي:

```
MOV    A, 25 H        ;ضع القيمة 25 H في السجل A
MOV    B, 34 H        ;ضع القيمة 34 H في السجل B
ADD    A, B           ; A=A+B
```

بعد أن إستعرضنا بنحوٍ عام ما التعليمة وطريقة كتابتها واللغة المعتمدة لذلك، لم يبقَ لنا إلا معرفة كيف يتم تنفيذ هذه التعليمة داخل المعالج، وهو ما سنتطرق له في الفقرة الآتية.

3-2 البنية الداخلية للمعالج وطريقة عمله

إنَّ المعالج الدقيق جزء مهم في الحاسوب، ولفهم عمله بنحوٍ واضح لا بُد من فهم بنيته الداخلية وعلاقته مع البنية الأساسية للحاسوب، ودرسنا سابقًا أنَّ البنية الأساسية للحاسوب تقسم على ثلاثة أجزاء: وحدة المعالجة المركزية (Central Processing Unit – CPU) أو ما يُعرف إختصاراً بالمعالج، والذاكرة بأنواعها (Memory) وأجهزة الإدخـــال والإخـــراج المتنوعة (Input / Output Devices) كما موضح في الشكل (2-2).



شكل 2-2 مخطط معماري لأجزاء الحاسوب الرئيسة

ينفذ المعالج البرنامج المخزون في الذاكرة وتؤمن وحدات الإدخال والإخراج مثل: لوحة المفاتيح والشاشة وسائل اتصال مع المعالج، يتصل المعالج مع وحدات الذاكرة والإدخال والإخراج عبر مجموعة من الأسلاك تُعرف بالنواقل (Buses).

يتعرف المعالج على وحدات الإدخال والإخراج والذاكرة بتخصيص عناوين محددة لها، ولابُدَّ للعنوان المخصص لجهاز ما من أن يكون وحيداً، فلا يسمح بعنونة جهازين بالعنوان نفسه، يوضع العنوان على ناقل العناوين (بالنظام الثنائي) ثم تقوم دائرات تفكيك الشفرة (Binary Decoder) بتحديد الجهاز المقصود، وبعد ذلك يتبادل المعالج البيانات مع الأجهزة عن طريق ناقل البيانات، ويحدد ناقل التحكم رغبة المعالج بإرسال المعلومات إلى الجهاز أو العكس، وذلك عن طريق إشارات القراءة والكتابة (Read/Write)، وتحدد نواقل البيانات والعناوين إمكانيات المعالج وقدراته.

والآن لتوضيح عمل المعالج داخليًا سنقوم بجمع ثلاثة أعداد وتحليل عمل المعالج خطوة خطوة لإنجاز هذه العملية. نفترض أنَّه لدينا معالجاً ما (افتراضي) يحتوي على:

1. السجلات الآتية: (A, B, C ,D).
2. ناقل عناوين بعرض (16 Bit)، لذا سيكون عدد مواقع الذاكرة المتاحة (2^{16} = 64 k) ومجال العنونة سيكون من 0000 إلى FFFFH أي إنَّ (مجموع المواقع FFFFH+1=10000H موقعًا).

3. موقع خزن البرنامج المطلوب يبدأ من الموقع 1400H.
4. قيم الأعداد الثلاثة المطلوب جمعها هي: 21H, 12H, 42H.

العمل سيكون كالآتي: يقوم المعالج بتخزين القيمة 21H في السجل A ثم تجمع معها القيمة 12H والقيمة 42H ويضع الناتج في السجل A، لنفرض أنَّ شفرة نقل قيمة إلى السجل A هي (B0H) 10110000 وشفرة نقل جمع قيمة إلى السجل A هي (04H) 00000100، الجدول التالي يوضح الخطوات المهمة والشفرة المقابلة لتنفيذ هذا العمل:

الشفـــرة	البيانـات	الشـــرح
B0H	21H	نقـل القيمة 21H إلى السجل A
04H	42H	جمع القيمة 42H إلى السجل A
04H	12H	جمع القيمة 12H إلى السجل A

وبما أنَّ البرنامج مخزون في الذاكرة بدءاً من الموقع 1400H، يظهر الجدول التالي محتويات كل موقع في الذاكرة كالآتي:

عنوان الذاكرة	محتويات العنوان	الشـــرح
1400	B0	شفرة نقـل القيمة إلى السجل A
1401	21	القيمة المنقولـة
1402	04	شفرة جمع قيمة مع السجل A
1403	42	القيمة المضافـة
1404	04	شفرة جمع قيمة مع السجل A
1405	12	القيمة المضافـة
1406	F4	شفرة التوقـف

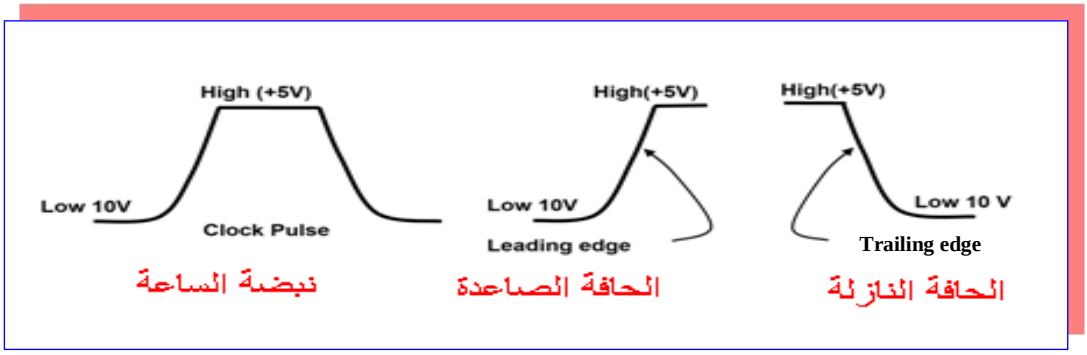
أمَّا خطوات تنفيذ المعالج لهذا البرنامج فهي كالآتي:

1. يمكن لعداد البرنامج أن يأخذ أي قيمة في المجال 0000 – FFFFH، وبما أن عنوان بداية البرنامج هو 1400H، لذا يجب تخزين تلك القيمة في عداد البرنامج ليشير إلى شفرة أول تعليمة من البرنامج، وبعد ذلك يكون المعالج جاهزاً لتنفيذ البرنامج.
2. يضع المعالج هذا العنوان على ناقل العناوين، وتقوم دائرة تفكيك الشفرة داخل الذاكرة بتحديد موقع الذاكرة، في حين يُفعِّل المعالج إشارة القراءة READ مشيراً بذلك إلى الذاكرة أنه يطلب البايت الموجود في الموقع 1400H، وبذلك يتم وضع محتوى الموقع المذكور B0 على ناقل البيانات ويجلب إلى المعالج.
3. يقوم المعالج بتفكيك شفرة B0 بمساعدة قاموس مفكك شفرة التعليمات الذي يُعْلِم المعالج بأنَّ هناك بايتًا في الموقع التالي يجب جلبه ووضعه في السجل A، ويعطي الأمر إلى السجلات للقيام بذلك. عند جلب القيمة 21H من الموقع 1400H يتأكد المعالج من أنَّ جميع أبواب السجلات مغلقة عدا السجل A، ولذلك عند وصول القيمة 21H إلى المعالج ستذهب مباشرة إلى السجل A حصرًا، وبعد الانتهاء من تنفيذ هذه التعليمة يشير عداد البرنامج إلى عنوان التعليمة التالية الواجب تنفيذها الذي يمثل في المثال العنوان 1402H، إذ يرسل العنوان 1402H على ناقل العناوين لجلب التعليمة التالية.
4. يجلب المعالج الشفرة 04H من العنوان 1402H، ويفهم بواسطة مفكك شفرة التعليمة أنَّها تمثل عملية جمع قيمة مع محتوى السجل A، وبالتالي يجلب القيمة من العنوان التالي 1403H ويجمعها مع السجل A، وبعد جلب القيمة 42H من ذلك الموقع يقدمها مع محتوى السجل A إلى وحدة الحساب والمنطق لإنجاز عملية الجمع، ويتلقى المعالج بعد ذلك ناتج الجمع من مخرجات وحدة الحساب والمنطق فيضعها في السجل A بينما يحمل عداد البرنامج القيمة 1404H التي تمثل عنوان التعليمة التي تليها مباشرة.
5. يوضع العنوان 1404H على ناقل العناوين وتجلب الشفرة إلى المعالج ويتم تفكيكها وتنفيذها كما مطلوب بالخطوات السابقة نفسها، وهذه الشفرة أيضًا تجمع قيمة مع السجل A، ثم يعدَّل عداد البرنامج إلى العنوان 1406H.
6. وفي النهاية يجلب المعالج محتوى العنوان 1406H وينفذه، إذ تشير الشفرة F4 إلى التعليمة HALT التي تخبر المعالج بوقف زيادة عداد البرنامج والاستفسار عن التعليمة التالية.

وفي حالة عدم وجود التعليمة HALT سيتابع المعالج تعديل قيمة عداد البرنامج وجلب التعليمات.

4-2 دورتا الآلة والتعليمة Machine and Instruction Cycle

يوجد في داخل كل حاسوب دائرة خاصة تسمى بمؤقت النظام (Clock System)، وهذه الدائرة ليست وظيفتها معرفة الوقت، وإنما لإرسال نبضات كهربائية صغيرة إلى المعالج الذي بدوره يقوم بتوظيف هذه النبضات للتحكم في العمليات التي ينجزها. فإذا كان المعالج هو بمنزلة القلب لجهاز الحاسوب فإنَّ مؤقت النظام هو بمنزلة جهاز تنظيم ضربات القلب (Pacemaker)، وكما لا يخفى أنَّ ضربات القلب المنتظمة مطلوبة لبقاء الكائن الحي على قيد الحياة، فإنَّ نبضات التوقيت (CLK) مطلوبة للعمليات المختلفة التي يتم إجراؤها في المعالج، فكل العمليات التي يجريها المعالج يتم التحكم بها إما بواسطة الحافة الصاعدة (Leading Edge)، أو الحافة النازلة (Trailing Edge) لنبضة النظام، والشكل (2-3) يوضح نبضة النظام.



شكل 2-3 نبضة النظام (للإطلاع)

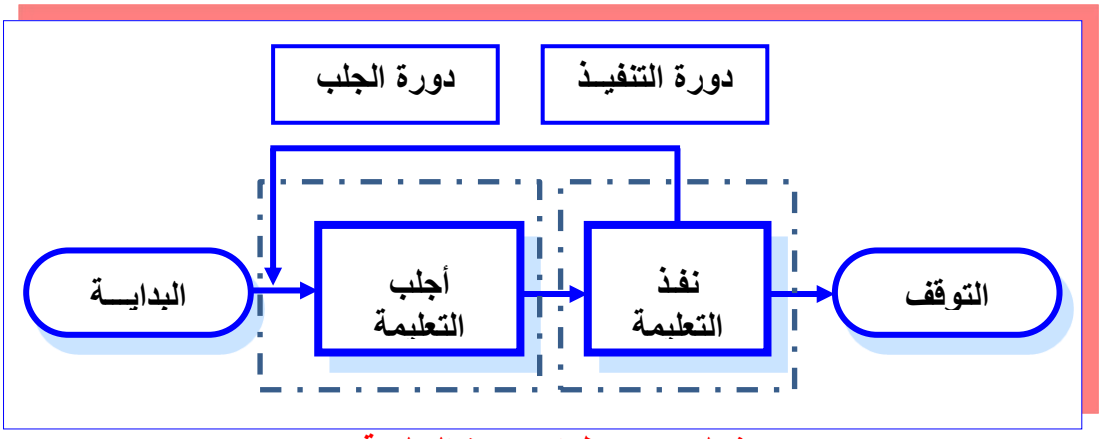
إنَّ نقل البيانات ضمن المعالج وبين المعالج والذاكرة يجب أن يكونا متزامنين لضمان أن تكون البيانات المطلوبة لتنفيذ كل تعليمة (Instruction) متوافرة عندما يصل تدفق التنفيذ لنقطة مناسبة، هذا التزامن يُنجز بواسطة تحريك البيانات في مدد فاصلة بحيث تستجيب لنبضات مؤقت النظام ويتم ذلك عن طريق إرسال إشارات التحكم التي تخبر مكونات المعالج والذاكرة متى ترسل البيانات أو متى تنتظر وهكذا، ولأن معظم الأجهزة لا يمكن أن تسير بسرعة المعالج نفسها، تخلق الدوائر في الأجزاء المختلفة من اللوحة الأم إشارات تحكم ثانوية تقوم بالعمل على نسب مختلفة لسرعة المؤقت الفعلية للنظام.

إنَّ المدة الزمنية المستغرقة لأي عملية قراءة أو كتابة يقوم بها المعالج تسمى بدورة الآلة (Machine Cycle) أمَّا المدة الزمنية التي تستغرقها السلسلة المتكاملة من الإجراءات أو العمليات لتنفيذ تعليمة معينة فتسمى بدورة التعليمة (Instruction Cycle)، وهذه المدة الزمنية لدورة التعليمة ليست ثابتة القيمة، فهي تتغير بحسب نوع التعليمة، وتتراوح ما بين دورة آلة واحدة وخمس دورات في بعض المعالجات، وكل دورة آلة تتراوح ما بين ثلاث في الأقل إلى خمس دورات للساعة، ويرمز لها بـ (T) ودورة الساعة هي نبضة إلكترونية واحدة من المعالج.

5-2 دورة التعليمة Instruction Cycle

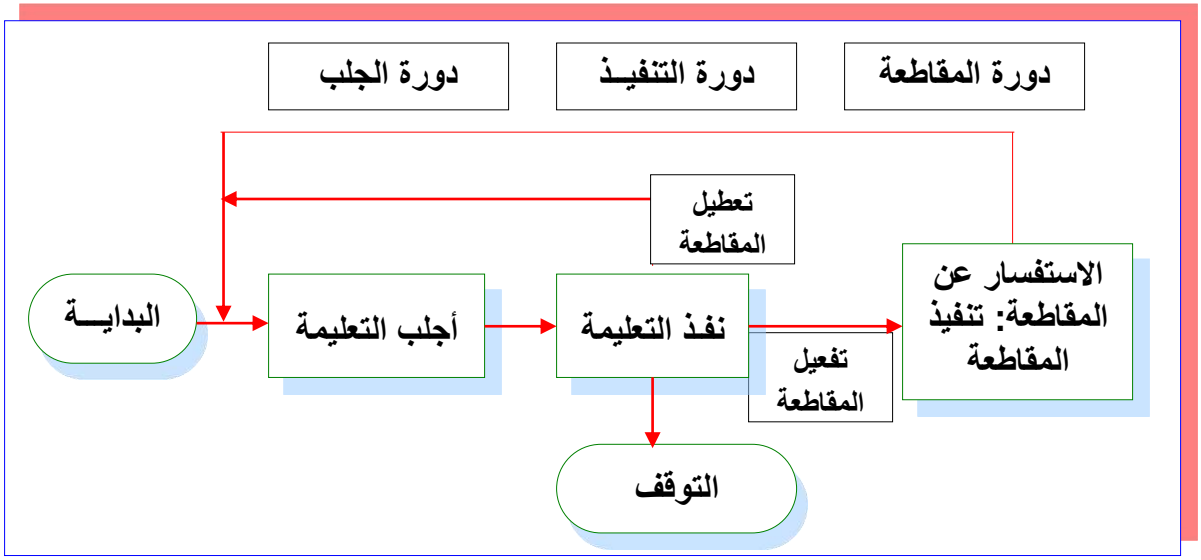
تتضمن دورة التعليمة بنحوٍ أساسي في جميع المعالجات من دورتين رئيسيتين كما في الشكل (2-4)، هما:

1. دورة جلب التعليمة (Fetch Instruction Cycle).
2. دورة تنفيذ التعليمة (Execution Instruction Cycle).



شكل 2-4 خطوات دورة التعليمة

وتتضمن كل دورة عدداً من الخطوات كل خطوة منها تسمى بالطور (Phase)، وعلى الرغم من أنَّ المعالج والذاكرة يكونان على اتصال لتنفيذ التعليمات بنحوٍ مناسب إلا أنَّ هناك أجزاء إضافية في الحاسوب قد يتصل بها المعالج للحصول على البيانات أو إظهار النتائج، ومنها وحدات الإدخال والإخراج (Input /Output Modules) فعند اتصاله بمثل هذه الوحدات ربما ستظهر مرحلة جديدة في دورة التعليمة وهي مرحلة المقاطعة (Interrupt) كما يظهر في الشكل (2-5).



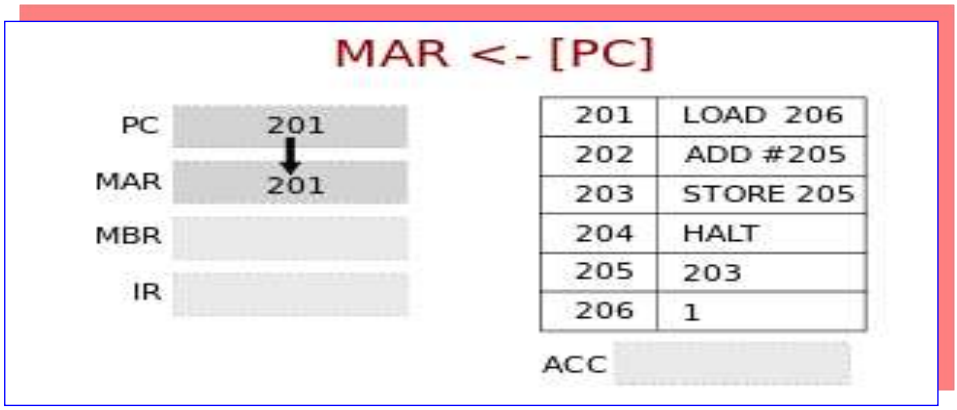
شكل 2-5 دورة التعليمة مع دورة المقاطعة

2-5-1 دورة الجلب Fetch Cycle

تقوم وحدة السيطرة المنطقية بدورة جلب التعليمة و تتكون هذه الدورة من ثلاثة أطوار رئيسة، هي:

1. طور العنوان (Address Phase):

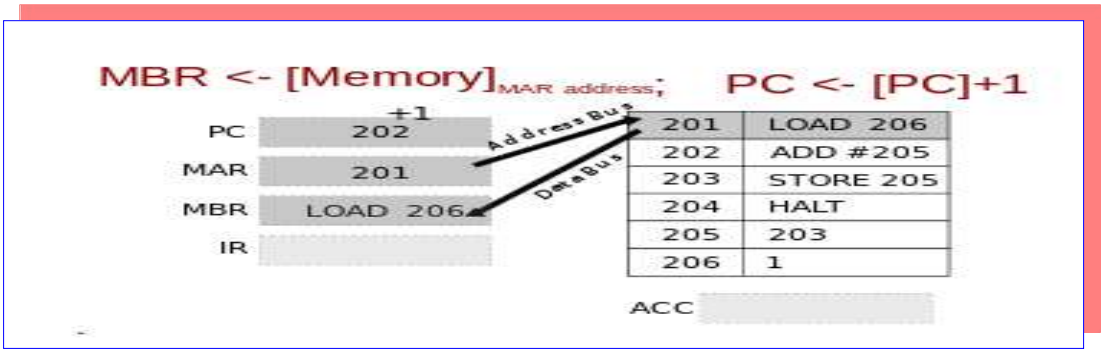
ويُعرف بـ T_0، هذا الطور يمثل بداية دورة جلب التعليمة، إذ يحتوي عداد البرنامج على عنوان التعليمة المراد تنفيذها، فيتم وضع هذا العنوان في مسجل عنوان الذاكرة Memory Address Register - MAR، كما في الشكل (2-6).



شكل 2-6 طور العنوان في دورة جلب التعليمة (للإطلاع)

2. طور الزيادة (Increment Phase):

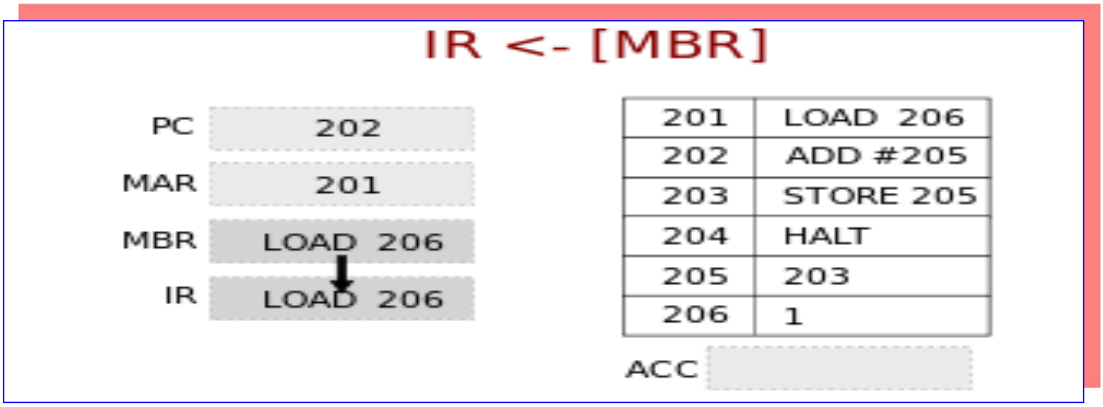
ويعرف بـ T_1، وفي هذا الطور يتم زيادة عداد البرنامج بواحد، ليكون لدينا مؤشر لموقع التعليمة التالية، ويقوم المعالج بوضع العنوان الموجود في مسجل MAR على ناقل العناوين ليتم تحديد موقع الذاكرة المطلوب، ثم بعد ذلك ينقل محتوى ذلك العنوان عبر ناقل البيانات إلى مسجل الذاكرة العازلة (MBR – Memory Buffer Register) الذي يقوم بتخزين شفرة التعليمة التي تم جلبها من الذاكرة لحين الانتهاء من تنفيذها وقبل نقلها إلى مسجل التعليمة (IR) كما في الشكل (2-7).



شكل 2-7 طور الزيادة في دورة جلب التعليمة (للإطلاع)

3. طور الذاكرة (Memory Phase)

ويعرف بـ T_2، وفي هذا الطور يتم نقل محتوى مسجل الذاكرة العازلة (MBR) إلى مسجل التعليمة (IR) كما في الشكل (2-8)، ليتم بعد ذلك فك شفرته وتنفيذه.



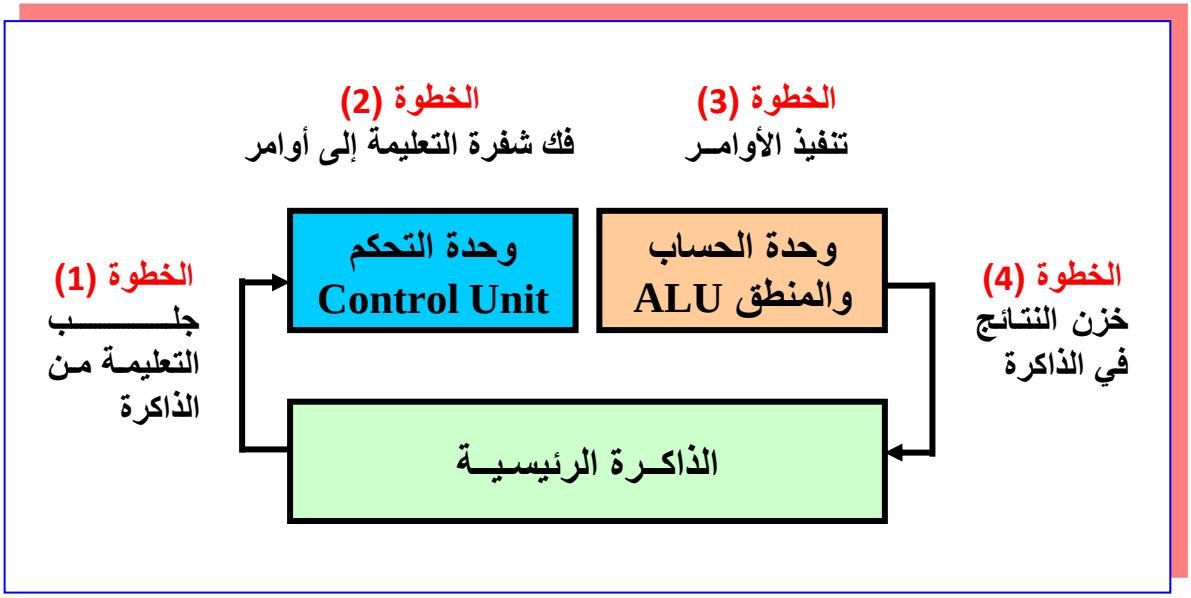
شكل 2-8 طور الذاكرة في دورة جلب التعليمة (للأطلاع)

2-5-2 دورة التنفيذ Execution Cycle

بعد أن تم تحميل التعليمة إلى السجلات المسؤولة تقوم وحدة السيطرة بالتأكد من محتوى هذه التعليمة وفك الشفرة التي بداخلها لتحدد نوع التعليمة التي يجب تنفيذها، والإجراء المناسب لتنفيذ التعليمة يعتمد على كل من الشفرة التي تبين التعليمة المطلوب تنفيذها والعنوان الذي يشير إلى مكان البيانات التي سيتم التعامل معها.

تحتوي دورة التنفيذ على ثلاثة أطوار تُعرف بـ (T_3, T_4, T_5) تعتمد انتقالات المسجل في أثناء دورة التنفيذ على التعليمة المراد تنفيذها، فعلى سبيل المثال LDA 21H هذه التعليمة تتطلب انتقالات مختلفة مثلما تتطلب التعليمة ADD A, B.

لنفرض في نهاية دورة الجلب كان سجل التعليمة (IR) يحتوي على التعليمة الآتية: ADD A,B ففي أثناء الطور الأول من دورة التنفيذ T_3 يذهب محتوى سجل التعليمة إلى محلل شفرة التعليمة لتحليل التعليمة، وفي الطور الثاني T_4 تسمح وحدة التحكم بجلب محتوى السجل B وعادة يتم ذلك أثناء منتصف هذا الطور، وفي أثناء الطور الثالث T_5 تقوم وحدة الحساب والمنطق بجمع محتوى السجل B مع المركم A وفي منتصف الطور يتم تحميل (خزن) النتيجة في المركم A، ويوضح شكل (2-9) العلاقة ما بين المعالج والذاكرة الرئيسة في أثناء أداء دورة التعليمة وما تمر بها من خطوات رئيسة في أثناء هذه الدورة.



شكل 2-9 العلاقة ما بين المعالج والذاكرة الرئيسة في أثناء دورة التعليمة

ومن أجل ضمان أن جميع التعليمات تم انجازها في الوقت الصحيح، يجب أن تكون الإشارات الخارجة من وحدة السيطرة تحكم إلى جميع أجزاء الحاسوب في أثناء مدد زمنية منتظمة، والإجراءات على هذه الإشارات يجب أن تتم في الوقت نفسه وضمن هذه المدد مما يعني أن عمل المعالج هو عمل متزامن، لذا ينفذ المعالج خطوات التعليمة على إيقاع نبضات مؤقت النظام فيتم تنفيذ كل خطوة في أثناء ذبذبة أو أكثر من ذبذبة للمؤقت.

6-2 العوامل المؤثرة في سرعة المعالج

إنَّ السرعة في تنفيذ المهام هي من الأمور المهمة التي تؤخذ بالحسبان عند التعامل مع أي معالج أو حتى عند شراء المستهلك أي حاسوب شخصي، فأول الاستفسارات تكون غالبًا متعلقة بسرعة المعالج، وهذه السرعة تُقاس بالزمن الذي تستغرقه النبضة الكهربائية لتقوم بالمرور دورة كاملة داخل الترانزستورات الموجودة في المعالج بحيث يتم في هذا الزمن تنفيذ الأمر الذي تلقاه المعالج، وتقاس السرعة بالميجاهرتز، فعندما تقول إنَّ المعالج 400 MHz، هذا يعني أنَّ عدد النبضات التي تمر في الترانزستورات تصل إلى 400 مليون نبضة في الثانية، أي إنَّ المعالج قادر على تنفيذ 400 مليون تعليمة (إيعاز) في الثانية، وبالتالي كلما زادت سرعة المعالج زادت سرعة الحاسب وكفاءته.

وفي الوقت نفسه لا بُدَّ من التنبيه إلى إنَّ سرعة المعالج ليست هي العامل الوحيد الذي يقرر سرعة الحاسوب، بل المهم أيضًا سرعة حركة البيانات بين الأجزاء المختلفة في الحاسوب وبخاصة من وإلى المعالج، وفي ما يلي أهم العوامل التي تجعل معالجاً ما يكون أسرع من معالج آخر:

1- **مجموعة التعليمات (Instruction Set):** وهي شفرة مبنية داخل المعالج تخبره كيف ينفذ واجباته، لا يملك مستعمل المعالج القدرة على التحكم بمجموعة التعليمات فهي مبنية داخل المعالج ولا يمكنه تحديثها (Update)، ومجموعة التعليمات مع البنية الداخلية للمعالج تؤثر بنحوٍ واضح في أداء المعالج، فالبنية الداخلية تحدد كم عدد الدورات والنبضات التي يحتاج إليها المعالج لتنفيذ تعليمة معينة. وبمعنى آخر بعض مجموعات التعليمات هي أكثر كفاءة من غيرها, فمثلا التقنيات المعتمدة في تصميم المعالج حتى أواخر العقد 1980 هي تقنية مجموعة التعليمات المعقدة (Complex Instruction Set Computer) المعروفة اختصارًا بـ(CISC)، وتتضمن هذه التقنية مئات التعليمات لتغطية كل الحالات الممكنة، ولذلك يحتاج المعالج إلى مئات الآلاف من الترانزستورات مما يجعل تصميمها معقدًا جدًا ويحتاج إلى وقت طويل، وكلفة عالية، ولكن بعد

أواخر العقد 1980 تم تطوير تقانة جديدة سميت بمجموعة التعليمات المختصرة (Reduced Instruction Set Computer) المعروفة اختصارًا بـ (RISC)، إقترحت هذه التقانة عدم إعتماد جميع تعليمات المعالجات (CISC) بل تقليل عدد التعليمات من المئات إلى نحو 40 تعليمة أو ما يقارب ذلك، ويتم استغلال جميع الترانزستورات البقية لتحسين قدرة المعالج وكفاءته.

2- **تردد المعالج أو تردد مولد النبضات:** يقصد بتردد المعالج تردد الساعة التي يعمل عليها المعالج، فكلما كان تردد الساعة أعلى كلما أصبح بإمكان المعالج عمل أشياء أكبر في وقت أقل، وتقاس سرعة المعالج بالهرتز، فمعالج سرعة تردده 200 MHz قادر على عمل 200 مليون دورة في الثانية، أما كم من العمليات الحسابية يتم في هذه الدورة فهذا راجع لبنية المعالج والجيل الذي ينتمي إليه.

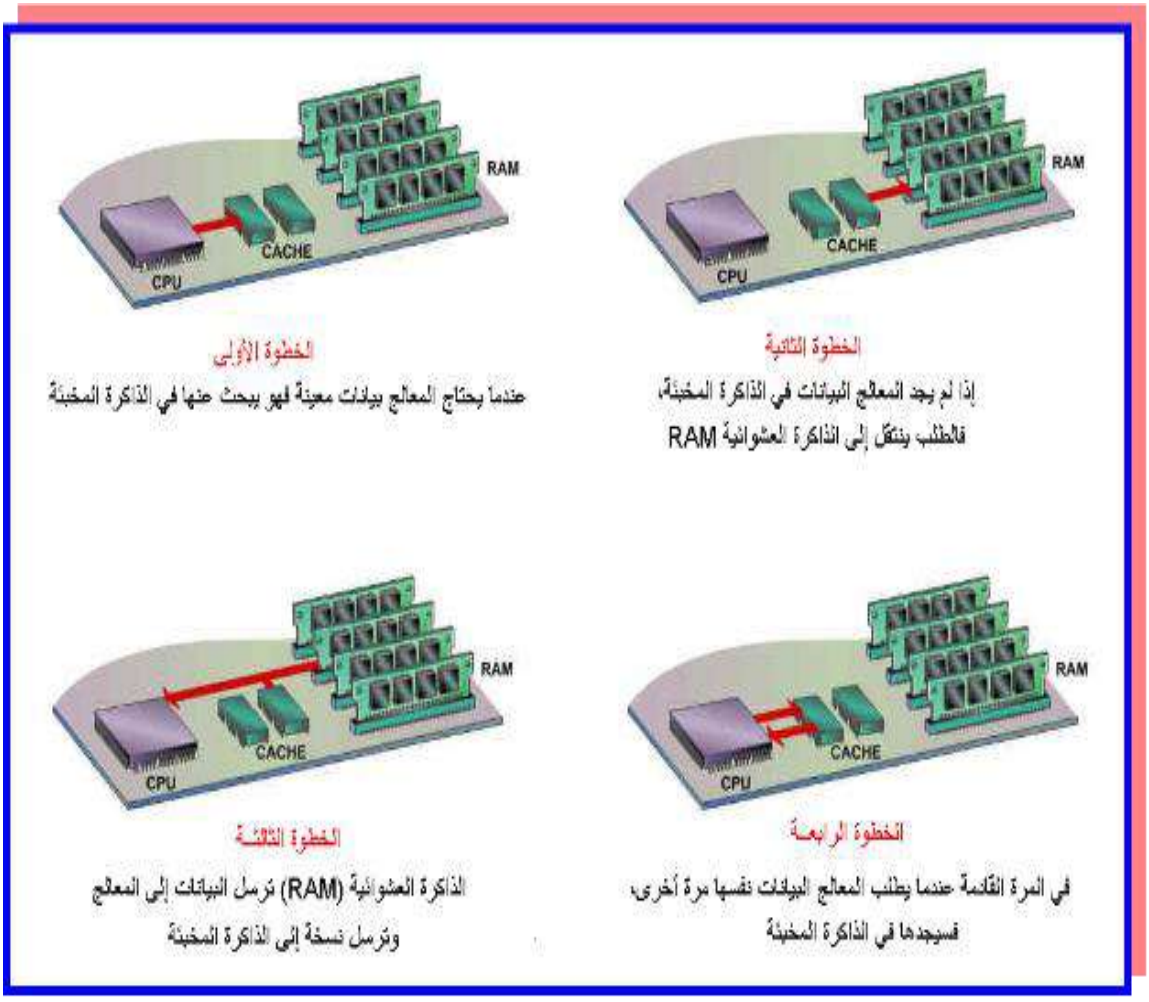
3- **سرعة النواقل Bus Speed:** إن الناقل السريع يضمن توصيل البيانات بالسرعة التي تجعل المعالج لا يكون في حالة انتظار، ويعد كل من تردد الناقل وعرضه مهمًا، وفيما يكون عرض ناقل النظام 64 بتاً في المعالجات الحديثة أجمعها.

يحكم تردد الناقل على سرعة الناقل بحسب المعادلة الآتية:

سرعة الناقل (بت/ ثانية) = عرض الناقل (بت) × تردده (هيرتز)
التردد = 1\ الزمن (ثانية)

4- **الذاكرة المخبئة (Cache Memory):** إنَّ حركة البيانات المتكررة ما بين ذاكرة الوصول العشوائي (RAM) وسجلات المعالج تعد واحدة من أكثر العمليات استهلاكًا للوقت التي يقوم المعالج بها وهذا بسبب أن سرعة هذه الذاكرة (RAM) أبطأ بكثير من سرعة المعالج، ولتحسين الأداء لجأ مصممو الحاسوب إلى استعمال الذاكرة المخبئة وهي ذاكرة صغيرة تشبه ذاكرة الوصول العشوائي إلا أنها أسرع منها وأصغر، وتوضع على ناقل النظام بين المعالج وذاكرة الوصول العشوائي مستغلين حقيقة أن المعالج يطلب البيانات نفسها أكثر من مرة في أوقات متقاربة فتقوم الذاكرة المخبئة بتخزين البيانات الأكثر طلبًا من المعالج مما يجعلها في متناول المعالج بسرعة حين يطلبها، شكل (2-10) يوضح عمل المعالج باستعمال الذاكرة المخبئة، فعندما يريد المعالج جلب بيانات أو تعليمات فإنه يبحث عنها أولاً في الذاكرة المخبئة، فإن لم يجدها جلبها

من الذاكرة العشوائية، إن حجم هذه الذاكرة وسرعتها شيء مهم جدًا ولها تأثير كبير في أداء المعالج، توضع الذاكرة المخبئة داخل المعالج وفي ذلك فائدتان: إحداهما هي السرعة، أما الأخرى فتبرز في حالة تركيب أكثر من معالج واحد على اللوحة الأم، لأن كل معالج له الذاكرة العشوائية الخاصة به فلا تتزاحم المعالجات على الذاكرة المخبئة.



شكل 2-10 ارسال البيانات ما بين المعالج وذاكرة الوصول العشوائي والذاكرة المخبئة

5-الحرارة وتبديد الحرارة (Heat and Heat Dissipation): أي قطعة إلكترونية في أي جهاز ومنها المعالج تحتاج إلى أن تكون ضمن مدى معين من درجات الحرارة إذا زادت درجة الحرارة عن هذا الحد فإنها تقصر من عمر المعالج وتبطئ أدائه وتتسبب بأخطاء في الحسابات، وقد يعيد الحاسب تشغيل نفسه من دون سبب، إن الحرارة الناتجة من مرور التيار الكهربائي في الترانزستورات تولد حرارة، وكلما كانت فولتية المعالج ومعماريته أقل كلما كانت الحرارة الناتجة أقل، لذا تنتج المعالجات المختلفة كميات مختلفة من الحرارة، فالمعالج بنتيوم 3 مثلاً ينتج

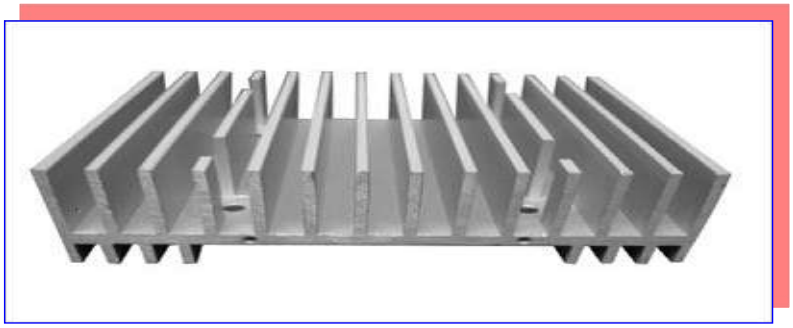
كمية من الحرارة أكبر من بنتيوم1، وتقاس كمية الحرارة الناتجة من المعالج بالسيليزية °C.

وهناك عدة وسائل لتبريد المعالجات الحديثة، منها:

(1) المبدد الحراري: هو عبارة عن شريحة من المعدن تلتصق بسطح المعالج مربعة الشكل أو مستطيلة عادة، إلا أن بعضها شبه دائري يخرج منها بنحوٍ عمودي عدد كبير من الأعمدة المعدنية، وفائدة هذا المبدد الحراري هو إنتشار الحرارة الناتجة من المعالج بين القضبان العمودية ذات المساحة السطحية الكبيرة فتقوم بتبديد الحرارة.

يصنع المبدد الحراري من الألمنيوم، لأنه موصل جيد للحرارة، ويجب أن يكون مدخل الهواء أبعد ما يمكن عن المخرج حتى لا يعود الهواء الساخن الخارج من المبدد للدخول مرة ثانية، وكذلك يجب أن يكون المبدد الحراري ملتصقاً بسطح المعالج تمامًا، إن حرارة المعالج في أثناء العمل تعتمد على كفاءة المبدد الحراري، وعلى كمية الحرارة التي ينتجها المعالج، وأيضاً على درجة حرارة علبة النظام، ولا يمكن لأي مبدد حراري أن يحفظ درجة حرارة المعالج إلى أقل من درجة حرارة علبة النظام، هذا لأن الهواء الذي يدفع بين أعمدة المبدد الحراري مأخوذ من علبة النظام نفسها.

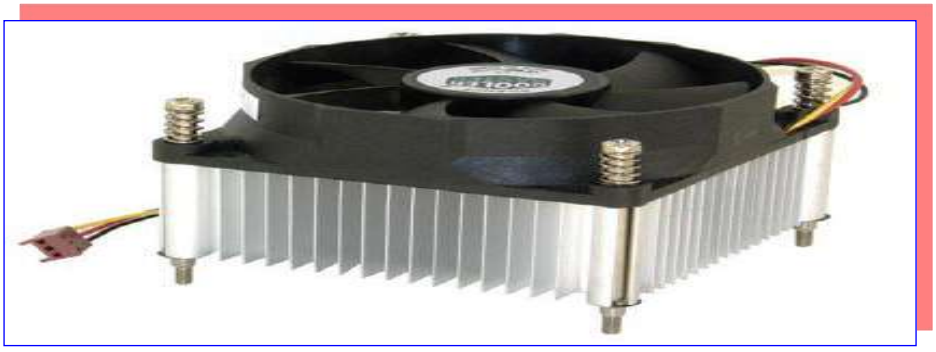
في علب النظام من نوع ATX تساعد العلبة نفسها في تبريد المعالج، إذ يقع المعالج تحت مزود الطاقة ليكون في مجرى الهواء، وهذا يساعد كثيرًا في تفادي مشكلة الحرارة. إن أحد أسباب ارتفاع درجة حرارة المعالج هو وجود الأوساخ داخل المبدد الحراري مما يمنع الهواء من المرور فيه، ويسمح بارتفاع درجة الحرارة ومن المفيد تنظيف الحاسب من الداخل، يوضح الشكل (2-11) المبدد الحراري.



شكل 2-11 المبدد الحراري

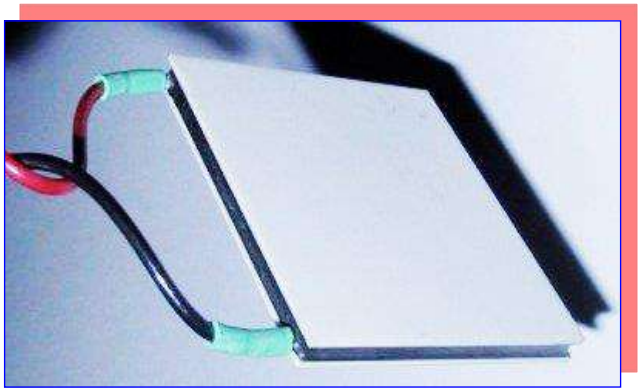
(2) مروحة التبريد: عملها هو دفع الهواء بين الأعمدة المعدنية للمبدد الحراري بحيث يمكن تبديد قدر أكبر من الحرارة، يمثل شكل (2-12) مبدداً حرارياً ملحقة به مروحة تبريد، وقد يستعمل

المبدد الحراري من دون مروحة تبريد، وهذا يقلل التكلفة ويجعل المعالج غير معرض للتلف بسبب توقف المروحة عن العمل، ولكن استعمال المروحة يجعل التبريد أفضل، بعض المعالجات مثل (Pentium Overdrive) لديها مروحة داخلية في الرقاقة، إذا حصل وعطبت هذه المروحة فإن المعالج يحمي نفسه بإنقاص تردده إلى (25 MHz) إلى أن يتم استبدال المروحة، بعض اللوحات الأم تزود بمقاييس للحرارة (Thermometers) لقياس درجة حرارة المعالج أو بأجهزة لمراقبة التيار الكهربائي الذاهب لمروحة تبريد المعالج وبذلك تتمكن من اكتشاف أي غلط أو مشكلة قد تؤدي لزيادة درجة حرارة المعالج.



شكل 2-12 المبدد الحراري ملحق بمروحة تبريد

(3) مبرد بالتير (Peltier Cooler): جهاز على شكل شريحة مربعة الشكل يوضع على سطح المعالج ويعمل بالكهرباء ويقوم بسحب الحرارة من سطح المعالج إلى السطح الآخر، ويثبت المبدد الحراري من أعلى، تقوم هذه الأجهزة بالتبريد بكفاءة تامة، ولكنها غالية الثمن، وتعد كلفتها واحدة من الأسباب التي قللت استعمالها في الحواسيب الحديثة، الشكل (2-13) يظهر مبرد بالتير.



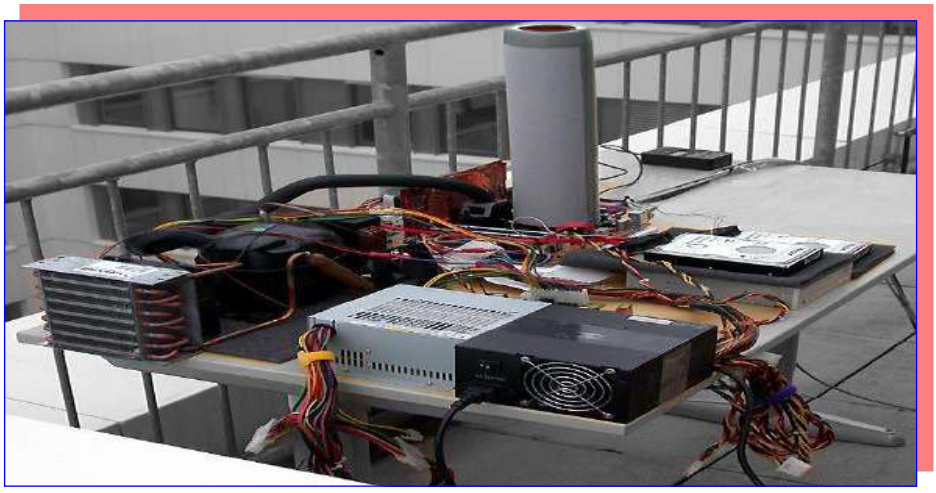
شكل 2-13 مبرد بالتير

وهناك أنواع أخرى من طرائق تبريد المعالج، وبعض هذه الطرائق ما زالت تثير الاستفسارات ومنها طريقة التبريد بالماء (Water Cooling)، إذ يستعمل الماء بطريقة مشابهة لتلك المعتمدة لتبريد السيارات، كما مبين في الشكل (2-14)، إذ تعتمد الطريقة على تمرير الماء داخل المبدد الحراري، أو يمكن التخلص من المبدد الحراري واستعمال الماء فقط عن طريق علبة صغيرة يمر فيها الماء.



شكل 2-14 تبريد المعالج بالماء

وبعض طرائق التبريد لا يمكن اعتمادها في كل مكان، بل لا تعتمد إلا في مختبرات الشركات المصنعة للمعالجات وبظروف خاصة لإختبار قدرة المعالج مثل طريقة التبريد بالنيتروجين السائل (Liquid Nitrogen Cooler) كما في الشكل (2-15).



شكل 2-15 تبريد المعالج بالنيتروجين السائل

وهناك طريقة استعمال الضاغطات (Compressor Cooling) وهي مشابهة لفكرة استعمال الضاغطات في أجهزة التبريد العامة المستعملة في المنازل والمؤسسات, ولكنها بالتأكيد

أصغر حجماً، ونظراً لإستهلاكها للكهرباء وكلفتها العالية فهي غير محبذة، يظهر الشكل (2-16) استعمال هذا النوع من التبريد للمعالجات.

ويمكن أن تدمج الأنواع السابقة للحصول على نتائج أفضل، وقد يكون هذا الدمج لأغراض الاختبارات والبحوث العلمية في الشركات المصنعة والمؤسسات العلمية، أو قد يكون لأغراض الاستعمال الشخصي، وهذا يعتمد على قدرة المستعمل في تحمل تكلفة بعض هذه الأنواع ومناسبة بيئة العمل لها.



شكل 2-16 تبريد المعالج باستعمال الضاغطات Compressors

وبعد أن درسنا أهم العوامل التي تؤثر في سرعة المعالج لا بد من التنبيه إلى أنَّ كل واحد من هذه العوامل لا يؤثر في السرعة ومستوى أداء المعالج بصورة منفردة، بل إن بعضها متعلق ببعض، فعلى سبيل المثال لو تم مضاعفة سرعة التوقيت من دون تغيير العوامل الأخرى، فإن ذلك لن يضاعف سرعة المعالج بالضرورة، فنوع المعالج وبنيته الداخلية المتضمنة لسعة السجلات وغيرها، وهندسة بناء الناقل، وطبيعة الأوامر الموضوعة، هذا كله يصنع الاختلاف، فضلاً عن طبيعة بعض البرامج التطبيقية التي يقوم بتنفيذها المعالج، فإن كمية الذاكرة العشوائية (RAM) مهمة كما لاحظنا ذلك، وكذلك عدد البرامج المنفذة في الوقت نفسه.

فللحصول على سرعة وكفاءة في عمل المعالج لا بد من أخذ جميع هذه العوامل بالحسبان وجعلها تعمل بنحوٍ متوافق للحصول على المطلوب.

أسئلة الفصل الثاني

س1/ عرف الآتي:

1) لغة الآلــة. 2) لغة التجميع. 3) برنامج المجمع.

4) البرنامج. 5) دورة الآلــة. 6) الذاكرة المخبأة.

7) دورة الإيعاز. 8) المبدد الحراري.

س2/ ما عيوب لغة الآلــة؟

س3/ ما الاختلاف الأساسي بين حواسيب (CISC) وحواسيب (RISC)؟

س4/ ما الغرض من استعمال الآتي:

1) برنامج المجمع Assembler.

2) مؤقت النظام System Clock.

س5/ املأ الفراغات التالية بما يناسبها ؟

1. التعليمة HALT تخبر المعالج بـ ..... و .....
2. تقسم دورة التعليمة على أربع مراحل هي: ..... و..... و.....
3. في دورة جلب التعليمة وفي أحد أطوارها يتم نقل محتوى المسجل الذاكرة عبر ..... إلى المسجل MBR.
4. من العوامل المؤثرة في سرعة المعالج .....و.....و .....

س6/ ما الفرق بين دورة الآلــة Machine Cycle، ودورة التعليمة Cycle Instruction؟

س7/ وضح بالشرح كيف تؤثر الحرارة في سرعة المعالج؟ وإذكر مع الشرح وسيلتين من الوسائل المعتمدة لتبديد حرارة المعالج ؟

س8/ ضع علامة (✓) أمام العبارة الصحيحة وعلامة (✗) أمام العبارة الخاطئة مع تصحيح الخطأ أينما وجد:

1. تعد لغة الترميز من اللغات منخفضة المستوى، لأنها لغة ضعيفة لا تدعم الكثير من التعليمات الضرورية.
2. تقوم التعليمة MOV بنقل محتويات معامل المصدر إلى معامل الهدف، بحيث يصبح محتوى معامل المصدر خاليًا من المحتويات (صفرًا).
3. إنَّ كفاءة أداء الحاسوب تعتمد بنحوٍ رئيس على سرعة المعالج الذي بدوره يتأثر بعوامل متفرقة تتعلق ما بين المكونات المادية والبرمجيات التي يستعملها الحاسوب.
4. دورة التعليمة ثابتة القيمة لا تتغير بتغير المعالج.
5. مبرد بالتير من الوسائل الناجحة لتبريد المعالج والمستعملة بكثرة لكفاءتها.
6. وحدة قياس الحرارة المتولدة من المعالج هي الكالفن.
7. تتميز لغة الآلة ولغة التجميع بتعاملها المباشر مع المكونات المادية (Hardware) للمعالج ؟

س9/ وضح بالرسم دورة التعليمة مع وجود دورة المقاطعة ؟

الفصل الثالث

طرائق انتقـال المعلومـــات

1-3 المقدمة

إن من أهم النشاطات الأساسية في نظام أي معالج دقيق هو حركة انتقال المعلومات بين المعالج والأجزاء الداخلية في الحاسوب من جهة مثل: الذاكرة (Memory)، والسجلات (Registers)، ووحدة الحساب والمنطق (Arithmetic and Logic Unit – ALU) وكذلك بين المعالج والأجزاء الخارجية من جهة أخرى مثل: لوحة المفاتيح (Keyboard)، أو الشاشة (Monitor)، أو الطابعة (Printer)، إلى آخره من أجزاء، ومبدأ عمل المعالج يقوم على التعامل مع البيانات على شكل كلمة (Word) التي تتكون من (8, 16, 32, 64, … Bits) بحسب نوعية المعالج المستعمل، فبالنسبة إلى معالج ذات (8 Bits) تسمى هذه الكلمة بالـ Byte، وهذا الـ bits كما درسنا سابقًا عبارة عن (واحدات وأصفار)، وكل بت يعد شحنة كهربائية يقوم المعالج بنقلها وخزنها والتعامل معها بحسب المطلوب، هذه الشحنات أصلها ربما صورة أو رسالة تم تحويلها إلى صيغة رقمية مناسبة للتعامل مع الحاسوب وأجزائه.

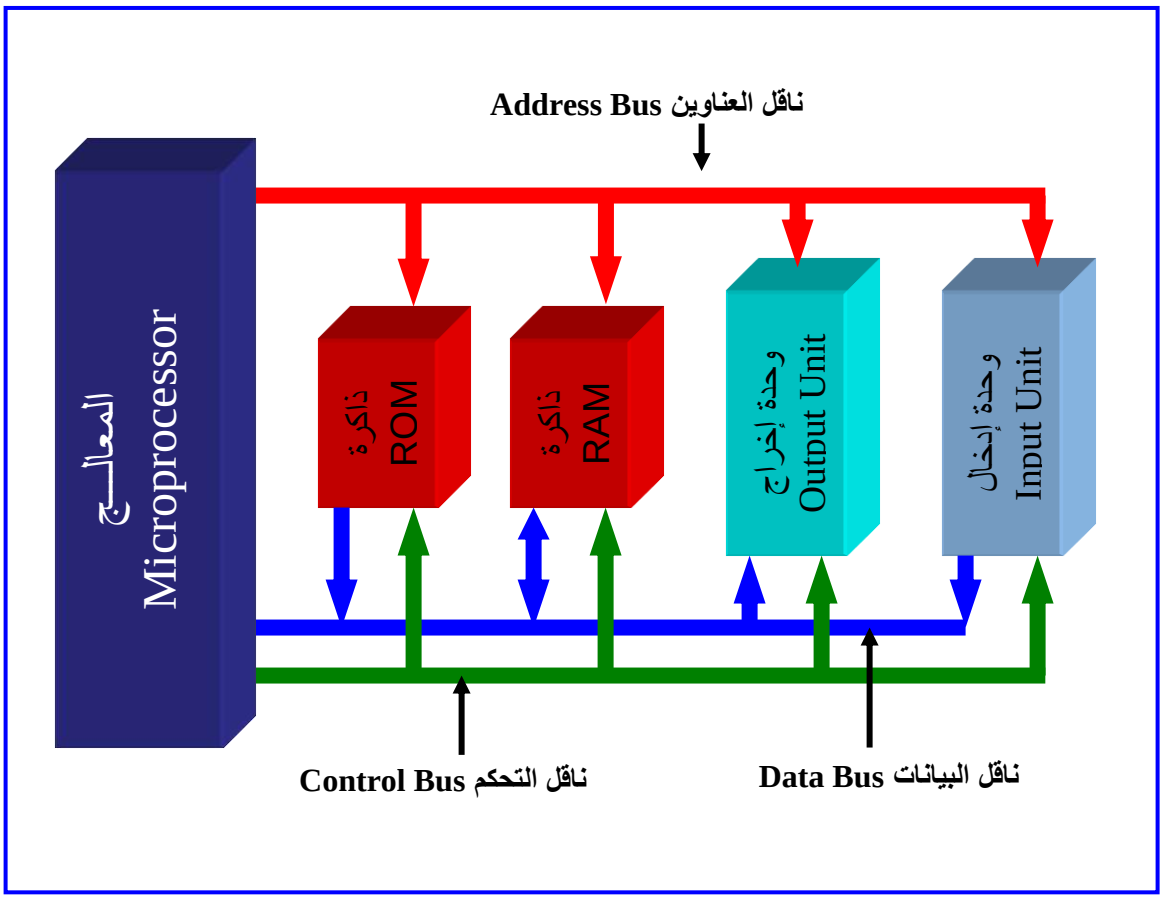
لكن هذه المعلومات كيف يتم نقلها ما بين المعالج والأجزاء المختلفة للحاسوب؟ وما الطرائق والتقنيات المستعملة في انتقال البيانات؟ وهل يعتمد اختلاف الطرائق على المكونات المادية Hardware أم على البرامجيات Software أم على كليهما معًا؟ وهل ظهور أنواع جديدة من المعالجات وتطورها أسهم في تطور طرائق انتقال المعلومات بين المعالج والأجزاء الأخرى للحاسوب؟

ولكن قبل الدخول بالتفصيل لا بد من التذكير بأنَّ المعالج الدقيق كقطعة إلكترونية هو غير قادر على القيام بأي وظيفة منفردًا، وذلك لأن عمله مرتبط باستعمال:

1. **ذاكرة القراءة فقط (ROM: Read Only Memory):** وهي ذاكرة قابلة للقراءة فقط، تستعمل لتخزين البرنامج الدائم الذي سيتبعه المعالج، والبرمجة تكون خارجية من طرف المستعمل.
2. **ذاكرة الوصول العشوائية (RAM: Random Access Memory):** وهي ذاكرة قابلة للقراءة والكتابة، وتستعمل من طرف المعالج لتخزين بيانات مؤقتة غير دائمة.

3. **وحدة إدخال (Input Unit):** تستعمل من طرف المعالج الدقيق للحصول على بيانات من الأجهزة الخارجية الموصلة مع هذه الوحدة.
4. **وحدة إخراج (Output Unit):** تستعمل من طرف المعالج الدقيق لإرسال بيانات إلى الأجهزة الخارجية الموصلة مع هذه الوحدة.
5. **وحدة الحساب والمنطق (ALU) ووحدة السيطرة (Control Unit)** (CU) : وهما وحدتان داخليتان يمثلان معًا المعالج الدقيق وفيها يتم تطبيق كل تعليمات البرنامج.

هذه الأجزاء تربط بعضها مع بعض والمعالج عن طريق ما يُعرف بالنواقل (Buses) كما موضح في الشكل (3-1).



شكل 3-1 إتصال المعالج الدقيق بالوحدات المختلفة

2-3 النواقل في الحاسوب

الناقل (Bus) هو عبارة عن مسار اتصال يربط جهازين أو أكثر، السمة الرئيسة للناقل هو أنه وسيلة نقل مشتركة، إذ إن هناك أجهزة متعددة تتصل بالناقل الواحد، والإشارة أو البيانات المرسلة من أي جهاز تكون متوافرة للتسلم من جميع الأجهزة المتصلة بالناقل.

يتكون عادة الناقل الواحد من مجموعة من الخطوط المتوازية التي تقوم بنقل قيمة الجهد على كل سلك (0 V) أو (5 V) أو حالة القيم المنطقية 0 أو 1 من مكان إلى مكان آخر، وعادة تكون هذه المعلومات خارجة من مصدر معين ومتجه إلى هدف آخر، تحتوي نظم الحاسوب على عدد مختلف من النواقل بعضها يكون أحادي الاتجاه، مثلًا أن يقوم بحمل الإشارات من المعالج إلى الأجهزة الطرفية، وبعضها الآخر يكون ثنائي الاتجاه، إذ يقوم بنقل الإشارة الخارجة من المعالج إلى الأجهزة الطرفية في أزمنة معينة أو العكس من الأجهزة الطرفية إلى المعالج في أزمنة أخرى.

كما ذكرنا آنفًا، إن النواقل هي وسيلة نقل مشتركة، فعند ربط المعالج مع أي جهاز من الأجهزة الطرفية تنشأ مشكلتان من هذا الربط يجب التغلب عليهما:

إحداهما: يجب التأكد من أنه في أي لحظة لا يتم نقل أي معلومة إلا لجهاز واحد فحسب، أي إنه عندما يكون المعالج في حالة اتصال (مخاطبة) مع أي جهاز من الأجهزة الطرفية، فإنَّه يكون على اتصال بهذا الجهاز فحسب من دون الأجهزة الأخرى، فمثلًا عندما يتم إرسال معلومة من المعالج إلى وحدة الإخراج مثلاً، فمن البديهي بأنَّها لن تذهب أيضًا إلى أي bits من بتات ذاكرة الوصول المباشر RAM.

والأخرى: يجب التأكد من أنه عند اتصال المعالج بأي واحد من الأجهزة الطرفية لن تشوش الأجهزة الأخرى أو تتداخل في عملية الاتصال، فمثلًا عندما يقوم المعالج بقراءة معلومة من ذاكرة الوصول العشوائي RAM، لا بد من التأكد بأن ذاكرة القراءة فقط ROM أو أي بوابة إدخال لن تتدخل وترسل هي الأخرى معلومات إلى المعالج في الوقت نفسه بحيث يحدث ما يسمى بتصادم المعلومات على الناقل.

ولحل هاتين المشكلتين وغيرهما تقوم نظم الحاسوب بتوفير أنواع مختلفة من النواقل لتجنب المشكلات المذكورة سابقًا، إذ توفر مسارات بين المكونات على شتى مستويات التسلسل الهرمي لنظام الحاسوب، وكما درسنا في الفصل الأول يسمى الناقل الذي يربط الأجزاء الرئيسة للحاسوب (المعالج، والذاكرة، ووحدات الإدخال والإخراج) بناقل النظام (System Bus)، كما موضح في

الشكل (3-1) الذي يتكون عادة من خمسين إلى مئات من الخطوط المنفصلة، وكل خط يخصص لوظيفة معينة، وبالرغم من وجود الكثير من تصاميم النواقل المختلفة يمكن تصنيف الخطوط على أي ناقل إلى ثلاث مجموعات وفقًا لنوع وظيفتها، وهي:

1. ناقل البيانات Data Bus.
2. ناقل العناوين Address Bus.
3. ناقل التحكم Control Bus.

3-2-1 ناقل البيانات Data Bus

يقوم ناقل البيانات بحمل البيانات من المعالج وإليه، فكلما كان عدد خطوط ناقل البيانات أكثر كلما كان ذلك أفضل, فلو تصورنا أن ناقل البيانات عبارة عن الطريق العام السريع فمن الواضح أنَّ الطريقة الفضلى هي التي يكون فيها عدد المسارات أكثر، وهذا ما يحصل بين المعالج والأجهزة الداخلية والخارجية الأخرى، إلا أنَّ زيادة المسارات تؤدي إلى زيادة كلفة المعالج وبالتالي الحاسوب.

إنَّ ناقل البيانات عبارة عن ممر باتجاهين, لأنه يرسل المعلومات ويستقبلها، ويتألف ناقل البيانات من (8,16, 32, 64, 128) أو حتى أكثر من الخطوط المنفصلة، عدد الخطوط يسمى بعرض ناقل البيانات (Width of Data Bus)، إن عرض ناقل البيانات هو عامل رئيس في تحديد الأداء العام للنظام فتقاس سرعة المعالج بعرض ناقل البيانات، إذ يمكن للناقل الذي عرضه (8 bits) أن يرسل بايت واحدًا فحسب، في حين يرسل الناقل ذو العرض (16 bits) يرسل بايتين إثنين في آن واحد، وبالتالي تكون سرعة المعالجة أعلى مرتين.

ويرمز لخطوط أو أسلاك ناقل البيانات بـ ($D_0, D_1, D_2, \ldots, D_7$) إذا كان المعالج يستعمل ثمانية خطوط.

3-2-2 ناقل العناوين Address Bus

تستخدم خطوط ناقل العناوين للدلالة على مصدر البيانات أو جهتها التي على ناقل البيانات، فعلى سبيل المثال إذا كان المعالج يرغب في قراءة بيانات عبارة عن كلمة (8, 16 or 32 bits) من الذاكرة، فإنه يضع عنوان الكلمة المطلوب نقلها على خطوط ناقل العناوين.

إنَّ أي مكان يريد المعالج أن يتعامل معه سواء كان ذاكرة أو وحدات إدخال وإخراج لا بد أن يقوم المعالج بتحديد عنوان لهذا المكان، وهذا العنوان يتم وضعه في صورة شفرة مكونة من

الواحدات والأصفار بواسطة المعالج على عدد من هذه الأطراف الخارجة من المعالج تسمى بناقل العناوين.

يحدد ممر العناوين الجهاز الذي يتعامل مع المعالج وبالتالي تسمح زيادة عدد خطوط العناوين بزيادة عدد الأجهزة الخارجية، وبمعنى آخر يحدد عدد خطوط العناوين للمعالج عدد المواقع التي يمكنها الاتصال به.

عدد المواقع = 2^x، إذ إنَّ x تمثل عدد خطوط العناوين، وليس لناقل البيانات علاقة بهذا، فعلى سبيل المثال المعالج الذي له 16 خط عنونة ($2^{16} = 65.536 \cong 64\,KB$) ويحتوي كل موقع على بيانات بطول 1 Byte.

ومثال آخر الحاسوب IBM AT المعالج فيه يمتلك 24 خط عنونة و16 خطاً للبيانات، وبالتالي فإنَّ سعة الذاكرة المتاحة في هذه الحالة هي 16 MB (2^{24} = 16 MB) وفي هذه الحالة هناك 2^{24} موقع عنونة، وكل موقع يحتوي بايتين 2Byte، ينظر إلى الجدول (3-1) الذي يبين العلاقة بين عدد خطوط ناقل العناوين وكمية الذاكرة التي يمكن التعامل معها في كل حالة.

ويرمز لخطوط ناقل العناوين أو أسلاكه بـ (A_0, A_1, A_2, …, A_{15}) إذا كان المعالج يستعمل 16 خطًا، علمًا أنَّ ناقل العناوين هو ممر ذو اتجاه واحد من المعالج إلى الوحدات الأخرى.

جدول 3-1 عدد خطوط ناقل العناوين وكمية الذاكرة

عدد خطوط ناقل العناوين	كمية الذاكرة التي يمكن التعامل معها
1	2 Byte
2	4 Byte
3	8 Byte
4	16 Byte
5	32 Byte
6	64 Byte
7	128 Byte
8	256 Byte

512 Byte	9
1024 Byte = 1 k Byte	10
2 k Byte	11
4 k Byte	12
8 k Byte	13
16 k Byte	14
32 k Byte	15
64 k Byte	16
128 k Byte	17
256 k Byte	18
512 k Byte	19
1024 k Byte = 1 Mega Byte	20
2 M Mega Byte	21
4 Mega Byte	22
8 Mega Byte	23
16 Mega Byte	24
32 Mega Byte	25
64 Mega Byte	26
128 Mega Byte	27
256 Mega Byte	28
512 Mega Byte	29
1 Gega Byte	30
2 Gega Byte	31
4 Gega Byte	32

8 Gege Byte	33
16 Gega Byte	34
32 Gega Byte	35
64 Gega Byte	36

3-2-3 ناقل التحكم Control Bus

نظرًا لأن خطوط نواقل البيانات والعنونة هي خطوط مشتركة من جميع مكونات الحاسوب، كان لا بد من وجود وسيلة للسيطرة على هذه النواقل، وهنا يأتي دور ناقل التحكم وهو عبارة عن مجموعة من الخطوط دورها القيام بضبط الأحداث بطريقة تزامنية والتحكم كذلك في وحدات الذاكرة ووحدات الإدخال والإخراج من طرف المعالج.

مثل هذا الخط أو الخطوط تجمع وتعرف بناقل التحكم (Control Bus) وعدد الخطوط في هذا الناقل يختلف من معالج إلى معالج، وسنذكر أربعة خطوط تحكم رئيسة، وهذه الخطوط هي:

1. **خط قراءة الذاكرة Memory Read (MEMR):** يقوم المعالج بتنشيط هذا الخط في حالة القراءة من الذاكرة سواء كانت RAM أو ROM.
2. **خط الكتابة في الذاكرة Memory Write (MEMW):** يقوم المعالج بتنشيط هذا الخط في حالة الكتابة في الذاكرة RAM.
3. **خط قراءة بوابة إدخال Input Port Read (IOR):** وهذا الخط يكون فعالًا عندما يكون المعالج في حالة استقبال معلومات من بوابة الإدخال.
4. **خط كتابة في بوابة إخراج Output Port Write (IOW):** وهذا الخط يكون فعالًا عندما يكون المعالج في حالة إرسال المعلومات إلى بوابة إخراج.

ولتوضيح أهمية ناقل التحكم نأخذ المثال الآتي: لو كان معالج ما يريد إرسال الرقم (34H) إلى الـ bits الذي عنوان E100H، فكيف سيعمل المعالج على ضوء دراستنا لوظيفة كل من ناقل العناوين والبيانات؟

لكي يقوم المعالج بالمهمة سيضع العنوان E100H على ناقل العناوين، وبذلك تصبح شريحة الذاكرة التي تحتوي هذا الـ bits نشطة (فعالة) وعلى إستعداد للتعامل مع المعالج، عند ذلك يقوم المعالج بوضع البيانات (34H) على ناقل البيانات فيتلقاها الـ bits المعني ويسجل فيه.

المشكلة التي ستظهر هي: أنَّ المعالج عندما قام بتنشيط شريحة الذاكرة التي تحتوي على هذه الـ bits لم يخبر الشريحة عما إذا كان سيرسل إليها معلومات أم سيستقبل منها، أي هل سيكتب فيها أم سيقرأ منها؟ هنا تظهر أهمية خط التحكم الذي يخرج من المعالج ليخبر الجهاز الذي سيتعامل معه المعالج أن الرقم (34H) هو في حالة إرسال أي يجب أن تكون إشارة ناقل التحكم هي (MEMW) أي كتابة الى الذاكرة.

وهناك خطان للتحكم بالطريقة نفسها للتعامل مع بوابات الإدخال والإخراج، وهناك أيضًا خطوط المقاطعة التي يتم بها مقاطعة أي برنامج يجري تنفيذه وخطوط المسك التي يتم بها فصل المعالج عن النواقل لأغراض معينة كما سيأتي شرحها لاحقاً.

ولا بُدَّ من الإشارة إلى أنَّ واحدًا فحسب من هذه الخطوط (خطوط التحكم) يكون فعالًا في أي لحظة، وبقية الخطوط تكون خاملة.

3-3 التخزين المؤقت في خطوط المعالج

تنشأ الحاجة إلى التخزين المؤقت (Buffering) عادة بين شيئين ينتج من إتصالهما المباشر بعض المشكلات، لذا يتم وضع وسيط يكون حلقة الوصل بين هذين الشيئين، وهذا الموقف يحدث في الكثير من الدوائر الالكترونية عند تحميل إحداها على الأخرى، فلو أنَّ الدائرة المصدر كانت غير قادرة على إدارة الدائرة الحمل أو تحميلها بسبب أنَّ الدائرة الحمل تحتاج إلى الكثير من التيار الذي لا تستطيع دائرة المصدر توفيره، فالذي سيحدث عندها هو أنَّ جهد خرج دائرة المصدر يضمحل أو يتلاشى، وبذلك تكون الدائرة غير قادرة على إدارة الحمل، والحل لهذه المشكلة هو إستعمال العازل Buffer بين الدائرتين.

في المعالج الدقيق جميع خطوطه الخارجة منه توصل على الكثير من الدوائر أو الشرائح الالكترونية على التوازي، فعلى سبيل المثال خطوط العناوين توصل على العديد من شرائح الذاكرة سواء كانت RAM أم ROM، والعديد من بوابات الإدخال والإخراج، وهذه الشرائح أجمعها تمثل أحمالًا بالنسبة إلى المعالج عليه الوفاء بحاجاتها من التيار، فعندما يكون خط العنوان الخارج من المعالج يحمل قيمة واحد (High) فإنَّ هذه الشرائح ستسحب تيارًا معينًا من المعالج لا بُدَّ من أن يستطيع توفيره، وعندما يكون خط العنوان الخارج من المعالج يحمل قيمة صفر (Low) فإنَّ هذه الشرائح ستصرف تيارات معينة لا بُدَّ من أن يكون المعالج قادرًا على صرفها.

بتجميع هذه التيارات الداخلة والخارجة على الخطوط المتناظرة لجميع الشرائح الموصلة على المعالج يمكن معرفة كم من التيار يجب على شريحة المعالج الدقيق أن توافره للشرائح الخارجية في حالة High، وكم من التيار يجب أن تصرفه في حالة Low بعد حساب مجموع التيارات المطلوب توفيرها أو صرفها من المعالج سيتم اتخاذ قرار بالحاجة إلى العازل أم لا وفقًا للحالات الآتية:

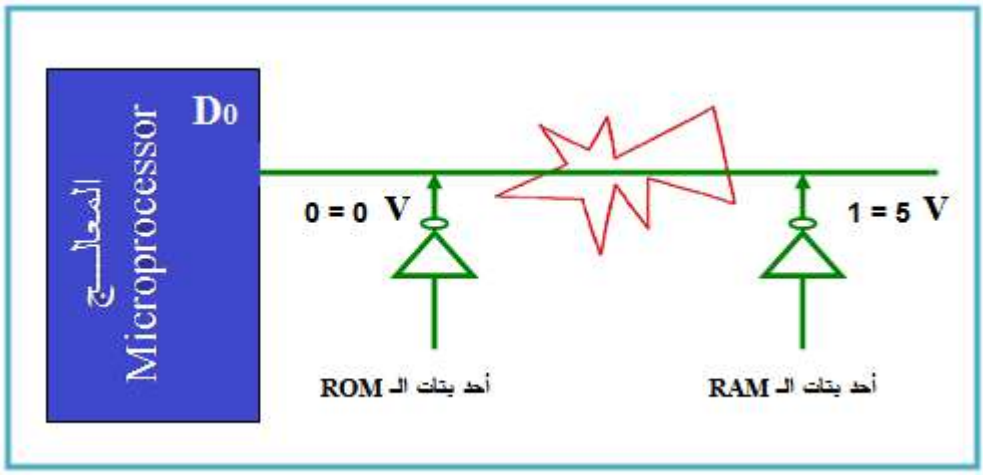
1. إذا كانت حاجات الأحمال من التيار ليست أقل مما يستطيع المعالج توفيره وبكمية كافية كعامل أمان فإنَّه في هذه الحالة لا بد من اللجوء إلى إستعمال عازل Buffer، ويجب عدم إهمال إستعمال العازل إذا كانت التيارات التي تحتاج إليها الأحمال أقل أو تكاد تساوي ما يستطيع المعالج الوفاء به فإنَّ هذا سيضع المعالج في المنطقة الحرجة ولا بد من توفير عامل أمان كافٍ، لأن ذلك بالتأكيد سيؤثر في تشغيل الدائرة في ما بعد.
2. إذا كانت المسافة طويلة بين الحمل والمعالج بحيث تظهر الحاجة إلى استعمال أسلاك توصيل Cables طويلة لإتمام عملية التوصيل، فإنَّه في هذه الحالة لا بُدَّ من استعمال دائرة عازل عند الخروج من المعالج وقبل السلك، لتجنب تشوه الموجة الخارجة من المعالج، وعمومًا إذا كان الحمل موجودًا على بطاقة (Card) غير البطاقة التي عليها المعالج فإنَّه في هذه الحالة يفضل استعمال العوازل على جميع النواقل بعد المعالج مباشرة.
3. هناك بعض المعالجات التي تعتمد فكرة المزج الزمني Time Multiplexing بين نواقلها مثل المعالج 8085 الذي يستعمل ثمانية الخطوط الأولى من ناقل العناوين كناقل للبيانات أيضًا بحيث إنَّ هذه الخطوط تحمل إشارة عناوين لمدة معينة من الزمن ثم بعد ذلك تحمل إشارة بيانات، ولمثل هذا النوع من المعالجات لا بُدَّ من إجراء عملية عزل لإشارة البيانات وحدها لتكون على ناقل خاص ولإشارة العناوين وحدها لتكون على ناقل آخر وذلك قبل توصيل الأحمال على هذه النواقل، وإذا تم ذلك فسيتم استعمال شرائح تقوم بعملية عزل للإشارات كل على ناقل خاص، وفي الوقت نفسه تكون هذه الشرائح عوازل تفي بأغراض الأحمال من التيارات.

ونظراً لكثرة الشرائح التي يمكن استعمالها كعوازل، هناك نقاط يجب أن تؤخذ بالحسبان عند اختيار الشريحة التي ستستعمل كعازل، وهي:

1. يجب أن يكون العازل قادرًا على الإيفاء بالتزامات التيار المطلوبة للأحمال وإلا فلا فائدة من إستعماله.
2. يجب أن يكون المعالج قادرًا على إدارة جميع العوازل المركبة على خطوطه وإلا فلا فائدة من إستعمالها أيضًا.
3. يجب أن لا تؤثر العوازل المستعملة على طبيعة الإشارة التي يتم نقلها، فهناك عوازل من طبيعتها أنها تعكس الإشارة (تجعل الواحد صفرًا والعكس).
4. يجب أن يناسب العازل طبيعة الإشارة التي ستمر من خلاله، فهناك مثلًا نواقل أحادية الاتجاه بمعنى أنَّ الإشارة عليها تمر في اتجاه واحد فحسب مثل ناقل العناوين الذي تكون الإشارة عليه دائمًا من المعالج إلى الأجهزة الخارجية لذلك يجب على العازل أن يسمح بذلك، كذلك فإنَّ طبيعة أشارة ناقل البيانات بأنها تمر في كلا الاتجاهين من المعالج وإليه، لذلك يجب على العازل المستعمل أن يأخذ ذلك بالحسبان.

3-3-1 أنواع التخزين المؤقت Buffering Types

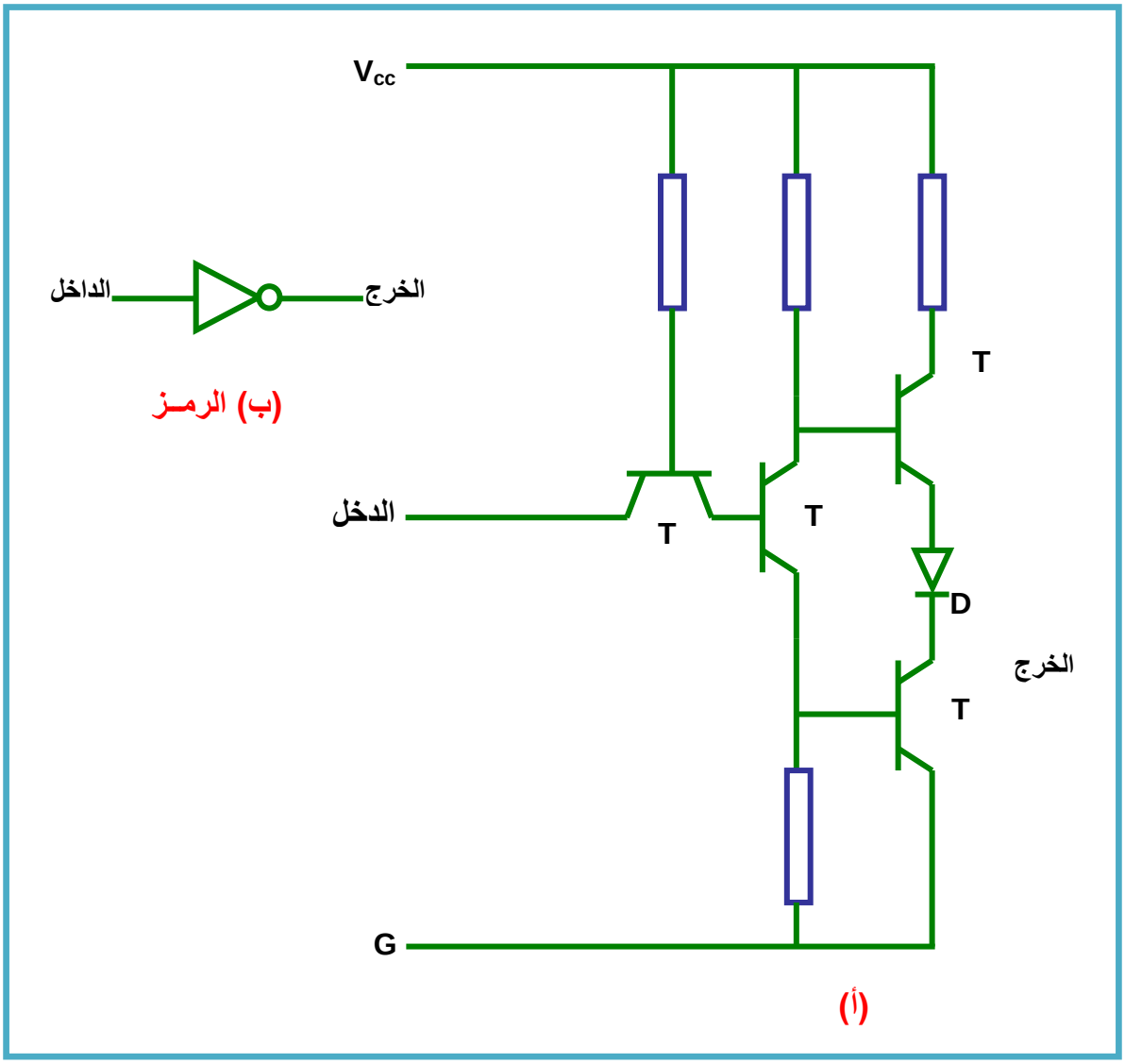
كما ذكرنا سابقًا أنَّ أي ناقل من نواقل المعالج هو عبارة عن مجموعة من الخطوط المتوازية التي تخرج من المعالج وموصل عليها على التوازي الكثير من الأجهزة الخارجية، في حالة المنطق الثنائي، إذ تكون الإشارة الخارجة من الأجهزة إما صفرًا أو واحدًا، فلذلك فإنَّ الخط D_0 من ناقل البيانات ربما يكون عليه (0) خرج من الـ RAM وفي الوقت نفسه يكون عليه (1) خرج من الـ ROM، وكما نعلم فإنَّ الواحد المنطقي في نظام منطق الترانزستور الترانزستور المعروف بـ Transistor Transistor Logic (TTL) يعني قيمة جهد أكثر من 2.4 V والصفر المنطقي يعني قيمة جهد أقل من 0.4 V ووجود هذين الجهدين على الخط نفسه وفي الوقت نفسه يعني قصراً كهربائياً Short Circuit، مما يؤدي إلى تخريب لمرحلة الخرج في أحد الجهازين إن لم يكن كليهما، ويظهر الشكل (2-3) خط البيانات D_0 وقد حدث عليه قصر كهربائي نتيجة توصيل الجهازين اللذين خرجهما عبارة عن مرحلة ثنائية المنطق أي تكون إما صفرًا وإما واحدًا.



شكل 3-2 قصر كهربائي على الخط D_0

وهذه المشكلة مصاحبة للأجهزة التي تقوم بإدخال معلومات إلى المعالج عن طريق ناقل البيانات مثل بوابات الإدخال والـ RAM والـ ROM فحسب، أمَّا الأجهزة التي تستقبل معلومات من المعالج مثل بوابات الإخراج فلا تعاني من هذه المشكلة؛ والحل لهذه المشكلة هو استعمال أحد أنواع التخزين المؤقت، وهي البوابات ثلاثية المنطق Tristate Logic Gate بدلًا عن البوابات الثنائية المستعملة Double State Logic Gate، وقبل دراسة أول نوع من العوازل وهي البوابات ثلاثية المنطق سيتم توضيح طريقة عمل البوابات ثنائية المنطق.

يبين الشكل (3-3) تركيب ورمز بوابة من البوابات ثنائية المنطق وهي بوابة NOT، هذه البوابة دائمًا يكون خرجها إما واحدًا وإما صفرًا بحسب حالة الدخل، إذا كان الدخل يساوي واحدًا H فإنَّ الترانزستور T_4 يكون مجمعه H وبالتالي فإنَّ T_3 يكون ON ويكون باعثه H مما يجعل T_2 يكون ON وبالتالي فإنَّ الخرج يكون موصلًا إلى الأرضي أي يكون الخرج صفرًا في هذه الحالة، مع ملاحظة أنَّ T_1 في هذه الحالة يكون OFF أي غير موصل، ولذلك فإنَّ الخرج يكون معزولًا من مصدر الجهد V_{cc} إذا كان الدخل صفرًا L فإن مجمع T_4 يكون L، وبالتالي T_3 يكون OFF مما يجعل T_2 يكون OFF، وبالتالي T_1 يكون ON، وعلى ذلك فإنَّ الخرج في هذه الحالة يكون متصلًا بمصدر الجهد V_{cc} ويكون واحدًا أي H وفي الوقت نفسه ينفصل عن الأرضي.



شكل 3-3 بوابة عاكس NOT ثنائية المنطق

وبعد أن أوضحنا واحدة من البوابات ثنائية المنطق سنتطرق لأول أنواع العوازل:

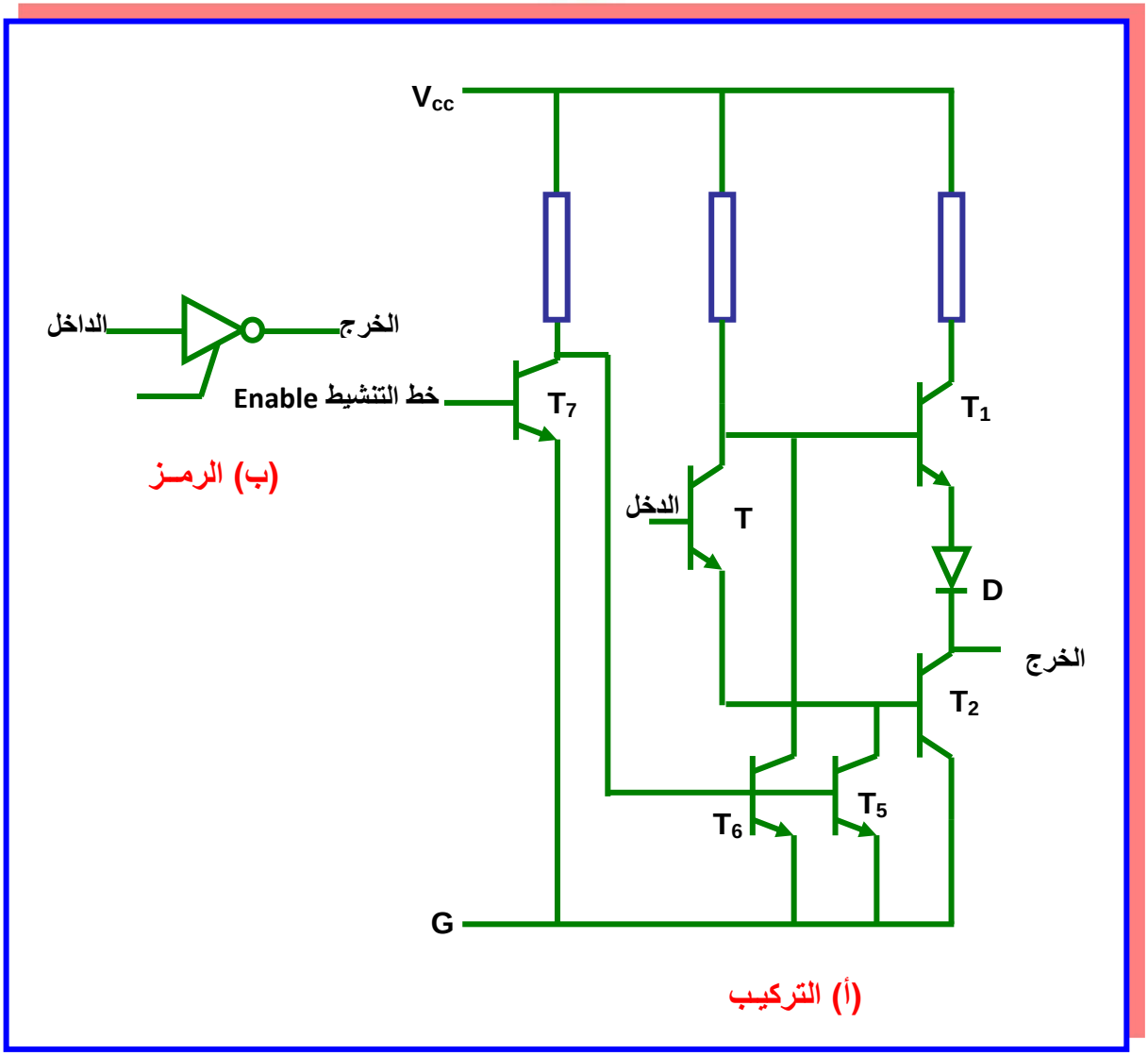
(1) البوابات ثلاثية المنطق Tristate Logic Gate

تتميز البوابات ثلاثية المنطق إنَّ لها طرفًا ثالثًا خاصًا بالتحكم في الخرج بحيث إذا كان هذا الطرف فعالًا فإنَّ البوابة ثلاثية المنطق تأخذ حالة جديدة غير معروفة في البوابات ثنائية المنطق، وهي أنَّ الخرج لا يكون صفرًا ولا واحدًا، وإنما يكون مفتوحًا Open Circuit أو مقاومة عالية جدًا High Impedance.

يوضح الشكل (3-4) الدائرة الالكترونية والرمز المستعمل لمرحلة خرج بوابة ثلاثية المنطق، وقد تم التحكم فيها عن طريق خط التنشيط Enable، ويمكن ملاحظة أنَّ مرحلة الخرج هذه مشابهة لمرحلة الخرج التي تم شرحها في الشكل (3-3) ولكن تم إضافة الترانزستورات الآتية: (T_5, T_6, T_7) إليها التي تعمل كمفاتيح يتم التحكم فيها عن طريق خط التنشيط Enable، فإذا كان خط التنشيط عاليًا H فإنَّ T_7 يكون ON مما يجعل T_5, T_6 كل منهما يكون OFF، وبالتالي فإنَّ البوابة تعمل كبوابة ثنائية المنطق مثل التي تم شرحها في الشكل (3-3)، بحيث إذا كان الدخل H فإنَّ الخرج يكون L والعكس.

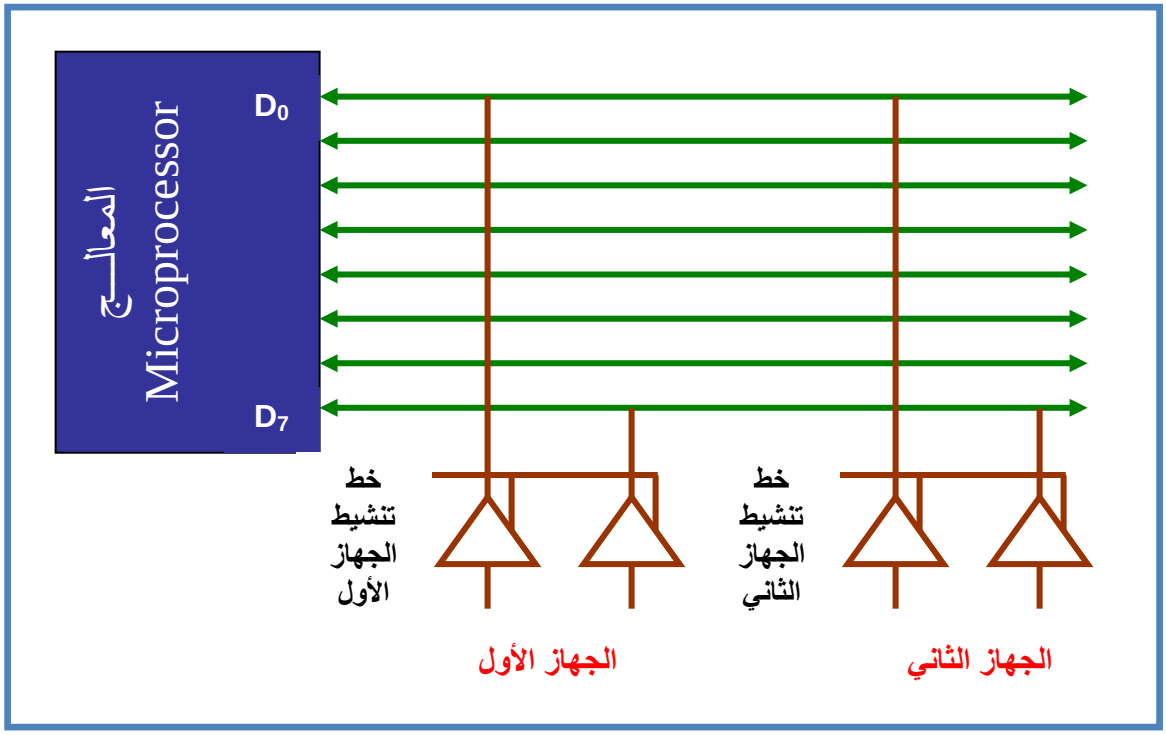
أمَّا إذا كان خط التنشيط خاملًا L فإنَّ T_7 يكون OFF مما سيجعل T_5, T_6 كل منهما يكون ON، وبالتالي فإنَّ كلاً من T_1, T_2 يكون OFF، وهكذا فإنَّ الخرج يكون غير موصل لا على الأرضي ولا على الجهد V_{cc} ولكن يكون كما لو كان مفتوحًا Open Circuit أو مقاومة عالية.

ليتم استعمال البوابات ثلاثية المنطق في الحماية من القصر الكهربائي الذي يحدث بسبب توصيل أكثر من جهاز على خطوط النواقل أنفسها كما في الشكل (3-4) فإن جميع الأجهزة التي ستوصل على ناقل البيانات للمعالج يجب أن تكون مرحلة الخرج فيها عبارة عن بوابات ثلاثية المنطق، وعن طريق خطوط التنشيط لكل جهاز سيجعل المعالج جميع الأجهزة في حالة خمول، أي إن خرجها سيكون كما لو كان غير موصل على الناقل إلى جهاز واحد فحسب وهو الجهاز الذي يتعامل معه المعالج في تلك اللحظة، أي إنَّه نتيجة استعمال هذا النوع من البوابات لن يكون هناك غير جهاز واحد فحسب هو الفعال في أي لحظة وهو الذي يتعامل معه المعالج وهو الذي سيكون موصلًا على ناقل البيانات، وأما بقية الأجهزة فستكون منفصلة عن ناقل البيانات نتيجة أن خط التنشيط الخاص بها غير فعال.



شكل 3-4 بوابـات ثلاثية المنطق

يبين الشكل (3-5) عملية توصيل أكثر من جهاز على ناقل البيانات بواسطة البوابات ثلاثية المنطق، والجدير بالذكر أنَّ هناك أنواعاً مختلفة للبوابات ثلاثية المنطق، فمثلًا هناك بوابات يكون خرجها مثل دخلها تمامًا إذا كان خط التنشيط فعالًا، في حين هناك بوابات يكون خرجها عكس دخلها إذا كان خط التنشيط فعالًا، وهناك أيضًا البوابات التي يكون خط تنشيطها فعالًا عندما يكون صفرًا، وأخرى يكون خط تنشيطها فعالًا عندما يكون واحدًا، ويعتمد إختيار البوابة المناسبة على طبيعة التطبيق المستعمل.



شكل 3-5 إستخدام البوابات ثلاثية المنطق في التوصيل على ناقل البيانات في المعالج

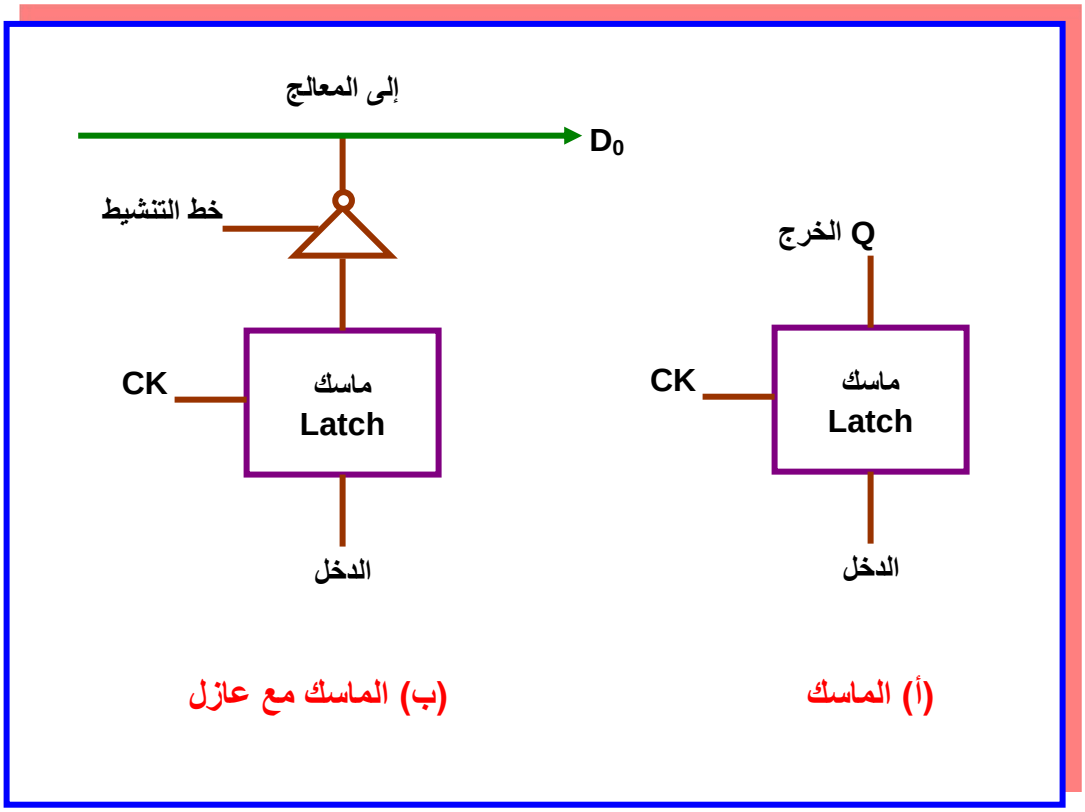
(2) الماسك Latch

تتطلب عملية العزل في بعض التطبيقات استعمال مكونات لها خاصية مسك المعلومة على خرجها بالرغم من فقدان هذه المعلومة على الدخل، والبوابات ثلاثية المنطق ليست لها هذه الخاصية، لأنها طالما خط التنشيط فعالًا تكون هناك معلومات معينة على الخرج، وبمجرد أن يكون خط التنشيط غير فعال تفقد المعلومة التي على الخرج تمامًا.

إنَّ الماسك عبارة عن قلاب Flip-Flop وغالبًا يكون من النوع D بحيث إنَّ المعلومة التي على طرف الدخل D تنتقل إلى الخرج Q بعد وجود نبضة على طرف التزامن CK، تبقى المعلومة الموجودة على الخرج كما هي لا تتغير حتى لو تغير الدخل D طالما أنَّه لم يعطي أي نبضة تزامن أخرى, لذلك يُقال في هذه الحالة إنَّ المعلومة قد مسكت على الخرج.

يوضح الشكل (3-6) الحالة (أ) شكل الماسك، إذ إن هناك الكثير من التطبيقات على سبيل المثال أجهزة إدخال البيانات إلى المعالج التي تتطلب وجود بوابة ثلاثية المنطق بعد الماسك لتكون بمنزلة عازل بين ناقل البيانات وخرج الماسك, لأن الماسك وحده لا يمكن توصيله على ناقل البيانات مباشرة, لأن خرجه يأخذ أحدى الحالتين فحسب، إما صفر وإما الواحد أي ثنائي المنطق,

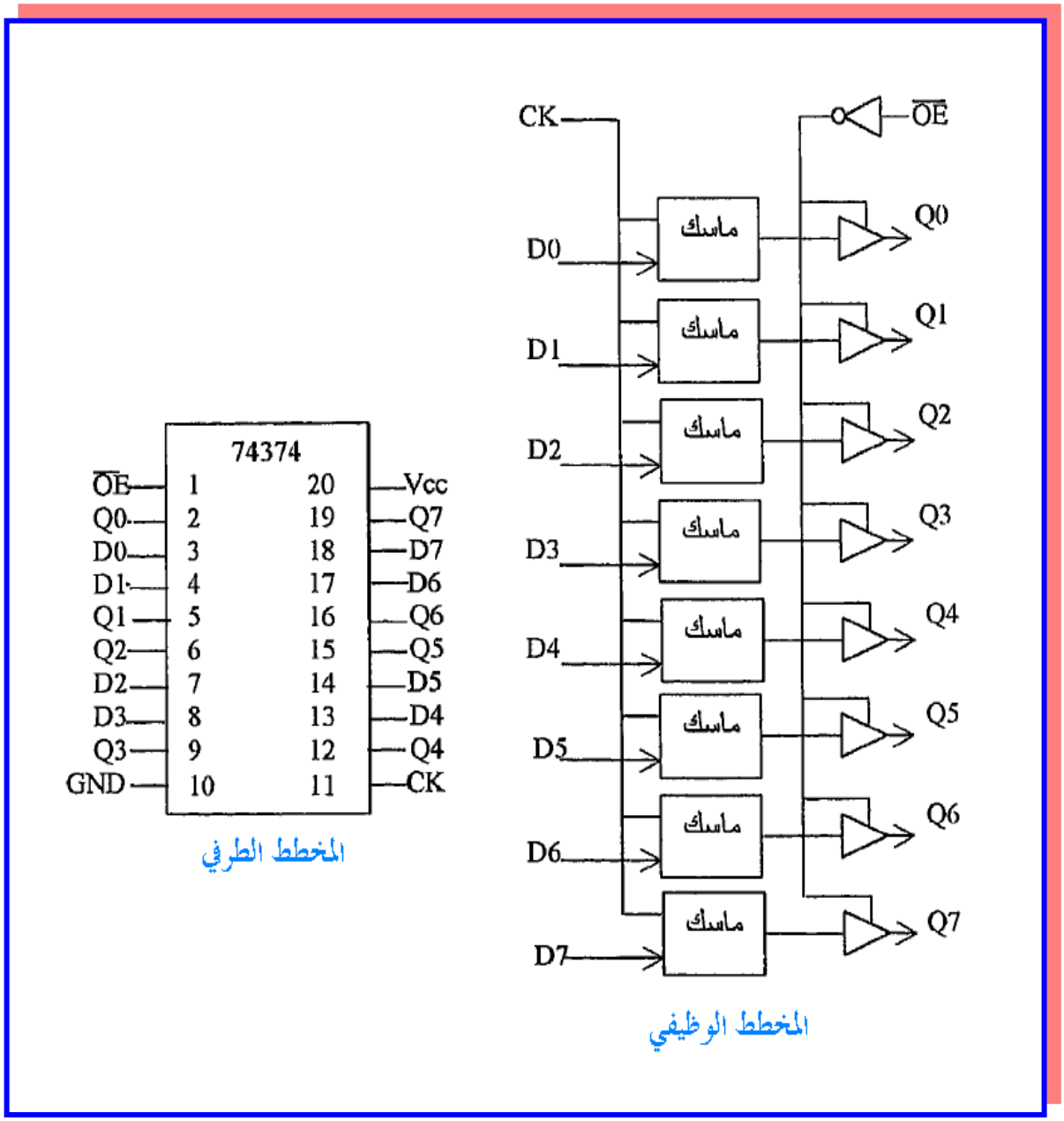
ولذلك عادة يوضع بعد الماسك بوابة ثلاثية المنطق بحيث يوصل خرج الماسك على ناقل البيانات فحسب، عندما يكون خط تنشيط البوابة ثلاثية المنطق فعالًا، يوضح الشكل (3-6) الحالة (ب) دائرة ماسك متبوعة ببوابة ثلاثية المنطق.



شكل 3-6 الماسك

حالــة دراسيــة رقم (1)

لتوضيح عمل بوابة ثلاثية المنطق مع الماسك سنأخذ شريحة تستعمل لعزل النواقل وهي الشريحة المعروفة بـ (74374): وهي عبارة عن عازل ماسك ذي ثمانية بتات تتكون هذه الشريحة من ثمانية قلابات من النوع (D Flip-Flop) كلها موصلة على طرف التزامن CK نفسه وهو الطرف رقم 11 في الشريحة، فعند إعطاء نبضة تزامن على هذا الطرف تنتقل الإشارة الموجودة على جميع أطراف الدخل D إلى الخرج المناظر Q، وتظل هذه الإشارة على الخرج طالما لم يتم أعطاء أي نبضات تزامن أخرى، كل واحد من هذه القلابات موصل على بوابة ثلاثية المنطق تسمح بمرور الإشارة إلى أطراف الخرج عندما يكون طرف التنشيط (الطرف 1) QE فعالًا (صفر)، وهذه الشريحة تستعمل في المعالج 8085 لغرض فصل ناقل العناوين، ويبين الشكل (3-7) الرسم الطرفي والوظيفي لشريحة 74374.



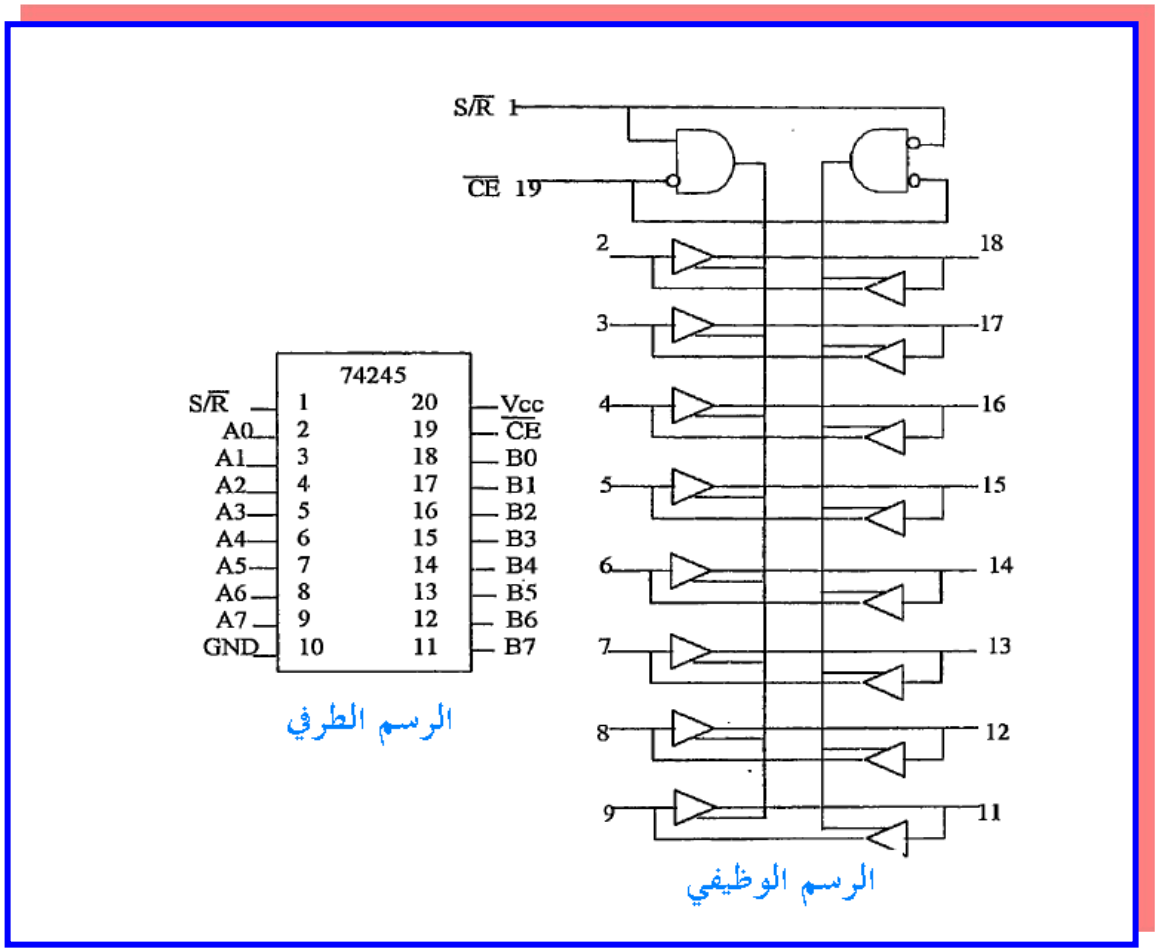
شكل 3-7 المخطط الوظيفي والطرفي لشريحة العازل 74374 (للأطلاع)

حالــة دراسيــة رقم (2)

لتوضيح عمل عازل ثماني ذي بوابة ثلاثية المنطق ثنائي الاتجاه سندرس شريحة (74245):

وهي عبارة عن عازل ثنائي الاتجاه أي يمكنه إرسال المعلومات واستقبالها، وخرجها ثلاثي المنطق غير عاكس ولها خط تحكم في الاتجاه وهو الطرف رقم 1، الطرف رقم 19 هو خط تنشيط أو فعالية الشريحة الذي يكون فعال عندما يكون صفرًا، إذ يرمز له بالرمزChip Enable-CE.

يوضح الشكل (3-8) التخطيط الطرفي والوظيفي للشريحة، عندما يكون الطرف 19 فعالًا (صفر) فإنَّ الشريحة تكون نشطة، وعندما يكون هذا الطرف خاملًا (واحد) فإنَّ الشريحة تكون خاملة ولا تعمل، الطرف 1 يتحكم في اتجاه البيانات خلال الشريحة، فعندما يكون هذا الطرف واحدًا فإنَّ الإشارات تمر في الاتجاه من A إلى B، وأمَّا إذا كان هذا الخط صفراً فإنَّ الإشارة تمر في الاتجاه من B إلى A, لذلك فإنَّ الشريحة مناسبة جدًا لفصل خطوط ناقل البيانات.



شكل 3-8 المخطط الوظيفي والطرفي لشريحة العازل 74245 (للأطلاع)

4-3 طرائق انتقال البيانات بين الأجزاء الداخلية للحاسوب

من دراستنا السابقة نعلم أنَّ المعالج الدقيق يحتاج إلى التواصل مع الأجزاء الأخرى للحاسوب سواء كانت داخلية مثل: (السجلات Registers، ووحدة الحساب والمنطق ALU، والذاكرة Memory) أم خارجية مثل: (وحدات الإدخال والإخراج الطرفية I/O Peripheral Devices).

وهذا الاتصال يتم عن طريق ثلاثة نواقل داخلية (Internal Buses) تم دراستها في مقدمة الفصل، وهي: ناقل البيانات (Data Bus) وناقل التحكم (Control Bus) وناقل العنوان (Address Bus)، وكذلك في حالة الاتصال بالأجهزة الخارجية فلدينا ما يُعرف بواجهات الإدخال والإخراج (Input/Output Interface).

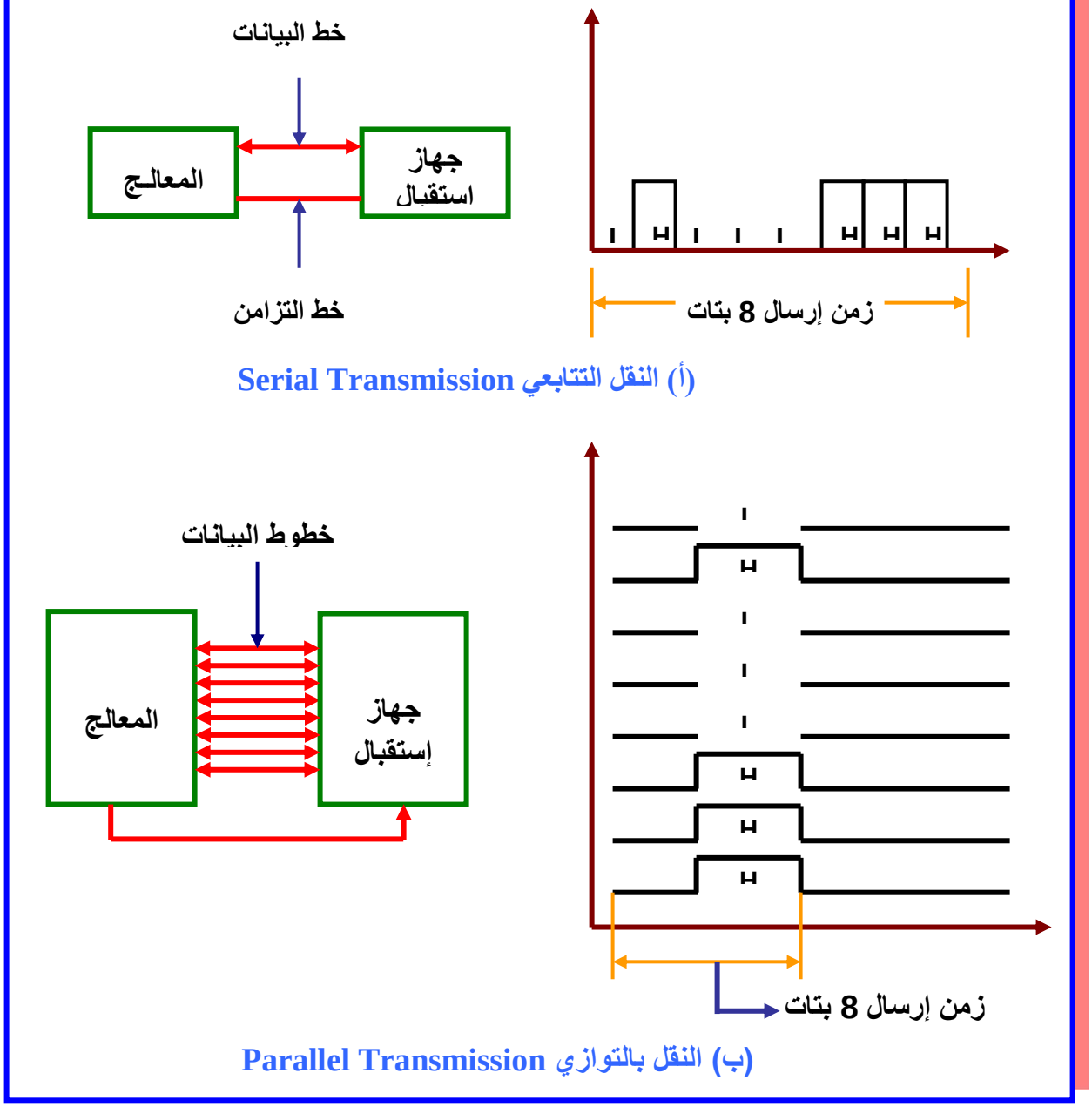
أمَّا آلية النقل فتتمثل بطريقتين رئيستين يتم بها نقل المعلومات في المعالجات، وهي:

1. النقل المتتابع للمعلومات Serial Data Transfer.
2. النقل المتوازي للمعلومات Parallel Data Transfer.

وكُل من الطريقتين لها مميزاتها والاستعمالات الخاصة التي تحتاج إليها بالضرورة، ففي الطريقة التتابعية (Serial) يتم إرسال البيانات من الأجهزة الخارجية وإليها على خط واحد فحسب ولا يرسل على هذا الخط إلا بت (bit) واحد فحسب في وحدة الزمن نفسها وهي الـ (Clock)، بحيث إنَّه لكي يتم إرسال معلومة من ثمانية بتات مثلًا فهناك حاجة إلى زمن مقداره ثماني نبضات تزامن لكي يتم إرسال المعلومة.

أمَّا في الطريقة المتوازية (Parallel) فإنَّ المعلومات يتم إرسالها من الحاسوب على أكثر من خط واحد، وعادة ما يكون عدد هذه الخطوط مساوياً عدد الخطوط في ناقل البيانات (Data Bus) للحاسوب الذي يتم التعامل معه، وفي هذه الحالة لكي يتم إرسال معلومة من ثمانية بتات مثلًا هناك حاجة إلى ثمانية خطوط متوازية بحيث ترسل كل بت على خط منفصل من هذه الخطوط، وبالتأكيد فإنَّه في هذه الحالة سترسل جميع هذه البتات في نبضة تزامن (Clock) واحدة فحسب، لذلك تعدَ هذه الطريقة أسرع بكثير من الطريقة التتابعية في إرسال المعلومات.

يبين الشكل (3-9) رسمًا توضيحيًا لطرائق إرسال المعلومات بالطريقتين التتابعية والمتوازية.



(أ) النقل التتابعي Serial Transmission

(ب) النقل بالتوازي Parallel Transmission

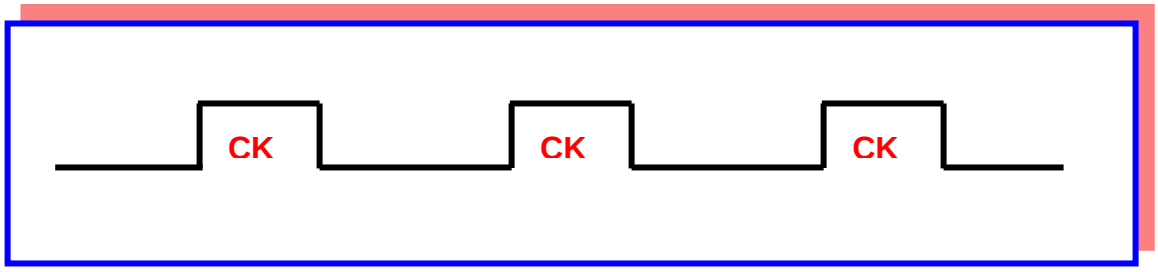
شكل 3-9 نقل المعلومات بالطريقة المتوازية والمتتابعة

يتم نقل المعلومات بين المعالج والأجزاء الداخلية للحاسبة مثل: السجلات، ووحدة الحساب والمنطق، والذاكرة، بواسطة ثلاث طرائق، وهي:

1. الطريقة التوافقية لنقل البيانات (المتزامنة) Synchronous Data Transfer.
2. الطريقة غير التوافقية لنقل البيانات (غير المتزامنة) Asynchronous Data Transfer.
3. الطريقة شبه التوافقية لنقل البيانات (شبه المتزامنة)Semi-synchronous Data Transfer.

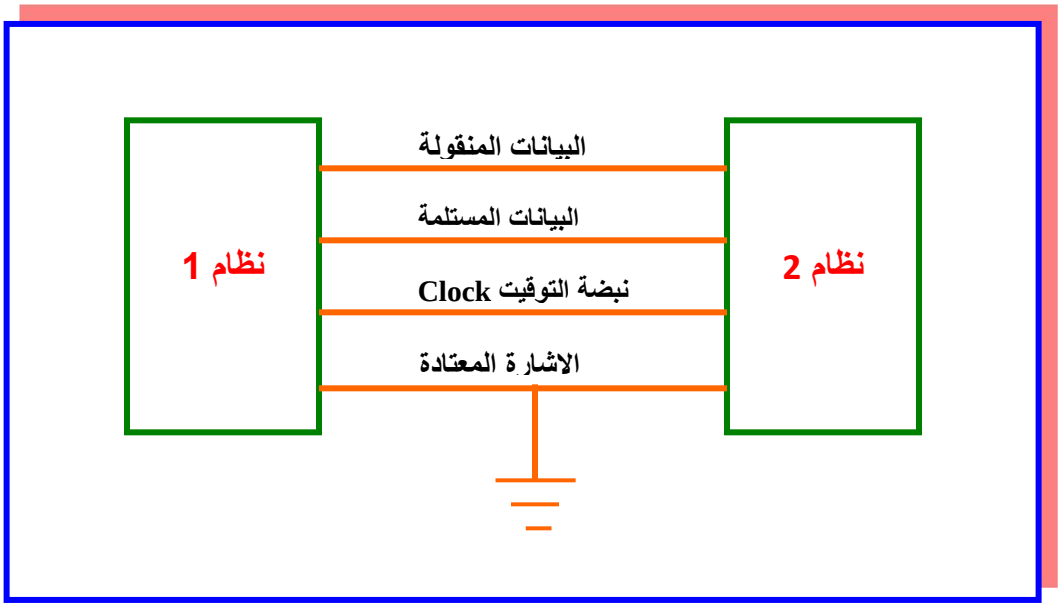
3-4-1 الطريقة التوافقية (المتزامنة)

تعد هذه الطريقة واحدة من أسهل وأيسر الطرائق في نقل البيانات بين الأجزاء الداخلية للحاسوب والمعالج، وتعتمد على نبضة التزامن (Clock Pulse – CK) كما في الشكل (3-10) أو النبضة الداخلية في المعالج الدقيق، فتكون سرعة انتقال البيانات معتمدة على سرعة النبضات.



شكل 3-10 نبضة التزامن

وتعتمد الطريقة التوافقية على أنَّ المتسلم (Receiver) والمرسل (Transmitter) للقطعتين اللتين يتم الاتصال بينهما لنقل المعلومات يجب أن يعملا متزامنين كما في الشكل (3-11)، إذ نجد الواجهة (Interface) تحتوي على خط النبضة (Clock Line) وكذلك البيانات المنقولة (Transmit Data) والبيانات المتسلمة (Receive Data) وخطوط الإشارة المعتادة (Common signal)، فإشارة النبضة هي التي تزامن كِلا البيانات المنقولة والمتسلمة.



شكل 3-11 واجهة الاتصال المتزامن

الصيغة المعتمدة في الطريقة المتزامنة أو التوافقية لنقل البيانات موضحة في الشكل (3-11)، فلإنشاء نقل متزامن للبيانات يحدث الآتي:

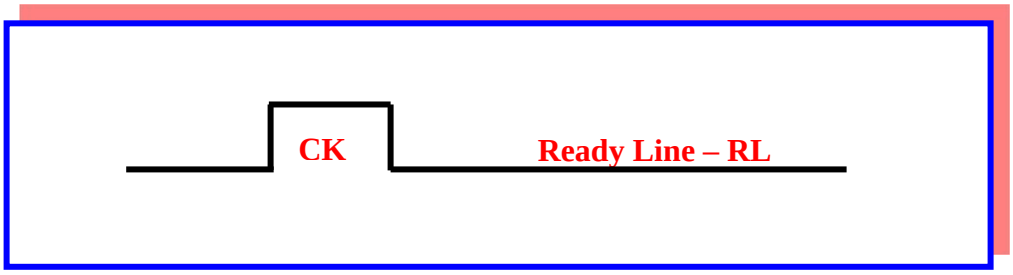
1. يقوم المرسل بإرسال رموز متزامنة (Synchronization Characters) إلى المتسلم.
2. المتسلم يقرأ أنموذج (Pattern) البت المتزامن ويقارنه ببت متزامن معروف.
3. في حالة التطابق بين الأنموذج المُرْسَل والمُتسلم يبدأ الأخير بقراءة البيانات من خط البيانات.
4. نقل البيانات يستمر إلى أن يتم إكمال قراءة كتلة من البيانات المتسلمة.
5. في حالة نقل كتل بيانات كبيرة، فإنَّ الرموز المتزامنة ربما يتم إعادة إرسالها دورياً لضمان التزامن.

ان النقل بالطريقة المتزامنة يستعمل عادةً في التطبيقات التي تحتاج إلى سرعة عالية جدًا في نقل البيانات.

3-4-2 الطريقة غير التوافقية (غير المتزامنة)

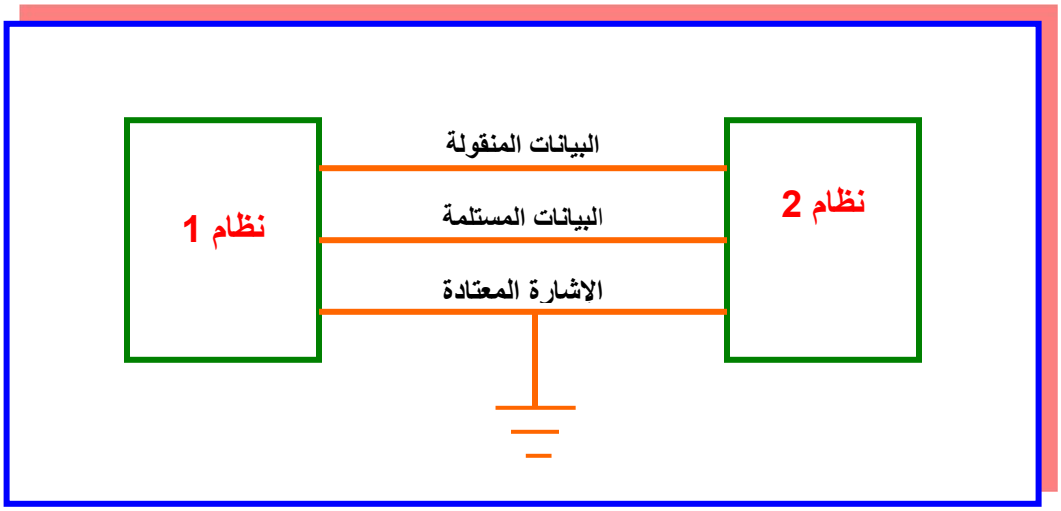
لا تعتمد هذه الطريقة على نبضة التوقيت (Clock Pulse – CK) الموجودة داخل الحاسوب وإنما تعتمد على خط الاستعداد (Ready Line – RL) كما في الشكل (3-12)، وفي

هذه الطريقة تنتظر البيانات إلى أن تصبح قيمة خط الاستعداد واحدًا للدلالة على إمكانية البدء بنقل المعلومات، أمَّا في حالة كونها صفرًا فلا يكون هناك أي نقل للبيانات.



شكل 3-12 خط الاستعداد

أيسر واجهة للطريقة غير المتزامنة موضحة في الشكل (3-13)، إذ يتكون من: البيانات المنقولة (Transmit Data) والبيانات المتسلمة (Receive Data)، وخطوط الإشارة المعتادة (Common Signal).



شكل 3-13 واجهة الاتصال غير المتزامن

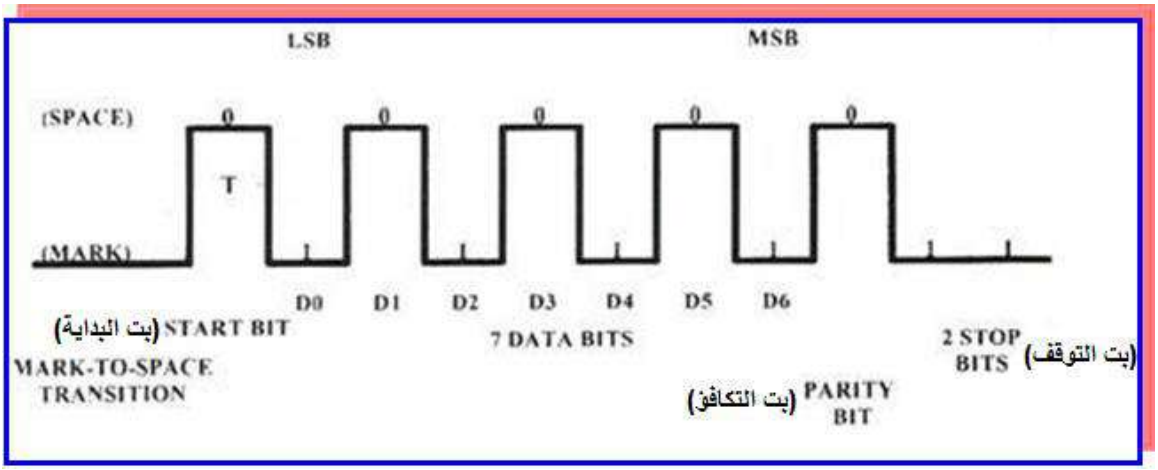
وفي هذه الطريقة يتم نقل البيانات على شكل رمز واحد في كل مرة، والمُتسلم يقوم بفحص بتات التزامن التي تكون موجودة في بداية كل رمز ينفذ نهاية التزامن لخط النقل ونهايته.

ويوضح الشكل (3-14) الصيغة المعتمدة في الطريقة غير المتزامنة أو غير التوافقية لنقل البيانات، إذ نرى الآتي:

1. يتم وضع بت التزامن في بداية الرمز المراد نقله، ويسمى (بت البداية Start Bit)، أمَّا البت الذي يوضع في نهاية الرمز، فيسمى (بت التوقف Stop Bit).
2. بتات الرمز الذي يتم نقله توضع بين بت البداية وبت التوقف.

3. بت البداية يدخل أو يخرج في البداية، بينما البت ذو القيمة الأدنى للرمز (LSB)، وما تبقى من بتات الرمز، والبت ذو القيمة الأعلى (MSB)، وبت التكافؤ (Parity Bit)، وبت التوقف (Stop Bit) أجمعها تأتي بعد ذلك بالتتابع.
4. إنَّ بداية نقل البيانات وتوقفها يعتمد على قيمة بت البداية التي تمثل قيمة خط الاستعداد فإن كانت القيمة (1) ويعبر عنه بـ (Space) فهذا يعني البدء بعملية النقل، وأنَّه لا بد من عمل تزامن من المتسلم إلى المرسل والإشارة إلى الوحدة التي تتسلم البيانات بأنه يجب عليها الآن البدء بترميز البيانات المتجهة لها، أما في حالة كون بت التوقف قيمته (0) ويعبر عن ذلك بـ (Mark) فهذا يعني التوقف عن نقل البيانات.

تعتمد هذه الطريقة على أنَّ وحدة التسلم دائمًا تتأكد من قيمة خط الاستعداد التي يمكن معرفتها عن طريق بت البداية.



شكل 3-14 إنتقال البيانات بالطريقة غير المتزامنة

3-4-3 الطريقة شبه التوافقية (شبه المتزامنة)

تعتمد هذه الطريقة على نبضة التوقيت (Clock Pulse – CK) الموجودة داخل الحاسوب وأيضًا على خط الاستعداد (Ready Line – RL)، كما في الشكل (3-15)، وفي هذه الطريقة لكي تنقل المعلومات بين نبضة ونبضة أخرى لا بُد من وجود وقت انتظار حتى يصبح خط الاستعداد قيمته واحدًا (يعطي نبضة توقيت أخرى) وعندها يبدأ نقل المعلومات.



شكل 3-15 نبضة التزامن مع خط الاستعداد

3-5 طرائق انتقال المعلومات بين الحاسوب والأجهزة الطرفية

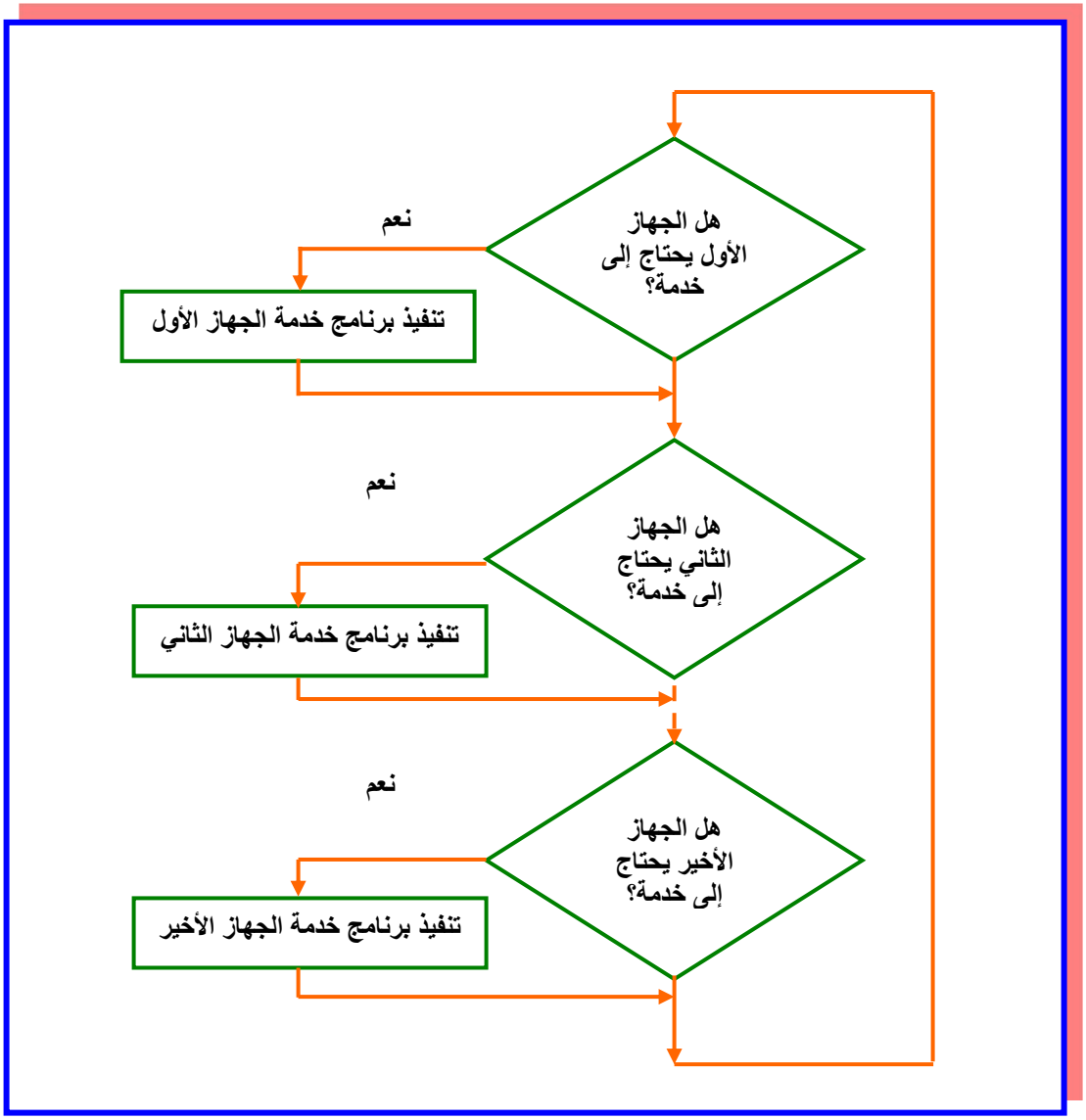
يتم نقل المعلومات بين المعالج والأجزاء الطرفية للحاسوب مثل: الطابعات، ولوحة المفاتيح، والشاشة وغيرها من الأجهزة الطرفية بواسطة طريقتين سندرسهما في هذا الفصل، وهما:

1. طريقة الإستجواب.
2. طريقة المقاطعة.

3-5-1 طريقة الاستجواب Request

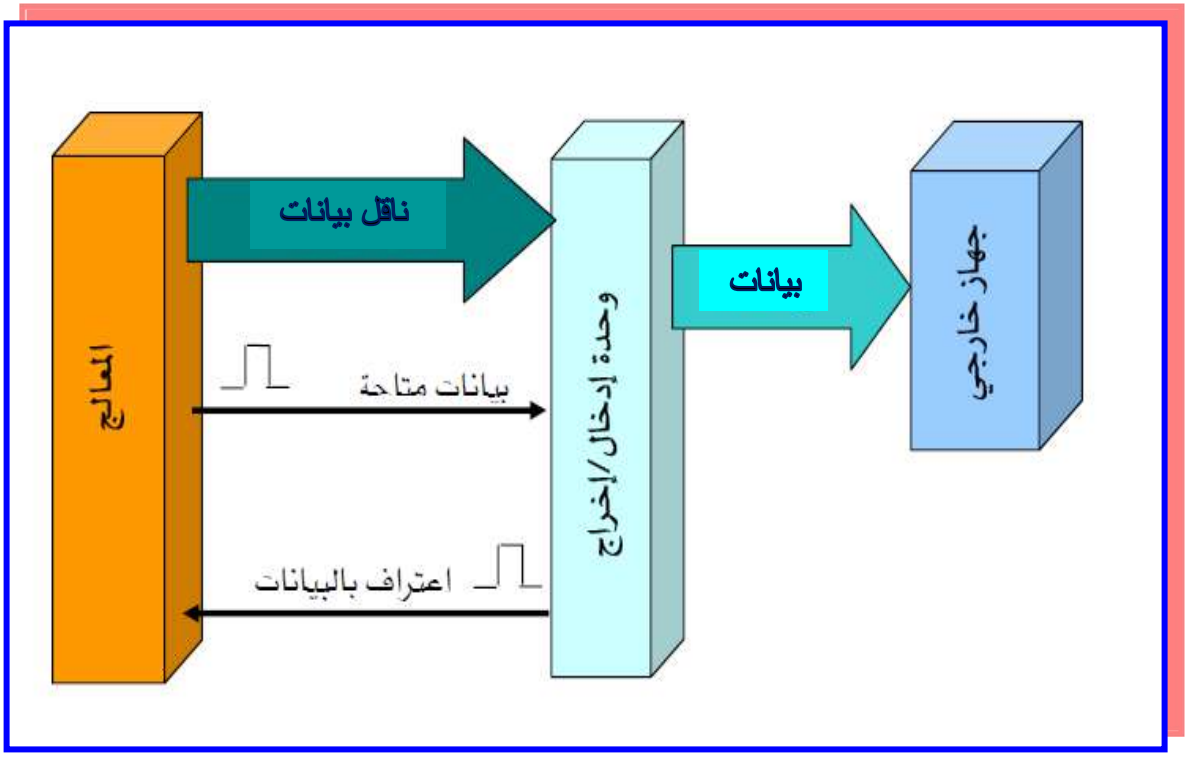
تسمى هذه الطريقة بطريقة خدمة الأجهزة المحيطة (Polling Service) فضلاً عن تسميتها بطريقة التصافح بالأيدي (Acknowledge Handshake) في بعض المصادر، وهذا الاختلاف في التسميات لا يؤثر في مضمون الطريقة التي تعتمد على أنَّ المعالج يقوم بطرق أبواب جميع الأجهزة المحيطة بالتتابع و يستفسر منها هل هناك خدمة يحتاجها ذلك الجهاز لكي يقوم المعالج بأدائها له، فإن كانت الإجابة بنعم فإنَّ المعالج يقوم بتنفيذ هذه الخدمة له وفي الحال من دون انتظار، أمَّا إذا كانت إجابة الجهاز بالنفي فإنَّ المعالج ينتقل إلى الجهاز التالي له ويوجه الاستفسار السابق نفسه، وهكذا إلى أن يصل المعالج إلى آخر جهاز، فإما أن يقوم بتقديم الخدمة له وإما لا وفقًا لإجابة الجهاز عن الاستفسار، وبعد آخر جهاز يعود المعالج إلى أول جهاز ويكرر العملية من جديد إلى ما لا نهاية, أي إنَّ الوظيفة الوحيدة للمعالج في هذه الحالة هي الدوران إلى ما لا نهاية على الأجهزة المحيطة وتقديم الخدمة لمن يطلبها، يمثل الشكل (3-16) خوارزمية لسير وظيفة المعالج في هذه الحالة. إنَّ من مميزات هذه الطريقة أنَّها سهلة البرمجة ولا تحتاج إلى الكثير من التجهيزات Hardware، ومن عيوبها أنَّ المعالج يكون مخصصًا لوظيفة خدمة هذه الأجهزة ولا يستطيع الفكاك منها، فالمبرمج عادة لا يستطيع الإفادة من المعالج في أي أغراض أخرى، ولا سيما إذا كان

عدد الأجهزة التي يقوم المعالج بالمرور عليها قليلًا وفي ذلك إهدارًا لفعالية المعالج، أمَّا إذا كان عدد الأجهزة التي يقوم المعالج بخدمتها بهذه الطريقة كبيرًا فإنَّ المعالج قد يتأخر على بعض الأجهزة التي تحتاج إلى خدمته على مدد متقاربة, لأن عليها الانتظار لحين أن يأتي دورها، كما أنَّها ليس من حقها أن تقاطع المعالج وتطلب الخدمة الفورية في حالات الضرورة، وكما نعلم هناك الكثير من المواقف التي تتطلب من المعالج التنفيذ فوراً كضرب جرس إنذار مثلًا عند حدوث أي طارئ في أي نظام تحكم مثل انقطاع التيار الكهربائي أو حدوث حريق أو ارتفاع زائد في ضغط غاز وغير ذلك من المواقف الكثيرة, لذلك فإنَّ المعالج يقدم للمبرمج نوعًا آخر من الخدمة وهي المقاطعة التي سنتطرق لها في الفقرة اللاحقة.



شكل 3-16 خوارزمية لبيان آلية عمل طريقة الاستجواب في نقل البيانات

هذه الطريقة تعتمد على إستعمال إشارات للتحكم في هذا النقل، وهذا النوع من إشارات التحكم يساعد على ضبط انتقال البيانات بين المعالج والأجهزة الطرفية، وآلية العمل بهذه الطريقة في حالة الإرسال موضحة في الشكل (3-17).



شكل 3-17 آلية عمل طريقة الاستجواب في نقل البيانات

تعتمد هذه الطريقة في النقل على أنَّ:

1. المعالج يجهز البيانات على ناقل البيانات ويرسل إشارة تدل على ذلك نحو وحدة الإدخال/ الإخراج.
2. الوحدة بعد استقبالها لإشارة بيانات متاحة تقوم بقراءة البيانات وترسل بدورها إشارة نحو المعالج لتخبره بذلك، المعالج عند استقباله لهذه الإشارة يمكن له إرسال بيانات أخرى وتجهيزها، وهكذا.

3-5-2 طريقة المقاطعة

في نظام خدمة الأجهزة الخارجية عن طريق المقاطعة لا يذهب المعالج إلى الأجهزة ويطرق بابها ليعرض عليها خدماتها وينتظر هل يحتاج الجهاز الى خدمة أم لا، بل إنَّ المعالج هنا يكون عادة مشغولًا في تنفيذ برنامج معين وعادة ما يكون هذا البرنامج لا نهائياً، فإذا احتاجت الأجهزة الى خدمة من المعالج فإنَّه يقاطعه ويطلب منه الخدمة فيقوم المعالج بتنفيذ هذه الخدمة

للجهاز المقاطع، وبعد الانتهاء من هذه الخدمة يعود المعالج لتنفيذ البرنامج الأساسي من حيث انتهى قبل المقاطعة.

وهذا المفهوم يمكن توضيحه بالمثال الآتي: هناك طالبان يتحادثان، يأتي طالب آخر ويقاطعهما ليستفسر عن موعد درس معين، في هذه الحالة يقطع المتكلم محادثته ليجيب السائل عن طلبه، ثم بعد الإجابة يستأنف حديثه الأول، إنَّ البرنامج الذي يقوم المعالج بتنفيذه عند المقاطعة يسمى ببرنامج خدمة المقاطعة.

ومن مميزات هذه الطريقة أنَّ الأجهزة المقاطعة تستطيع مقاطعة المعالج في أي وقت تريد، وليس عليها الانتظار إلى أن يأتي دورها مثل طريقة الاستجواب، وإذا حدثت وتمت المقاطعة في الوقت نفسه من أكثر من جهاز فإنَّ المعالج يخدمها بحسب أولويات تحدد له من المستعمل مسبقًا، ولكن ما الطريقة لذلك؟، في أثناء انشغال المعالج في تنفيذ برنامج خدمة مقاطعة معينة، هل يستطيع جهاز آخر أن يقاطعه؟ كيف يرجع المعالج إلى المكان نفسه في البرنامج الأصلي بعد انتهاء خدمة المقاطعة؟ هذه الأسئلة وغيرها سيتم معرفة إجابتها في دراستنا للآتي.

تكون المقاطعة عادة عن طريق إشارة يرسلها الجهاز المقاطع إلى المعالج على أحد أطرافه المخصصة لذلك، وعندما يكتشف المعالج هذه الإشارة فإنَّه يقوم بتنفيذ برنامج خدمة المقاطعة، وذلك يرجع لأن المعالج يقوم بقراءة جميع أطراف المقاطعة واختبارها قبل الدخول في تنفيذ أي أمر، وفي ما يلي بعض الأمثلة التي تستعمل فيها المقاطعة:

- ★ الأجهزة الخارجية مثل الطابعة ولوحة المفاتيح يمكنها أن تقاطع المعالج وترسل أي معلومات أو تستقبلها.
- ★ يمكن في أي وقت مقاطعة أي برنامج يتم تنفيذه إذا كان هذا البرنامج ينفذ بطريقة خاطئة.
- ★ يمكن للعمليات الصناعية التي يتم مراقبتها بواسطة المعالج أن تقاطعه في أي لحظة طوارئ تحدث للعملية الصناعية.

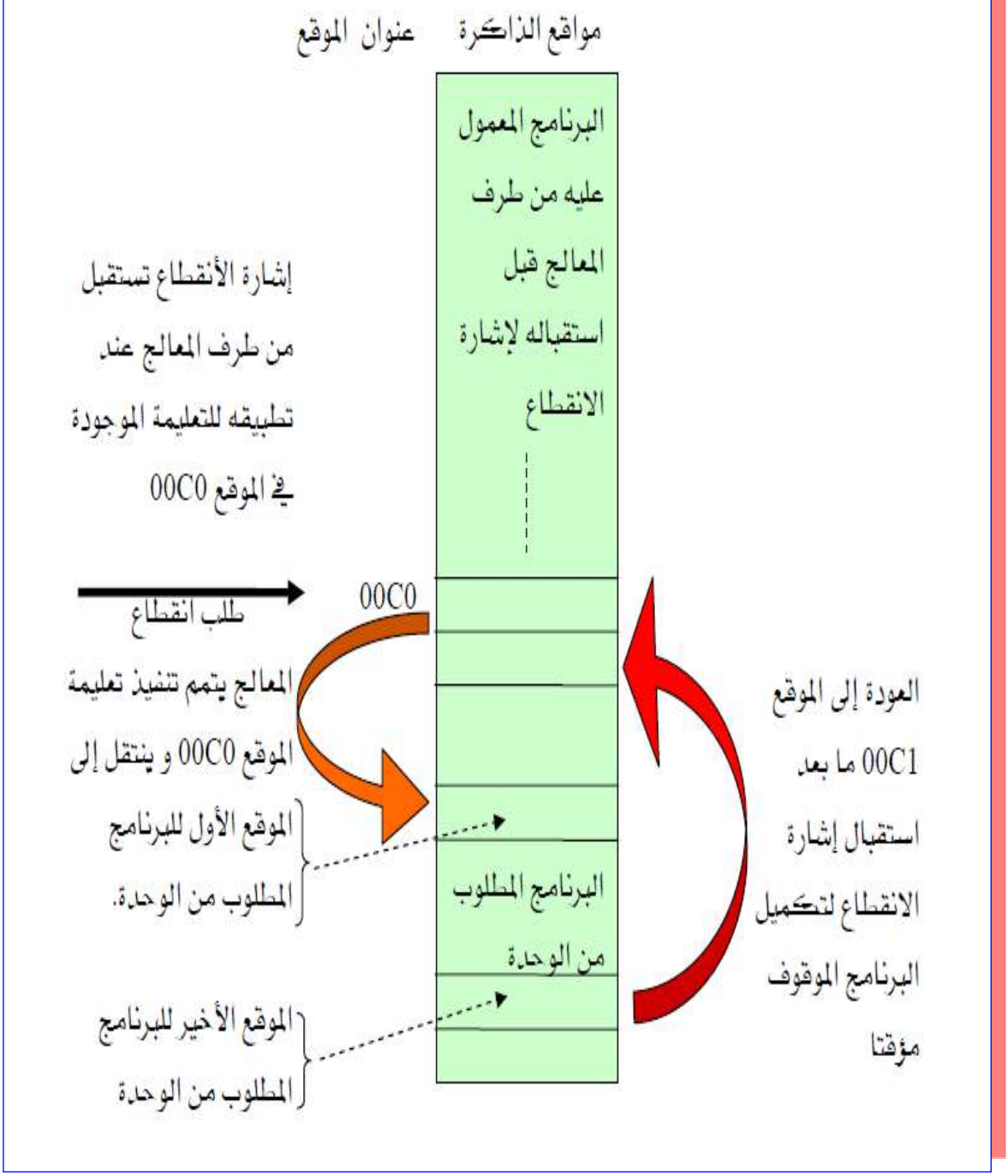
عند إعطاء إشارة المقاطعة لأي معالج من المعالجات يحدث التالي في الغالب مع بعض الاختلافات اليسيرة من معالج إلى معالج آخر:

1. الأمر الحالي يتم إكمال تنفيذه بواسطة المعالج.
2. عنوان الأمر الذي عليه الدور في التنفيذ (محتويات عداد البرنامج) تخزن في المكدس stack حتى يمكن العودة إليه عند الانتهاء من خدمة المقاطعة، كما يتم تخزين محتويات أي سجل يخشى من تغيير محتوياته في أثناء خدمة المقاطعة، ويتم ذلك عن طريق المبرمج.

3. كل إشارة مقاطعة لها عنوان خاص مصاحب لها، يتم وضع هذا العنوان في عداد البرنامج (عن طريق المعالج)، إذ يقفز المعالج إلى هذا العنوان، ويبدأ في تنفيذ البرنامج الذي يكون هذا هو عنوان أول أمر فيه، ويسمى هذا البرنامج ببرنامج خدمة المقاطعة، وتتم كتابته عن طريق المستعمل.
4. بعد الانتهاء من برنامج خدمة المقاطعة يعود المعالج إلى البرنامج الأصلي ليستأنف تنفيذه من مكان المقاطعة نفسه بالاستعانة بالعنوان الذي تم تخزينه في المكدس كما في الخطوة الثانية.

إنَّ الجهاز المقاطع في الكثير من الأحيان وبعد إعطاء إشارة المقاطعة وقبول المعالج لها والبدء في برنامج الخدمة فإنَّ الباب يظل مفتوحًا للأجهزة الأخرى للمقاطعة، مما قد ينشأ عنه موقف يجب الحرص منه. لو أنَّ المعالج بدأ في خدمة المقاطعة وما زال طرف المقاطعة الموصل على الجهاز المقاطع فعالًا فإنَّ المعالج سيبدأ في خدمة المقاطعة نفسها من جديد على الرغم من أنَّه لم ينته من الخدمة السابقة، وبمجرد أن يبدأ في خدمة المقاطعة من جديد للمرة الثانية وما زال خط المقاطعة فعالًا من قبل الجهاز المقاطع فإنَّ المعالج سيبدأ في الخدمة من جديد أيضًا، وهكذا فإنَّ المعالج سيدخل في حلقة لا نهائية لن يخرج منها من دون أن ينفذ برنامج خدمة المقاطعة، لتجنب هذا الموقف يجب على المبرمج أن يضع أمرًا معينًا في بداية برنامج خدمة المقاطعة يمنع المعالج من خدمة أي مقاطعة إلى أن ينتهي من الخدمة الحالية التي دخل فيها.

ويمكن مشاهدة المراحل المطبقة من طرف المعالج عند استعماله لطريقة المقاطعة كما في الشكل (3-18).



شكل 3-18 آلية عمل طريقة المقاطعة في نقل البيانات

وبعد بيان مفاهيم طريقة الاستجواب والمقاطعة يمكن تلخيص بعض الفروق بين الطريقتين بالآتي:

1. طريقة المقاطعة تحتاج إلى برمجيات فضلاً عن المكونات المادية حتى يتم معرفة عنوان الجهاز الذي يقوم بالمقاطعة, أمَّا في طريقة الاستجواب فلا نحتاج إلا البرمجيات فحسب.
2. طريقة الاستجواب تحتاج إلى برنامج حتى تتم معرفة إذا كان الجهاز مشغولًا أم لا، وهذا يحتاج إلى وقت طويل، ولكن في طريقة المقاطعة يكون ضياع وقت الحاسبة قليلاً جدًا، لأنه عندما يكون الجهاز جاهزًا يقاطع الحاسوب.
3. طريقة المقاطعة يجب أن يتم حفظ قيمة السجلات فيها قبل الذهاب إلى وحدة المعالجة المركزية في برنامج حزمة الجهاز بواسطة إيعاز Push – Pop، أمَّا في طريقة الاستجواب فلا حاجة إلى ذلك.

أسئلــة الفصل الثالث

س1/ عرف الآتي:

1) الطريقة التتابعية. 2) الطريقة المتوازية. 3) الطريقة التوافقية.

4) الطريقة شبه التوافقية. 5) طريقة الاستجواب. 6) طريقة المقاطعة.

س2/ ما أنواع النواقل في نظم الحواسيب؟ اذكرها واشرح باختصار عمل كل منها ؟

س3/ إرسم مخططًا يوضح طريقة اتصال المعالج الدقيق بالأجهزة المختلفة عبر استعمال النواقل ؟

س4/ حدد النواقل أحادية الاتجاه والنواقل ثنائية الاتجاه في الحالتين(ناقل البيانات, ناقل العناوين) ؟

س5/ إذا كان ناقل العناوين في حاسوب ما مكونًا من (16) خطًّا، فما الحجم الأقصى للذاكرة التي يستطيع الحاسوب الدخول إليها؟

س6/ ما الحاجة إلى خطوط التحكم $\overline{MEMR}$ و $\overline{MEMW}$؟

س7/ ما الأسباب الموجبة لاستعمال العوازل ما بين الدوائر الإلكترونية؟

س8/ وضح وظيفة شريحة 74374 المضافة إلى المعالج 8085 بالتفصيل.

س9/ كيف تحدث عملية النقل المتزامن للبيانات؟

س10/ وضح آلية النقل غير المتزامن للبيانات ؟

س11/ ارسم خوارزمية لبيان آلية عمل طريقة الاستجواب في نقل البيانات ؟

س12/ على ماذا تعتمد طريقة الاستجواب؟

س13/ عند إعطاء إشارة مقاطعة لأي معالج ما الذي يحدث؟

س14/ ما الفرق بين طريقة الاستجواب وطريقة المقاطعة؟

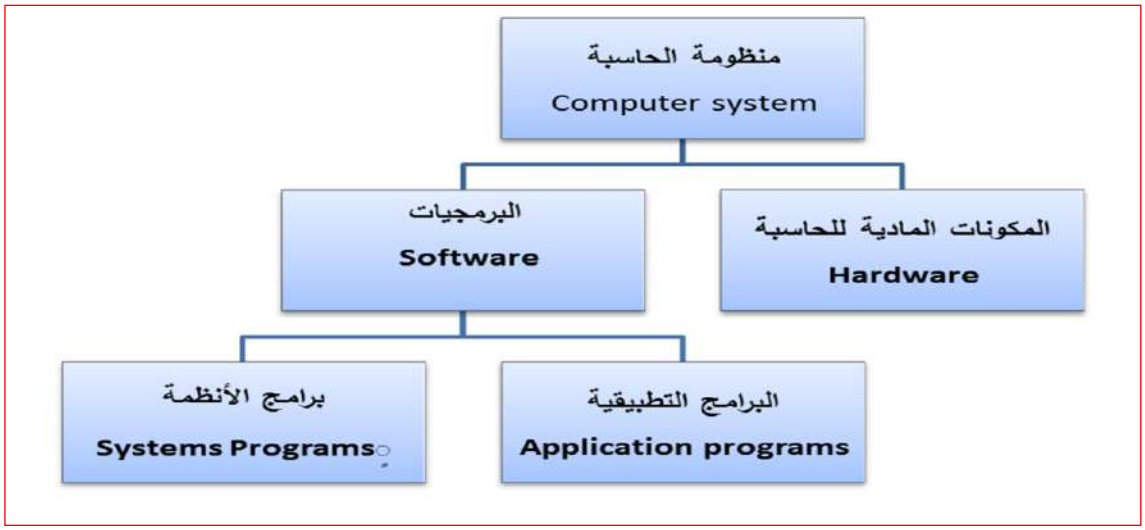
س15/ أين يتم إستعمال المقاطعة؟

الفصل الرابع

الأوامر

1-4 المقدمة

إن منظومة الحاسوب هي مجموعة من الأدوات التي تساعد المستعملين في انجاز مهمة معالجة البيانات وحل المسائل الرياضية, وهذه الأدوات تشمل المكونات المادية للحاسوب Hardware، والبرمجيات Software التي تشمل برامج الأنظمة Systems Programs كبرامج انظمة التشغيل Operating systems، برامج المترجمات Assemblers، تجعل هذه البرامج المستخدم يستفيد من امكانيات مصادر الحاسوب من دون التعمق في معمارية الحاسوب والغرض منها هو جعل نظام الحاسوب سهل الإستعمال لغير المختصين وتوظيف امكانية مصادر الحاسوب بكفاءة، والبرمجيات الأخرى كالبرامج التطبيقية Application programs، فهي برامج مكتوبة بواسطة المستخدمين لحل مسائل معينة، عرفت الأنظمة الرقمية بانها سلسلة من العمليات الدقيقة تنجزها الأنظمة على البيانات مخزونة في سجلاتها، والغرض الأساسي من الحاسوب الرقمي هو قدرته على تنفيذ شتى العمليات الدقيقة وموجهة بأي تسلسل معين من العمليات وجب على الحاسوب تنفيذها بواسطة البرنامج، يمكن ان يتم عمل البرنامج وتنفيذه بواسطة إستخدام لغة التجميع، يبين الشكل (4-1) مكونات منظومة الحاسوب.



الشكل 4-1 معمارية نظام الحاسوب

2-4 تصنيف الأوامر

الأوامر عبارة عن شفرات ثنائية Binary Code تطلب من المعالج الدقيق تنفيذ عملية معينة كجمع رقمين مثل الأمر ADD او خزن معلومة معينة في الذاكرة مثل الأمر STA أو تحميل معلومة معينة من الذاكرة مثل الأمر LDA.

ان الشكل العام للتعليمة (الأمر) في لغة التجميع الاسمبلي يمكن تمثيله بالصورة الآتية:

Label: Operation Operand(s) ; Comment

إن جميع حقول الإيعاز أو التعليمة هي حقول اختيارية ما عدا حقل التعليمة (Operation Field) والمقصود بالاختيارية ان لها حالات متعددة تختلف باختلاف نوع التعليمة أوالأمر، والحقول الاختيارية يمكن ان تتواجد في بعض التعليمات وتختفي في التعليمات الأخرى بحسب الحاجة اليها. يتم الفصل بين الحقول بواسطة مفتاح الـ TAB أو المسطرة (SPACE)، أي يكون هناك فراغ واحد في الأقل بين كل حقل والحقل التالي.

في ما يلي سنتحدث عن الحقول المختلفة بالتفصيل:

❖ **حقل العنوان أو العلامة Field Lable:**

يستخدم هذا الحقل في حالة حدوث عملية تفريع لهذا الأمر كإعطاء عنوان لأمر محدد أو إعطاء إسم لبرنامج فرعي كذلك لإعلان أسماء المتغيرات، يتم تحويل هذا الحقل إلى عناوين في الذاكرة وان اختيار اسم العلامة يجب ان يخضع لشروط تكوين الأسماء في لغة التجميع الأسمبلي، وهي كالآتي:

1. يتكون الاسم من الحروف A إلى Z.
2. تضمن الاسم الأرقام من 0 إلى 9.
3. يتضمن الاسم الرموز الخاصة مثل @، $، ويجب الا يكون ذلك الاسم من الرموز المحجوزة في لغة التجميع مثل أسماء السجلات المختلفة وغيرها.

يمكن أن يكون هذا الحقل بطول حتى 31 حرفاً، وغير مسموح وجود مسافات بداخل الحقل، كذلك لا يستعمل الحرف "." إلا في بداية الاسم، ولا يبدأ برقم، ولا يتم التفريق بين الحروف الكبيرة والصغيرة فيه، إليك بعض الأمثلة للتوضيح في الجدول الآتي:

أمثلة لأسماء مقبولة في لغة التجميع	أمثلة لأسماء غير مقبولة في لغة التجميع مع ذكر السبب	
isra-adil	isra adil	يحتوي علي فراغات
.@test	@.test	يحتوي علي الحرف (.) في منتصفه
$1000_done?	1000$_done	يبدأ برقم

❖ **حقل تعليمة الأمر (Operation Field):**

هو الإختصار الرمزي لاسم العملية OpCode، ويمكن لهذا الاختصار أن يتكون من حرفين حتى ثمانية أحرف، ويحتوي هذا الحقل علي شفرة الأمر المطلوب تنفيذها من قبل المعالج، ويجب أن تكون إحدى التعليمات المعروفة للبرنامج الذي سيقوم بمعالجتها هو المجمع Assembler، إذ سيقوم بتحويلها إلي لغة الآلة كمثال لذلك التعليمات ADD CMP و MOV وكلها تعليمات معرفة في برنامج المعالج. أما إذا كان إيعازاً Pseudo-Op (شبه التعليمة) فلا يتم تحويله للغة الآلة ولكنه يأمر المجمع Assembler بالقيام بشيء محدد، مثلاً Proc الذي يستعمل لتعريف برنامج فرعي Procedure.

❖ **حقل المعاملات (Operand Field):**

يحتوي هذا الحقل علي المعاملات من سجلات ومتغيرات وثوابت التي سيتم تنفيذ الأمر الحالي عليها (مثل عملية الجمع مثلاً)، ويمكن لهذا الحقل أن يحتوي على قيمتين أو قيمة واحدة أو لا يحتوي على أي قيمة على الإطلاق، وذلك حسب نوع الأمر المستعمل، والأمثلة التالية توضح ذلك.

الحقل Operand(s)	الأمر
لا توجد معاملات	HLT
يوجد معامل واحد وهو السجلC	INC C
يوجد معاملان وهما المتغير text والرقم 5	ADD text, 5

في حالة الحقول ذات المعاملين توضع فاصلة بينهما يكون المعامل الأول (القريب من التعليمة) هو الذي سيتم تخزين النتيجة فيه ويسمى بالهدف Destination Operand، وهو إما أحد السجلات، وإما موقع محدد في الذاكرة (لاحظ أن بعض الأوامر لا تقوم بتخزين النتيجة أصلاً) أما المعامل

الثاني فيحتوي علي المصدر Source Operand وعادة لا يتم تغيير قيمته بعد تنفيذ الأمر الحالي، ويكون الشكل العام لهذا النوع من التعليمات كالشكل السابق مع تغيير حقل المعاملات بالمعاملين الهدف والمصدر بهذه الصيغة:

Label: Operation Destination,Source ; Comment

أما بالنسبة إلى الإيعازات فيحتوي المعامل عادة على معلومات إضافية عن الإيعاز.

❖ **حقل التعليقات والملاحظات (Comment Field):**

يحتوي هذا الحقل على ملاحظات من المبرمج وتعليقات على الأمر الحالي وهو عادة ما يقوم بتوضيح وظيفة الأمر وأي معلومات إضافية قد تكون مفيدة لأي شخص قد يقرأ البرنامج وتساعده في فهمه، يتم بدء هذا الحقل بالفاصلة المنقوطة ";" وأي عبارة تقع بعد هذه الفاصلة المنقوطة يتم تجاهلها علي أنها ملاحظات، يستعمل عادة للوصف او التعليق على الأمر الحالي وهو يستعمل لتوثيق البرنامج. فهو حقل اختياري (أي يمكن ان نهمله) في جميع الأوامر، ولكن بعض الأوامر في لغة التجميع تحتاج الى ملاحظات لغرض التوضيح، لذا من الأفضل أن يتم وضع تعليقات على أي أمر غير واضح، أو يحتاج إلى تفسير وعادة ما يتم وضع تعليق علي كل سطر من أسطر البرنامج، ويتم إكتساب الخبرة بمرور الزمن عن كيفية وضع التعليق المناسب، فمثلاً التعليق التالي غير مناسب:

MOV C. 00 H : move 0 to C

وكان من الأفضل أن يتم كتابة التعليق الآتي:

MOV C, 00 H ; C counts terms, initialized to 0

هنالك نوعان رئيسان من التعليمات في لغة التجميع:

1- الأوامر أو التعليمات Instructions

إذ يقوم الـ Assembler بتحويلها إلى لغة الآلة، لاحظ عزيزي الطالب المثال الآتي:

Start: MOV C, 5 ;initialize counter

▼ ▼ ▼▼ ▼

الأمر عنوان الامر (وضع رقم 5 في C) حقل الملاحظات

(يوضح أن 5 هي القيمة الابتدائية للعداد)

2- موجهات المجمع Assembler-Directives:

وهي أيعازات إلى المجمع (Assembler) الذي يأمر بتنفيذ عملية التجميع للقيام ببعض العمليات المحددة، مثل تخصيص جزء محدد من الذاكرة لمتغير محدد وتوليد برنامج فرعي، ومثال على تلك الإيعازات:

Main Proc وهذا الايعاز يقوم بتعريف برنامج فرعي (إجراء باسم Main)

1-2-4 تصنيف الأوامر (التعليمات) في لغة التجميع على أساس النوع

تصنف التعليمات بنحوٍ رئيس الى أربعة أصناف وظيفية واسعة الإستخدام لعدة مجاميع من الأوامر وتشمل:

- **عمليات معالجة البيانات والعمليات التي تجري على الذاكرة:**

وتشمل أوامر نسخ البيانات وأوامر (تعليمات) التحميل والخزن حيث تحمل هذه الأوامر القيم وتنقل البيانات فوراً بين الذاكرة وسجلات عامة الأغراض مثل أوامر (MOV ، Push، POP) وأوامر سلسلة نصية String وهي خاصة للتعامل مع سلسلة نصية من أحجام مختلفة مثـل أوامـــر (النقل Moves، التحميل Loads، الخزن Stories)، وأوامر الإدخال والإخراج حيث تقوم هذه الأوامر بعمليات الإدخال والإخراج مثل أوامر (IN، OUT).

- **عمليات منطقية وحسابية (Arithmetic and Logic Operations):**

وتشمل الأوامر المنطقية والحسابية والإزاحة والدوران، فالأوامر الحسابية تقوم بالعمليات الحسابية للقيم في السجلات مثل (Add,Sub,Mul,Div, Dec,Cmp) والأوامر المنطقية تقوم بالعمليات المنطقية مثل (And، Or، Xor، Not) وأوامر الإزاحة والدوران مثل:

(Shl، Shr، Rcl، Rcr).

- **عمليات ضبط الإنسياب (Flow Control Operation):**

وتشمل أوامر التحكم أو التنقل في البرنامج، مثل أوامر القفز والتفرع التي تغير برنامج ضبط الإنسياب، ومن الأمثلة على هذه الأوامر (Jmp، Call، Ret).

- **عمليات تحكم المعالج (Processor Control):**

وتشمل أوامر التحكم بالمعالج أوامر خاصة بالمقاطعة أوالتوقف مثل (Interupt، Halt).

2-2-4 تصنيف الأوامر على أساس العنونة

يتم تصنيف الأوامر على أساس العنونة تبعاً لصيغة الأمر (Instruction Format)، وكل أمر له قسمان: أحدهما يشير إلى نوع العملية المطلوب تنفيذها في المعالج يسمى Opcode (شفرة العملية)، والآخر يشير إلى البيانات المطلوب إجراء العمليات عليها يسمى Operand (المعامل)، ويمكن للمعامل أن يتخذ أشكالاً متعددة هي:

1. قيمة فورية مباشرة أو ثابتة (Expressions، Constants) مثل 15، 20 ، 'a'.
2. أحد سجلات المعالج (Register) مثل A، B، D.
3. موقع محدد في الذاكرة يتم تحديده عن طريق العنوان (Memory address) مثل: [500]، [300].
4. منافذ الإدخال والإخراج I/O ports.

وكل أمر له حجم خاص به وهو يصنف إلى اربع مجاميع، وهي:

أ- أوامر بحجم بايت واحد (One-byte instructions) ويحوي العملية والمعامل في نفس البايت (أي المعامل من ضمن الإيعاز) مثل الأوامـــــر التي ليس لها معامـــــل كالأوامـــــر: (PUT، GET، BRK، HLT).

ب- (أوامر بحجم بايتين Tow-byte) اول byte تحدد للعملية، وثاني byte تحدد للمعامل، مثل أوامر لها معامل واحد فحسب (أوامر القفز بانواعها Instructions JUMP، NOT ، INC،MUL،DIV، LDA).

ج- (أوامر بحجم ثلاثة بايت Three Byte) أول byte يحدد للعملية، و 2byte التاليـــــان يحددان لعنـــوان المعامـــل بحجـــم 16-bit مثل أوامـــر لها معامليـــن: (MOV، ADD ، SUB، CMP،AND،OR).

د- (أوامر بحجم أربعة بايت Four Byte) أول byte يحدد للعملية، و3byte بعدها تحدد لعنوان المعامل بحجم 24-bit، مثل أوامـــر لهـــا ثلاثـــة معامـــلات (SHLD ,ADD).

وفي أدناه أمثلة لهذه الأوامر:

وصف العملية	المعاملOperand	شفرة العملية Opcode
أوقفْ المعالجَ	بدون معامل	HLT
أضف واحداً للسجل A	A (معامل واحد)	INC
اخزن 100 في السجل A	A,100 (معاملان)	MOV
أزح بمقدار اربع خانات أو بتات لليسار من سجل A إلى السجل D	D,A,4 (ثلاثة معاملات)	SHLD
أضف ثلاثة قيم مطلقة للمركم	$3,$2,$4 (ثلاثة معاملات)	ADD

3-4 العنونة Addressing

يقصد بالعنونة هو تنظيم الذاكرة وتسهيل ادارتها عن طريق تقسيمها الى أجزاء ووضع عناوين محددة لكل جزء بحيث يسهل تحميل البرامج في الذاكرة. وهناك عدة طرق للعنونة منها:

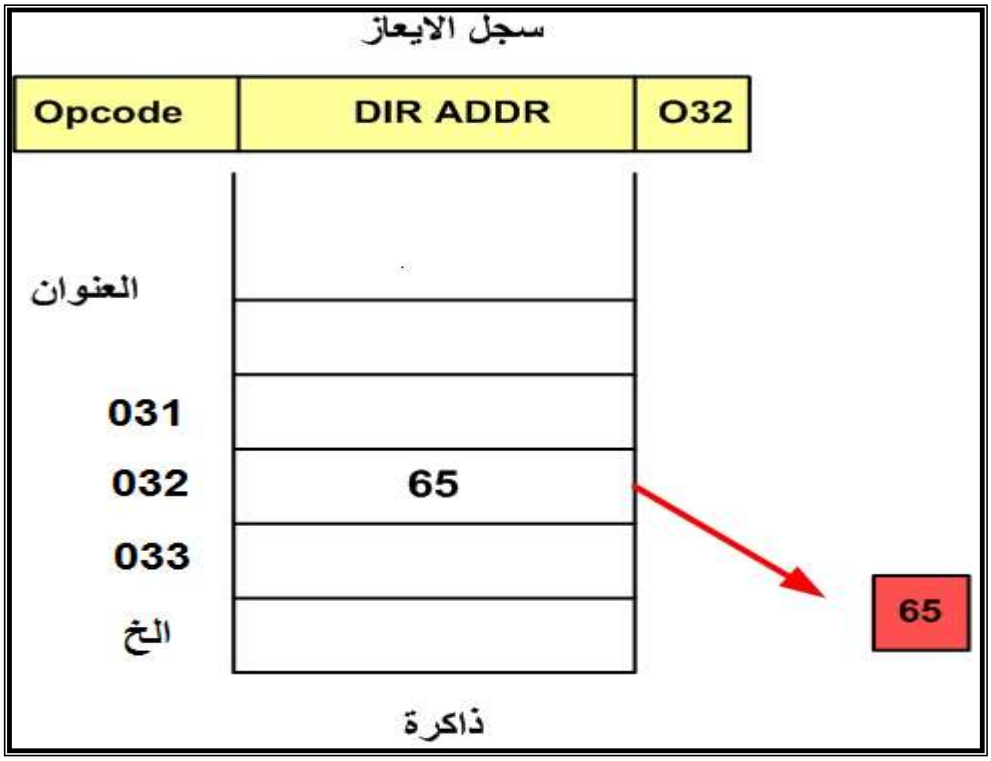
1- العنونة الفورية Immediate Addressing:

عند إعتماد هذه الطريقة فان التعليمة تحتوي على المعامل كجزء منها. أي إن حقل العنوان يحتوي على المعامل نفسه, ولكن من مساوئ هذه الطريقة إن قيمة المعامل ثابتة ولابد من الرجوع الى البرنامج في كل مرة عندما نرغب في تغيير قيمة المعامل. كما في الآمثلة التالية:

المعنى	تعليمات لغة التجميع
نقل نسخة المعامل 30 فوراً إلى مسجل A	Mov A, 30 H
نقل نسخة المعامل محرف الاسكي فوراً إلى مسجل A	Mov A,'a'
إضافة المعامل 5 فوراً إلى المركم	Add 5

2- العنونة المباشرة Direct Addressing

وهي أكثر طرائق العنونة انتشاراً، ويتم فيها الحصول على العنوان الفعلي من العنوان المبين في التعليمة مباشرة، أي إن حقل العنوان يحتوي على عنوان موقع الذاكرة الذي يخزن المعامل، ويبين الشكل (2-4) طريقة الحصول على المعامل في حالة العنونة المباشرة.



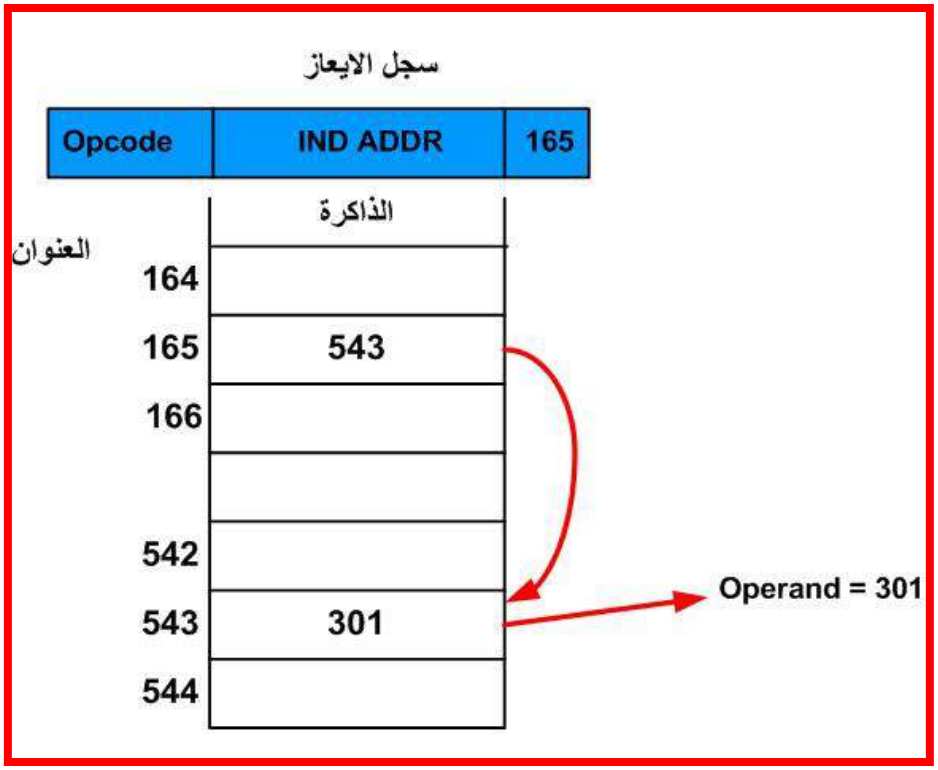
الشكل 4-2 آلية عمل العنونة المباشرة

ومن مساوئ العنونة المباشرة إن عدد مواقع الذاكرة المعنونة محدود، فإذا كان حقل العناوين يتكون من مواقع ثنائية عددها N فإنه يمكن عنونة N موقعاً فقط، ويمكن حل هذه المشكلة بعدة طرائق منها:

1- زيادة حجم حقل العناوين وبالتالي حجم التعليمة، الا ان هذا يتطلب زيادة حجم مواقع الذاكرة.
2- تخصيص اكثر من موقع ذاكرة لتخزين التعليمة، وتستعمل هذه الطريقة في الحواسيب المايكروية.

3- العنونة غير المباشرة Indirect Addressing:

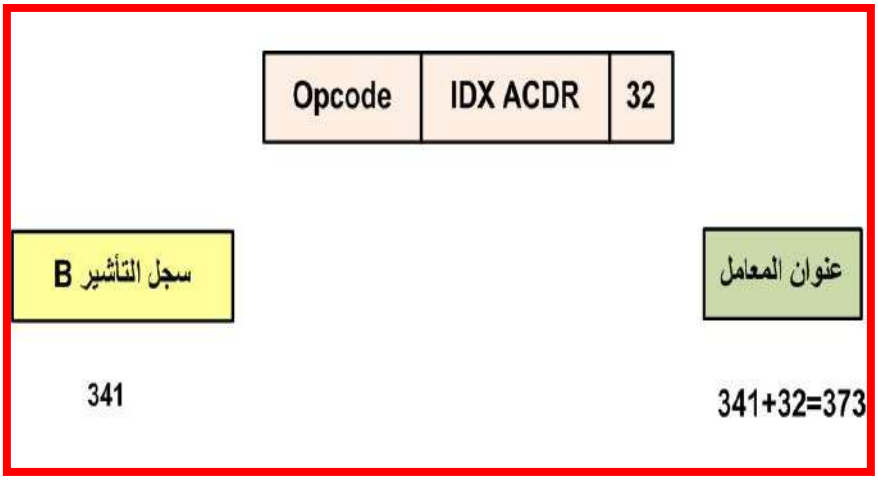
إن العنوان الفعلي في هذه الطريقة هو محتوى موقع الذاكرة المعنون بواسطة التعليمة. أي إن موقع الذاكرة المعنونة بواسطة العنوان المبين في التعليمة يحتوي على عنوان موقع الذاكرة الذي يخزن فيه المعامل، فللحصول على المعامل لا بد من الرجوع الى الذاكرة مرتين, ويبين الشكـــل (3-4) طريقة الحصول على المعامل في حالة العنونة غير المباشرة.



الشكل 4-3 آلية عمل العنونة غير المباشرة

4- العنونة المؤشرة Indexed Addressing:

بحسب طريقة العنونة المؤشرة، للحصول على العنوان الفعلي يجب جمع العنوان المبين في التعليمة مع محتوى سجل خاص يسمى بسجل التأشير Index Register، ويبين الشكل (4-4) طريقة الحصول على العنوان الفعلي.

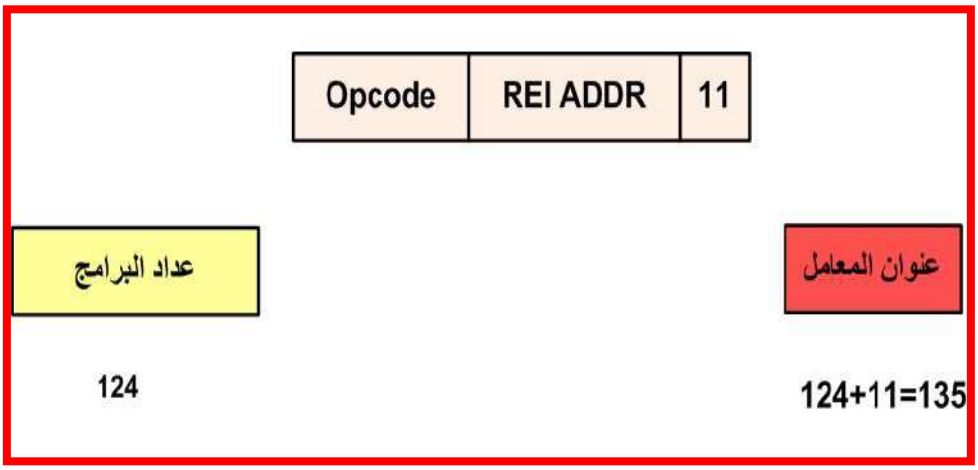


الشكل 4-4 آلية الحصول على العنوان الفعلي في العنونة المؤشرة

من مساوئ إعتماد العنونة المؤشرة ضرورة إجراء عملية الجمع للحصول على العنوان الفعلي وهذا يتطلب دائرة خاصة بالجمع, ولكن من مزايا هذه الطريقة بالمقارنة مع العنونة غير المباشرة، هي الرجوع الى الذاكرة مرة واحدة بدلاً من مرتين.

5- العنونة النسبية Relative Addressing:

هذه الطريقة تشابه طريقة العنونة المؤشرة من حيث طريقة العمل ولكن تختلف عنها باستخدم عداد البرنامج بدلاً من سجل التأشير للحصول على العنوان الفعلي، حيث يجب جمع العنوان المبين في التعليمة مع العنوان المخزون في عداد البرنامج, ويبين الشكل (4-5) طريقة الحصول على العنوان الفعلي في العنونة النسبية.



الشكل 4-5 آلية الحصول على العنوان الفعلي في العنونة النسبية

أسئلة الفصل الرابع

س1: ما أهم الفروقات بين حقل التعليمة الأمر وحقل الملاحظات؟

س2: إشرح باختصار عمليات معالجة البيانات والعمليات التي تجرى على الذاكرة؟

س3: ما أهم الفروقات بين العنونة المباشرة والعنونة الفورية؟

س4: إشرح العنونة المؤشرة مع الرسم ؟

س5: إشرح العنونة النسبية مع الرسم ؟

س6: إشرح العنونة غير المباشرة مع الرسم؟

الفصل الخامس

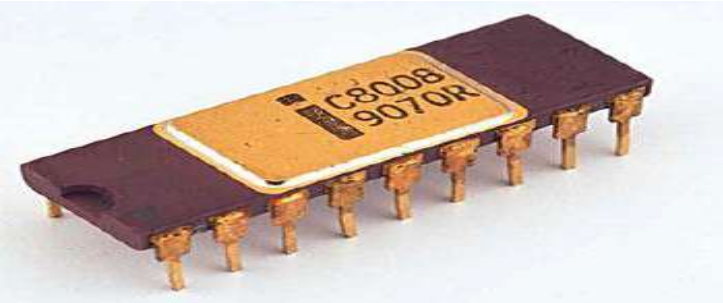
المعالجان الدقيقان 8080 و 8085

1-5 المقدمة

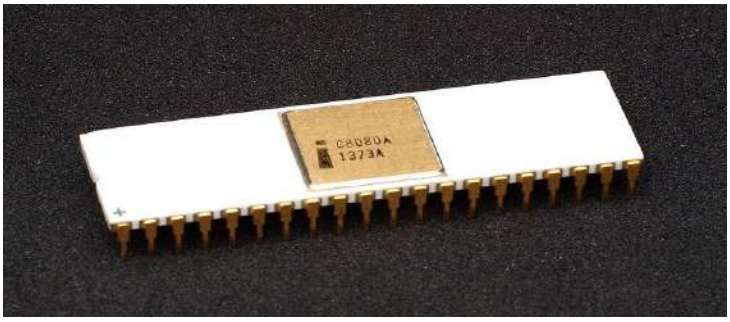
من أهم الأجيال القديمة للمعالج الدقيق هو المعالج 8080 والمعالج 8085، هذان النوعان من المعالجات كان لهما دور مميز في فهم عمل المعالجات، وأهم الإشارات الضرورية التي تسيطر على جميع فعاليات الأجزاء والوحدات الأخرى في جهاز الحاسوب الآلي، في هذا الفصل سيتم التعرف على المخطط الكتلي لكل نوع من هذين النوعين مع استعمالاتهما فضلاً عن تقديم أهم المواصفات الفنية التي تمتاز بها.

2-5 المعالج الدقيق 8080

هو ثاني معالج مصغر 8 بتات قامت شركة إنتل (intel) بتصميمه وتصنيعه وتم إطلاقه في نيسان من عام 1974م، وكان امتداداً ونسخة مدعمة للتصميم السابق 8008، الشكل رقم (5-1) يوضح معالج أنتل (intel) طراز 8008 اما الشكل الذي يليه رقم (5-2) فهو أيضاً معالج أنتل (intel) لكن من الطراز الذي يعد امتداداً لطراز 8008.



الشكل 5-1 المعالج الدقيق 8008



الشكل 5-2 المظهر الخارجي للمعالج الدقيق 8080

تم تصنيع طراز 8080 بإستعمال ترانزستور نوع NMOS ذي الحمل المعزز غير المشبع والذي يتطلب 12 V+ إضافياً ومزود 5 V-.

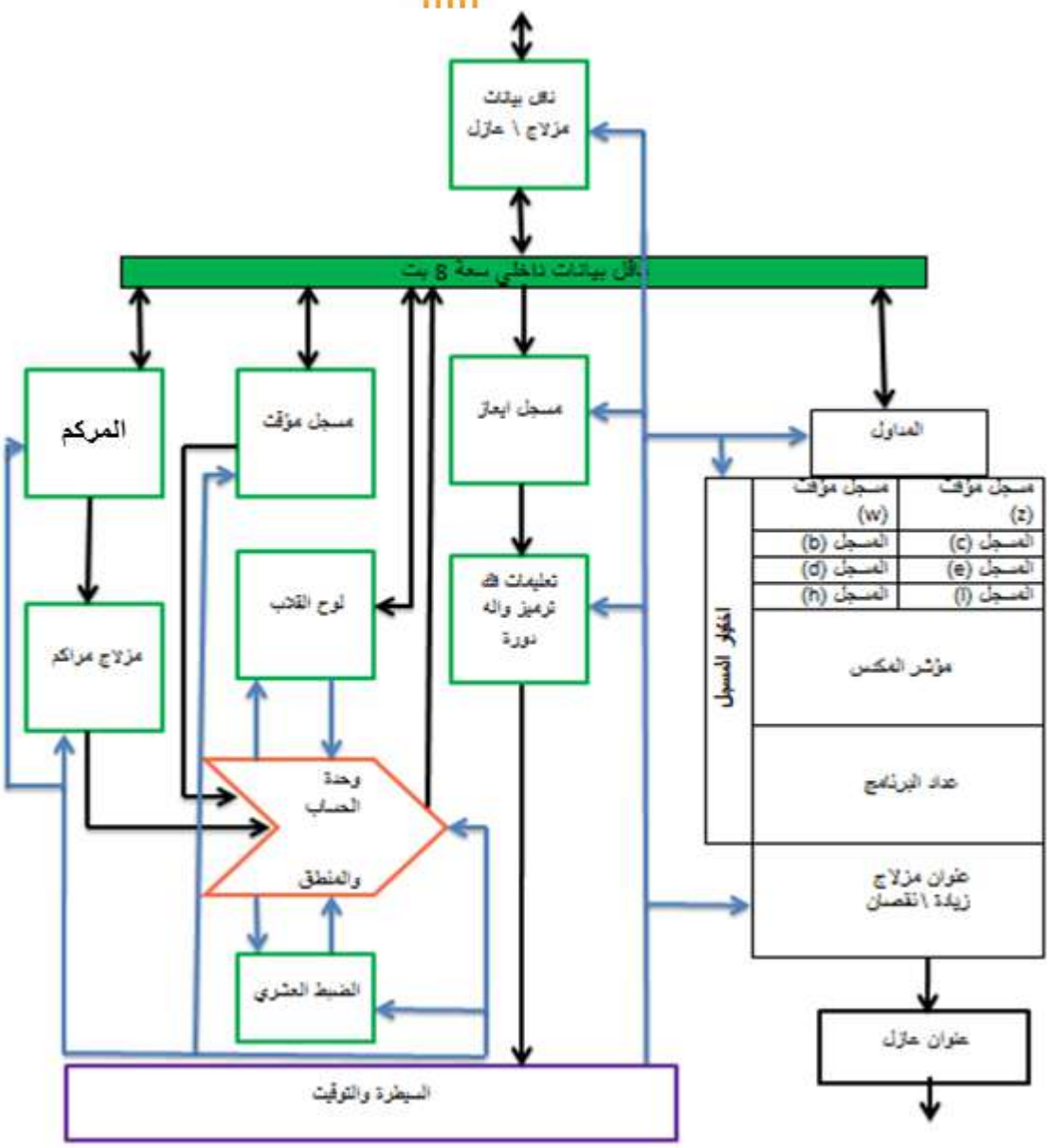
ويمكن تلخيص المواصفات الفنية للمعالج الدقيق 8080 بالنقاط الآتية:

1. عبارة عن معالج ذي 8 بتات.
2. أقصى تردد اشتغال يتراوح بين MHz (2 الى 4).
3. ذاكرة داخلية سعة 64 kB.
4. عبارة عن دائرة متكاملة ذات 40 طرفاً على شكل صفين.
5. فيه ناقل عنوان ذو 16 بتاً وناقل بيانات ذو 8 بتات.
6. يمتلك المعالج 7 سجلات ذات 8 بتات وهي (A,B,C,D,E,H & L).
7. السجل A هو مركم ذو 8 بتات اما بقية السجلات الستة فيمكن أن تستعمل كثلاثة سجلات على شكل أزواج ذات 16 بتاً (BC,DE,HL).
8. وفيه أيضاً سجل عنوان للذاكرة ذو 16 بتاً وعداد برنامج ذو 16 بتاً.

ولكل معالج له معمارية خاصة، إذ تتكون معمارية المعالج الدقيق 8080 من:

- سجل المصفوفة ومنطق العنوان.
- وحدة الحساب والمنطق.
- سجل الامر (الايعاز) وقسم السيطرة.
- ناقل بيانات عازل ثنائي الاتجاه، ثلاثي الحالة.

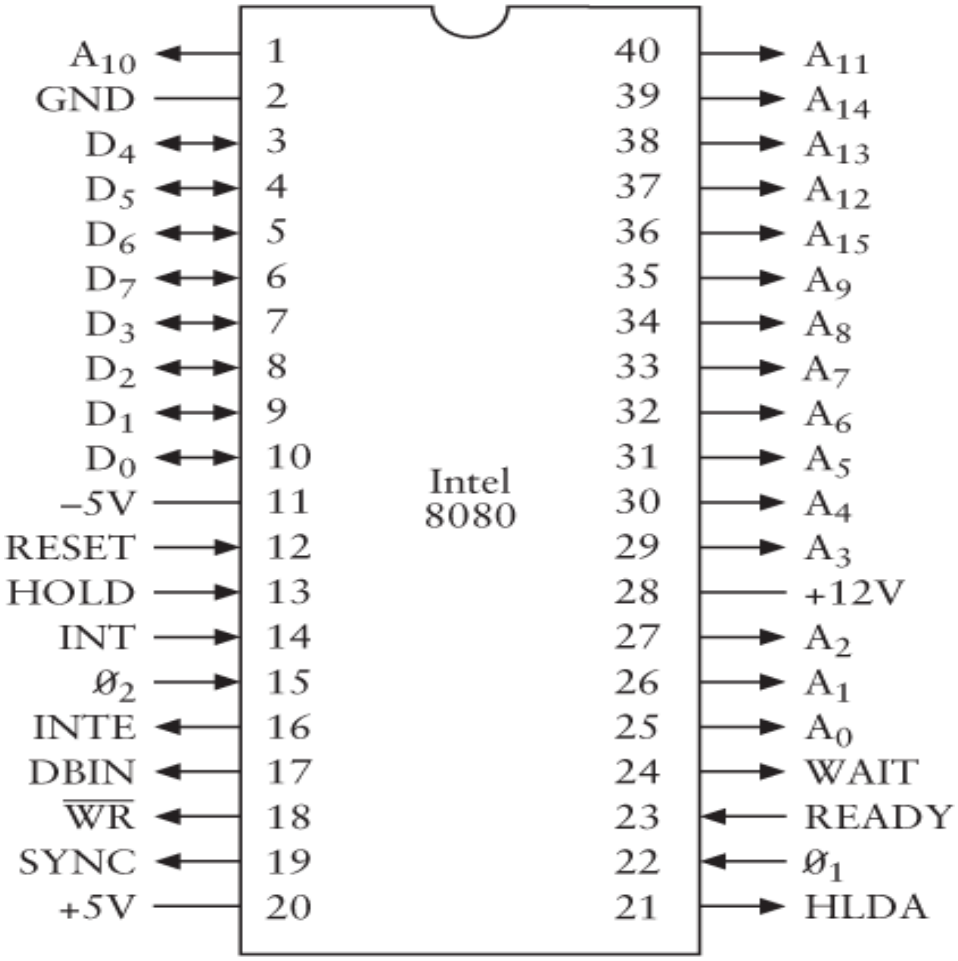
يوضح الشكل (3-5) المخطط الكتلي للمعالج الدقيق 8080 وهو عبارة عن وسيلة توضيحية تفسر كيفية عمل المعالج الدقيق.



الشكل 5-3 المخطط المعماري للمعالج الدقيق 8080 (للإطلاع)

1-2-5 الأطراف الخارجية للمعالج الدقيق 8080

يوضح الشكل (5-4) الأطراف الخارجية للمعالج الدقيق 8080 وإتجاه الإشارات.



الشكل 5-4 الأطراف الخارجية للمعالج الدقيق 8080 (للإطلاع)

1- يحتوي المعالج 8080 على ستة مخارج للتزامن والتحكم هي:

(INTE, HLDA, WR, WAIT, DBIN, SYNC)

2- يحتوي على أربعة مداخل للتحكم هي (RESET, INT, HOLD, READY)

3- يحتوي على أربعة أطراف للتغذية الكهربائية (-5, +5, +12, GND)

4- يوجد مدخلان لنبضات الساعة (Φ1, Φ2).

جدول رقم 5-1 الاشارات الموجودة في المعالج الدقيق 8080 ووصفها

اسم الإشارة	الوصف
خطوط العنوان من A0 الى A15	يتم تعيين خطوط العنوان الستة عشر من A0 إلى A15. خطوط عنوان ثلاثية الحالة، إما ان تكون ذات حالة منطقية عالية وإما منخفضة، وإما حالة المقاومة العالية، تستعمل حالة المقاومة العالية للسماح للأجهزة الأخرى الموصلة مع المعالج 8080 للسيطرة على عنوان DMA (الوصول المباشر إلى الذاكرة) عند إجراء العمليات المنطقية والحسابية.
خطوط البيانات من D0 الى D7	يتم تعيين خطوط البيانات الثمانية من D0 إلى D7. بيانات الخطوط هي خطوط ثنائية الاتجاه تستعمل لإدخال البيانات وإخراجها، وتستعمل أيضا لإظهار الناتج في أثناء الدورة الأولى من أي عملية يتم اجراؤها، خطوط البيانات تكون ثلاثية الحالة.
خط SYNC	اشارة SYNC تكون عالية في غضون الفترة الأولى من كل نبضة ثنائية لكل دورة، الإخراج SYNC تسمح بتزامن الحالة المنطقية مع البيانات الموجودة على خطوط البيانات.
خط DBIN	يشير خط DBIN إلى أن المعالج 8080 هو على استعداد لقراءة البيانات عبر خطوط البيانات، اما من الذاكرة أو من جهاز I / O. ويمكن استعمال DBIN لتمكين إدخال البيانات.
خط WR	يكون خط WR منخفضا عندما تكون البيانات الداخلة مستقرة، مشيراً إلى أن 8080 على استعداد لكتابة البيانات إلى الذاكرة أو جهاز I/O ويمكن استعمال WR لتمكين الكتابة.

خط READY	عند وضع خط READY في الحالة المنخفضة فانه يؤدي الى وضع 8080 في حالة الانتظار، ويقوم بإضافة نبضات بحسب الحاجة إلى تمديد دورة الزمن على النحو المطلوب في إجراء العمليات المنطقية الخارجية.
خط WAIT	تكون حالة خط WAIT مرتفعة في غضون طلب الانتظار التي تسببها حالة خط READY.
خط INT	يكون خط INT في مستوى عال، ليسمح بالمقاطعة الخارجية لـ 8080.
خط INTE	يصبح خط INTE عالياً إلى أن يتم إنهاء المقاطعات بواسطة 8080، ويصبح INTE منخفض عند انتهاء المقاطعة.
خط RESET	عند وضع مدخل RESET في مستوى منخفض لمدة لا تقل عن زمن ثلاث نبضات يؤدي الى إعادة تعيين عداد البرنامج وتصفير جميع المسجلات.
خط HOLD	عند وضع خط HOLD في حالة مستوى عال، يتسبب في توقف 8080 عن إجراء العمليات، في حالة استعمال HOLD فانه يتسبب في وضع ناقل العنوان وناقل البيانات في حالة المقاومة العالية. إن تحديد حالة النواقل ضرورية في حالة استعمال DMA.
خط HLDA	يصبح خط HLDA عالياً عند انتهاء HOLD، فيقوم بتحويل ناقل النظام الى حالة المقاومة العالية.

3-5 المعالج الدقيق 8085

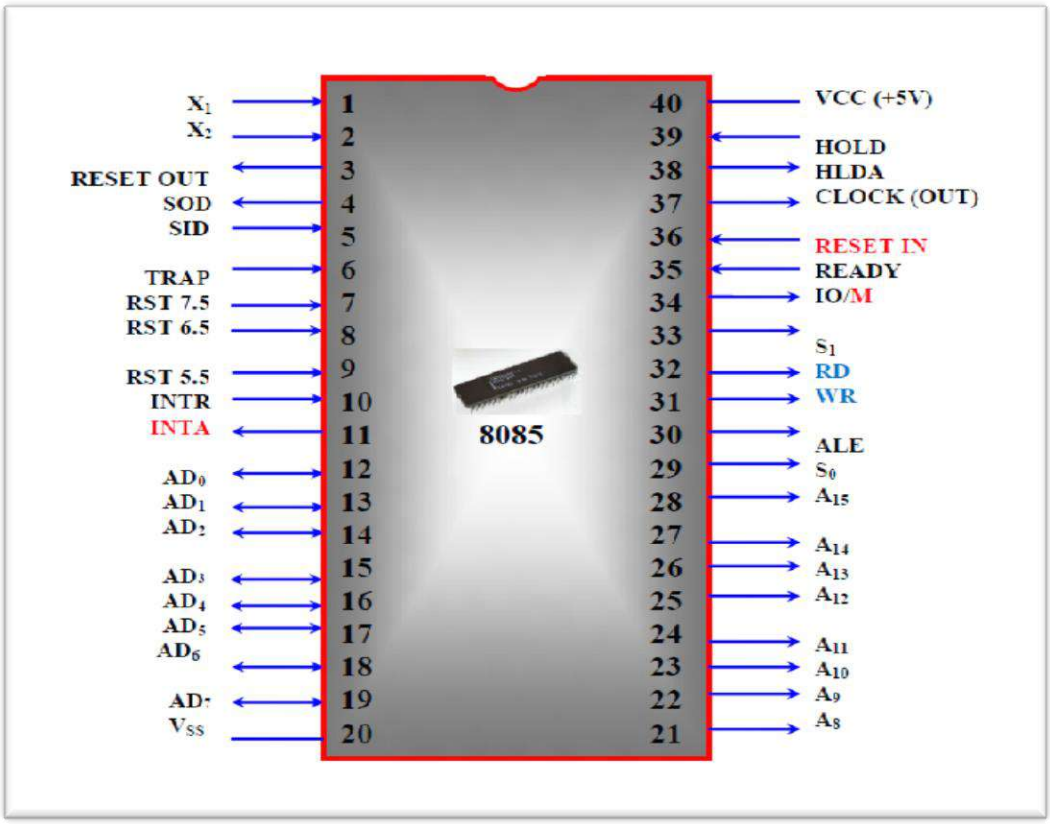
جاء المعالج الدقيق 8085 لتعزيز وحدة المعالج الدقيق 8080، حيث يستعمل المعالج 8085 أيعازات معالج 8080 نفسها, ويعمل على إمدادات طاقة بمقدار +5V. الشكل رقم (5-5) يوضح المظهر الخارجي للمعالج الدقيق 8085.



الشكل 5-5 المعالج الدقيق 8085

1-3-5 الاطراف الخارجية للمعالج الدقيق 8085

يوضح الشكل (5-6) مخطط الأطراف الخارجية للمعالج الدقيق 8085. ويمتاز المعالج 8085 عن المعالج 8080 بخاصية المزج الزمني Time Multiplexing إذ أن كل من ناقل العناوين وناقل البيانات يستعملان نفس الخطوط AD0-AD7 بحيث أن الإشارة الموجودة على هذه الخطوط تكون إشارة عناوين في بداية كل دورة أمر ثم تكون بعد ذلك إشارة بيانات. ويتم التعرف على نوع الإشارة على الخطوط الثمانية AD0-AD7 عن طريق الطرف Address (ALE) Latch Enable والذي يسمى بمنشط ماسك العنوان على الطرف 30 فعندما تكون قيمة هذا الخط 1 فان الإشارة على الخطوط تمثل عناوين, وعندما تكون قيمة الخط 0 فإنها تمثل بيانات.



شكل 5-6 الأطراف الخارجية للمعالج الدقيق 8085 (للإطلاع)

5-3-2 وظيفة أطراف المعالج 8085

1. X1,X2 (مدخل): يستعملان لتوصيل مولد نبضات توقيت خارجية، وذلك لتحديد تردد نبضات الساعة الداخلية أو توصيل للبلورة.
2. Reset Out (مخرج): تبين أنه حدث تصفير للمعالج.
3. SOD (مخرج): يدل على ان البيانات في حالة اخراج بصورة متسلسلة عبر جهاز خارجي.
4. SID (مدخل): يدل على ان البيانات في حالة دخول متسلسل من جهاز خارجي.
5. TRAP (مدخل): بداية المقاطعة ولا يمكن منعها.
6. RST 7.5 , RST 6.5 , RST 5.5 (مدخل): إعادة البدء بالمقاطعة.
7. INTR (مدخل): طلب مقاطعة ويستعمل كمقاطعة عمومية ويمكن السماح أو عدم السماح بالمقاطعة.

8. INTA (مخرج): الموافقة على المقاطعة وذلك لإدخال تعليمات اعادة بدء أو تعليمات استدعاء.

9. AD0---AD7 (مداخل ومخارج): خطوات نقل ثنائية الاستعمال، إذ تستعمل لنقل العناوين البيانات.

10. A15 --- A8 (مخارج): خطوط العناوين التي تحمل الإشارات الثماني ذات القيمة العليا، إذ تكون الإشارات الثمانية الأخرى على الخطوط AD7 --- AD0.

11. S0,S1 (مخرج): تمثل هذه المخارج إشارات تحكم تقوم بإخطار الوحدات الأخرى بنوع العمل الذي يقوم به المعالج الدقيق بحسب الجدول التالي:

S1	S0	الحالة
0	0	HALT
0	1	WRITE
1	0	READ
1	1	FETCH

12. ALE (مخرج): وهي إشارة ذات ثلاث حالات لبيان إشارة العنوان موجود على خطوط العناوين والبيانات ليتم خزنها.

13. WR (مخرج): إشارة كتابة تبين أن البيانات موجودة على خطوط البيانات، وسيتم كتابتها في مكان من الذاكرة أو جهاز الإدخال أو الإخراج.

14. RD (مخرج): إشارة قراءة تبين أن محتويات الذاكرة أو جهاز الإدخال او الإخراج سيتم قراءتها، وإن خطوط البيانات جاهزة لنقل البيانات.

15. IO / M (مخرج): يبين ما إذا كانت عملية القراءة أو الكتابة إلى الذاكرة أو إلى جهاز الإدخال او الإخراج.

16. READY (مدخل): جاهزة وهي إشارة تدخل على المعالج الدقيق لاخطاره بأن الوحدات الاخرى جاهزة لاستقبال او ارسال البيانات.

17. RESET IN (مدخل): وهي إشارة تقوم بإعادة ضبط المعالج (تصفير)، وتجعل عداد البرامج يساوي صفراً.
18. CLK (OUT) (مخرج): وهي مخرج لنبضات التوقيت (التزامن) لإشارات التحكم.
19. HLDA (مخرج): الموافقة على طلب الإمساك.
20. HOLD (مدخل): وهي إشارة تقوم بإخطار المعالج الدقيق بأن جهازاً آخر يريد إستعمال خطوط العناوين والبيانات.
21. Vcc: مصدر كهربائي ذو جهد 5V+.

Vss: طرف ارضي.

أسئلة الفصل الخامس

س1: إذكر مقدار الجهود (الفولتيات) التي يعمل بها المعالج الدقيق 8080 والسرعة التي يعمل بها؟

س2: إذكر خمسة خواص من الخواص الفنية للمعالج الدقيق 8080؟

س3: ما مكونات معمارية المعالج الدقيق 8080؟

س4: من دراستك للمعالج 8080، إذكر مدخلات ناقل التحكم ومخرجاته؟

س5: ما وظائف الأطراف التالية في المعالج 8080؟

DBIN-1 READY -2 RESET -3 INTE -4 INT -5

س6: من دراستك للمعالج 8085، ما الفرق بين طرف الإمساك HOLD، وطرف الانتظار WAIT؟

س7: ما المقصود بخاصية المزج الزمني في المعالج 8085؟

س8: ما وظيفة الأطراف التالية في المعالج الدقيق 8085؟

X1 , X2 -1 SID -2 INTA -3 ALE -4

س9: ما هي أطراف المقاطعة في المعالج 8085؟ عددها مع الشرح؟

س10: وضح بصورة جدول الإشارات الخاصة بالطرفين S0,S1 في المعالج الدقيق 8085؟

الفصل السادس

أجيال المعالج الدقيق وتكنولوجيا التطور

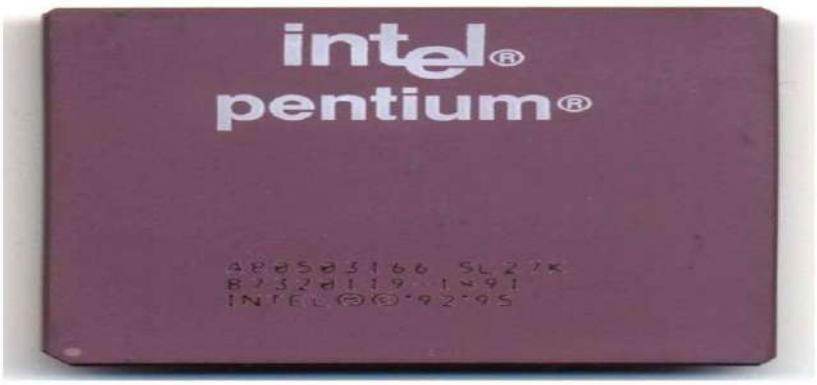
1-6 المقدمة

إن أجيال المعالج الدقيق عديدة ومختلفة من حيث المواصفات الفنية، إذ سيتم التعرف في هذا الفصل على العديد من الأنواع مثل المعالج بنتيوم والأجيال التي تلته, وما لحقه من تطورات في مجال التصميم وسنتعرف على أهم المواصفات الفنية التي يمتاز بها كل نوع من هذه الأنواع.

2-6 المعالج الدقيق بنتيوم I

في بداية التسعينيات، 1993م ظهرت سلسلة المعالجات بنتيوم Pentium، وأطلق عليها سلسلة لأنها خضعت لتطورات سريعة جداً ومتلاحقة, هذه المعالجات تعتمد طريقة القوائم المخفضة RISC لزيادة سرعة تنفيذ الأوامر فيه إلى 330 MIPS (Million Instruction Per Second) للمعالج الذي يعمل بنبضات ساعة مقدارها 200 MHz بدلا من 54 MIPS للمعالجات التي سبقته والتي تعمل بنبضات ساعة مقدارها 66 MHz، و من أهم المواصفات الفنية لهذا النوع من المعالجات هي:

1- اثنان من الأنسيابيات Pipelines، إحداهما: لتنفيذ الأوامر التي تتعامل مع البيانات الصحيحة Integer Pipelines، والأخرى: لتنفيذ الأوامر التي تتعامل مع البيانات الحقيقية Floating Point Instructions.
2- خاصية توقع أوامر التفريغ مثل القفز والنداءات على البرامج الفرعية، التي يكون لها دخل كبير في إسراع التعامل مع الذاكرة المخبأة.
3- ذاكرة مخبأة خاصة بالتعامل مع البيانات، وأخرى خاصة بالتعامل مع الأوامر.
4- مسار البيانات الخارجي يكون 64 bit.
5- حالة تشغيل جديدة وهي حالة توفير القدرة Power Saving Mode.



الشكل 6-1 المظهر الخارجي للمعالج الدقيق بنتيوم I

لا بد من الإشارة الى أن المدى العنواني للإدخال والإخراج I/O Address Space للمعالج بنتيوم يبلغ 64 kB للبوابات ذات 8 بتات، أو 32 kB للبوابات ذات 16 بتاً أو 16 kB للبوابات ذات 32 بتاً، إذ يمكن للمعالج التعامل مع كل هذه الأنواع أو مع بعضها.

الذاكرة المخبأة تكون غالية الثمن كما ذكرنا، لأنها تتميز بصغر زمن الاتصال بها، وتنقسم هذه الذاكرة الى عدة مستويات من حيث اتصالها بالمعالج. فهناك الذاكرة المخبأة ذات المستوى الأول Level 1 التي يرمز لها بالرمز L1، والذاكرة المخبأة ذات المستوى الثاني Level 2 التي يرمز لها بالرمز L2 والذاكرة المخبأة ذات المستوى الثالث ويرمز لها L3 والتي سيأتي شرحها لاحقاً بالتفصيل.

3-6 المعالج الدقيق بنتيوم II

أنتج هذا النوع من المعالجات الدقيقة في العام 1997م، وأتى بعدد من التغيرات عن المعالج السابق ومنها:

1. زادت الذاكرة المخبأة مستوى أول L1 إلى 32 kB ، وإستخدمت ذاكرة Cache مستوى ثاني L2 بسعة 512 kB.
2. سرعة الساعة الداخلية له وصلت إلى 450 MHz مع ناقل بسرعة 100 MHz.
3. تم وضع المعالج والمستوى الثاني من الكاش ومبرد الحرارة معا على لوحة واحدة موصلة بفتحة Slot على اللوحة الأم.
4. إن المعالج بنتيوم الثاني يعمل على 2.8 V.



الشكل 6-2 المظهر الخارجي لأحد أنواع المعالج الدقيق بنتيوم II

6-4 المعالج الدقيق بنتيوم III

في العام 1999م تم إنتاج المعالج الدقيق بنتيوم ثلاثة Pentium III بواسطة شركة Intel، بأسم Coppermine، ومن التكنولوجيا الجديدة في ذلك المعالج الدقيق:

1- بناؤه على عمليات دقيقة جدا بنحوٍ 0.18 مايكرون أي نحو 500/1 من سمك شعرة رأس الإنسان.

2- ووصل الى سرعة 1GHz وأكثر، إذ صمم بسرعات GHz (700, 733, 500, 850, 866).

3- تم زيادة عدد الحواجز الخزنية Buffers بين المعالج وناقل النظام الخاص به مما يؤدي إلى زيادة تدفق البيانات.

4- في العام 2000م تم إنتاج أنواع حديثة من هذا المعالج التي بلغت سرعتها 1.13 GHz فأكثر.



الشكل 6-3 المظهر الخارجي لأحد أنواع المعالج الدقيق بنتيوم III

5-6 المعالج الدقيق بنتيوم IIII

في نهاية عام 2000م إستحدث جيل جديد من المعالجات الدقيقة من عائلة بنتيوم وهو المعالج بنتيوم الرابع Pentium4، إذ يتصف هذا المعالج بأنه يعتمد على بنائية وهيكلية جديدة، فهو بداية لجيل جديد من المعالجات الذي يتميز:

1- المستوى الأول من الذاكرة المخبأة زادت لتصبح أكثر من 32 kB.

2- إضافة محرك تنفيذ سريع، فالشريحة تستعمل ثلاث ساعات تنفيذية منفصلة وهي: تتابعات الجزء المركزي، وتتابعات وحدة الحساب والمنطق، وأخيراً تتابعات النواقل.

3- هذا النوع بمستوى ثانٍ من الذاكرة المخبأة 512 kB التي يطلق عليها (المخبأة ذات النقل المتقدم) التي يبلغ معدل النقل بها 32 Byte في دورة الساعة الواحدة وسرعة نقل البيانات عن طريقها حوالى 44.8 kB في الثانية الواحدة.

4- تكنولوجيا الأمر الواحد والبيانات المتعددة SIMD2 التي زودت بنحو 76 أمراً جديداً التي تمسك بنحو 128 بتاً لوحدتي معالجة النقط العائمة (FPU) وحساب الأعداد الصحيحة التي تؤدي إلى سرعة معالجة الصور الفيديوية والوسائل المتعددة والرسومات ثلاثية الأبعاد.

5- ظهور أول ناقل للنظام بسرعة 400 MHz زيادة عن السابق الذي تبلغ سرعته 133MHz.

6- يبلغ عدد الترانزستورات المستعملة فيه نحو 42000000 ترانزستور وتبلغ ذاكرة المستوى الثالث نحو1 MB، يبين الشكل (6-4) المظهر الخارجي لأحد أنواع هذا المعالج.



الشكل 6-4 المظهر الخارجي لأحد أنواع المعالج الدقيق بنتيوم 4

6-6 تطور المعالجات

في ظل التطور والتقدم العلمي والتكنولوجي الهائل الذي يشهده العالم في مختلف الأصعدة شهدت الأجهزة الملحقة بالحاسوب وبرمجياته قفزة علمية سريعة مما تطلب إنتاج معالجات ذات سرعة وكفاءة عالية, وقد تم بذل جهود كبيرة في مجال الأبحاث والأموال من قبل الشركات الرائدة في صناعة المعالجات كشركة AMD و IBM و Intel في مجال البحث والتطوير لغرض إنتاج معالجات دقيقة ذات سرعة وكفاءة عالية.

إن من أهم العقبات التي واجهتها صناعة المعالجات الحديثة هي كثافة الترانزستورات, فكلما زادت كثافة الترانزستورات زادت الفولتية المستهلكة وبالتالي تزداد درجة الحرارة فتنخفض كفاءة المعالج وتزداد نسبة الأخطاء فيه حتى في حالة استخدام أنظمة تبريد حديثة تقوم بتقليل الحرارة بشكل نسبي. وقد اتفقت الشركات المنتجة للمعالجات الدقيقة على وجود عنصرين مهمين يتحكمان بأداء المعالج الدقيق وكفاءته وهما:

1. تصميم المعالج الدقيق Actual Processor Design: الذي يمثل مجموعة الميزات والتقنيات الجديدة التي يقدمها تصميم المعالج نفسه.
2. تقنية تصنيع المعالج الدقيق Processor Manufacturing Technology: حيث تعتبر تقنية التصنيع والإنتاج هي العنصر الأهم في تقرير الحدود التي يمكن أن نصل إليها في تصميم جيل معين من المعالجات.

لقد ظهرت طرق جديدة وتقنيات كثيرة تم استخدامها في تصميم وتصنيع المعالجات الدقيقة أهمها:-

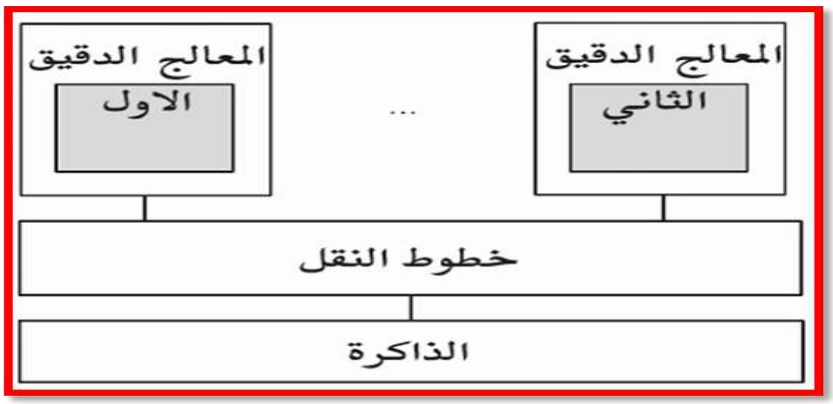
1- تقنية تعدد المعالجات Multiple processor Technology:

يقصد بتعدد المعالجات هو إستخدام أكثر من معالج دقيق ضمن منظومة الحاسوب الواحد ونظام تشغيلي قادراً على دعم أكثر من معالج دقيق وتوزيع المهام بيسر وإنتظام فيما بينهم, وتشترك المعالجات في هذه الأنظمة في الذاكرة الرئيسية ووحدات الإدخال والإخراج بالنسبة لمنظومة الحاسوب, لاحظ الشكل (6-5).



الشكل 6-5 تقنية تعدد المعالجات

وفي هذه التقنية يتم إستخدام نمط المعالجة المتوازية لعدة عمليات بمساعدة النظام التشغيلي بالنسبة للمعالجات الدقيقة, لاحظ الشكل (6-6).



الشكل 6-6 مخطط يبين عمل تقنية تعدد المعالجات

من أهم خصائص هذه التقنية هي:

1- انجاز الوظائف المتعددة في وقت قصير وبسرعة عالية.

2- استخدام نمط المعالجة المتوازية بالنسبة لمنظومة الحاسوب.

3- المرونة العالية في تنفيذ وانجاز المهام والوظائف بسبب التكرار في المعالجة.

4- ارتفاع نسبة الفولتية المستهلكة بسبب استخدام اكثر من معالج ضمن نفس المنظومة.

أما أهم المشاكل التي تواجهها هذه التقنية هي:-

1. الحاجة الى وجود أكثر من نظام تبريد بسبب استخدام اكثر من معالج ضمن منظومة الحاسوب الواحدة.
2. توقف منظومة الحاسوب عن العمل في حالة حصول خلل في احد المعالجات.

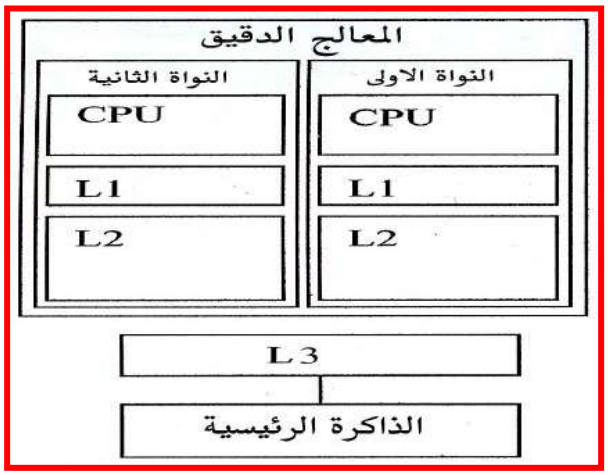
2- تقنية تعدد الأنوية Multiple Core Technology:

يقصد بتعدد الأنوية هو استخدام أكثر من معالج دقيق (نواة) ضمن شريحة معالج واحدة, والنواة هي المعالج نفسه بجميع مكوناته و وحداته و وظائفه و بالتالي عندما نذكر النواة نحن نقصد وحدة الحساب و المنطق و وحدة السيطرة المنطقية ومجموعة السجلات حيث أن هذه الوحدات الثلاثة وكما اسلفنا سابقاً مكملة لبعضها و تؤدي وظيفتين رئيسيتين أو طورين رئيسيين، الطور الأول هو جلب البيانات و الطور الثاني هو تنفيذ التعليمات، في الطور الأول يقوم المعالج بجلب التعليمة المراد تنفيذها و في الطور الثاني يقوم المعالج بتنفيذ التعليمة و تخزين ناتج العملية في أحد السجلات المتخصصة و عند الانتهاء من الطور الثاني يعود للطور الأول وهكذا. وببساطة فإن المعالجات متعددة الأنوية هي تلك التي تملك أكثر من معالج داخلها، على اعتبار أن المعالج يتكون من وحدة الحساب و المنطق و وحدة السيطرة ومجموعة سجلات و بالتالي فإن المعالجات ثنائية النواة مثلاً تملك وحدتان للحساب و المنطق و وحدتي سيطرة ومجموعتين من السجلات، و كذلك الحال للمعالجات التي تملك أكثر من نواتين.

إن المعالج الذي يملك نواتين قادر على تنفيذ دورتين في نفس الوقت أي أنه أسرع بالضعف من معالج أحادي النواة بنفس التردد، فمثلاً لو أخذنا معالج بتردد 1000 GHz أحادي النواة فإن الزمن المستغرق لتنفيذ دورة واحدة هو 0.001 ثانية و لكن خلال هذا الزمن سيتم تنفيذ دورة واحدة أما في المعالجات ثنائية النواة فإن زمن تنفيذ دورة واحدة لن يختلف و سيبقى 0.001 ثانية و لكن سيتم تنفيذ دورتين خلال هذا الزمن و هذا لا يعني أن كل دورة تحتاج لنصف الزمن للتنفيذ، بل أن كل دورة ستأخذ 0.001 ثانية و لكن بسبب وجود نواتين فإن كل نواة تقوم بتنفيذ دورة خلال الزمن و بالتالي في نفس الزمن نحصل على دورتين، و ينطبق الشيء نفسه على المعالجات التي تحتوي على أكثر من نواتين فالمعالج الذي يحتوي على 4 أنوية ينفذ 4 دورات خلال نفس الفترة.

إن الفائدة الحقيقية من المعالجات متعددة الأنوية تكمن في أنه يمكن تشغيل برنامجين أو أكثر و العمل عليهم بنفس الوقت دون تأثر أحدهما بالآخر لأن كل نواة ستنفذ برنامج، و لكن بالحقيقة إن للمعالجات متعددة الأنوية استخدامات أكثر فاعلية مثل تنفيذ البرامج بطريقة أسرع. ولا يقتصر وجود معالجات تعمل بتقنية تعدد الأنوية على الحاسوب فحسب بل توجد ايضاً في الهواتف الخلوية الحديثة وبعض التطبيقات الأخرى بسبب احتوائها على بعض البرمجيات والتطبيقات التي لا تنفذ بصورة متسلسلة بل مصممة لتنفذ تعليمتين أو أكثر في نفس الوقت. ومن الأمثلة على هذا النوع من المعالجات هو معالج CORE i3 و CORE i5 و CORE i7.

أما بالنسبة للذاكرة المخبأة بالمستوى الأول L1 والمستوى الثاني L2 فأنها تكون معزولة لكل نواة أي أن كل نواة لها ذاكرة مخبأة من المستوى الأول والمستوى الثاني خاصة بها, أما الذاكرة المخبأة في المستوى الثالث L3 فغالباً ما تكون مشتركة بين أنوية المعالج, لاحظ الشكل (6-7).



الشكل 6 -7 مخطط يبين تصميم المعالج الدقيق ثنائي النواة

من أهم خصائص هذه التقنية هي:

1- تمتاز بالتعقيد من حيث التصميم والتصنيع.

2- تستهلك فولتية عالية نسبياً تصل الى الضعف أحيانا بسبب طبيعة عملها.

3- كفاءة عالية وسرعة كبيرة في تنفيذ البرامج والتطبيقات.

4- نظراً لإرتفاع حرارة المعالج الدقيق بسبب إرتفاع الفولتية المستهلكة يفضل إستخدام نظام تبريد خاص ذو كفاءة عالية لتقليل الحرارة المتولدة ضمن الشريحة الواحدة.

3- تقنية الذاكرة المخبأة Cache Memory:

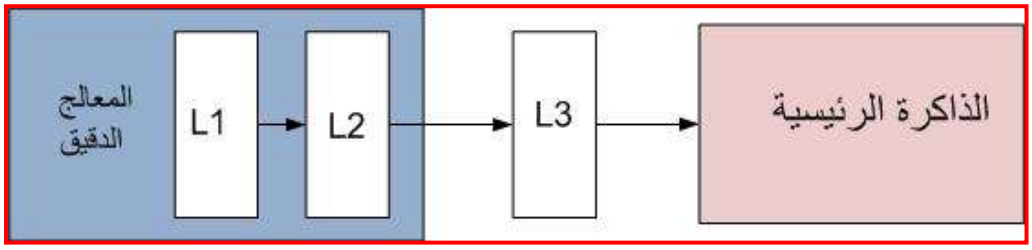
هي ذاكرة صغيرة الحجم وسريعة جداً توجد داخل المعالج الدقيق أو خارجه بين المعالج الدقيق والذاكرة الرئيسية والهدف الرئيسي من استخدامها هو خزن البيانات الاكثر طالباً من المعالج .

ولقد تم تطوير تصميم الذاكرة المخبأة, حيث توجد ثلاث مستويات لها، لاحظ الشكل (6-8):

أ- المستوى الأول Level 1:- وهو المستوى الذي يكون اقرب الى المعالج من بين المستويات ويرمز له بالرمز L1 ويكون داخل المعالج نفسه.

ب- المستوى الثاني Level 2:- وهو المستوى الذي يكون بعد المستوى الأول بين المستويات ويرمز له بالرمز L2 ويكون داخل المعالج نفسه.

ج- المستوى الثالث Level 3:- وهو المستوى الذي يكون بعد المستوى الاول والثاني من بين المستويات ويرمز له بالرمز L3 وغالباً ما يكون خارج المعالج الدقيق.



الشكل 6-8 مستويات الذاكرة المخبأة

أما آلية عمل الذاكرة المخبأة فيمكن إيجازها بما يأتي:

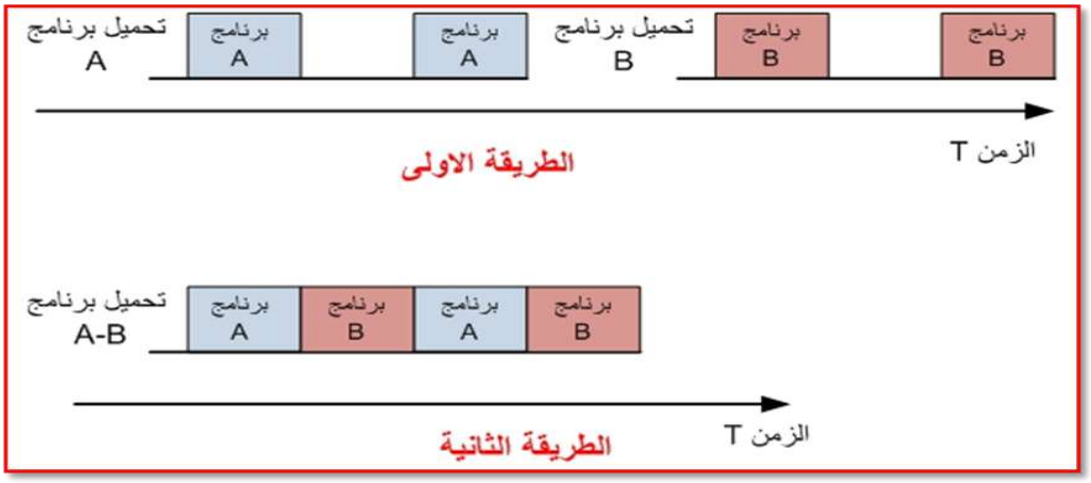
عندما يطلب المعالج بعض البيانات من إحدى اجزاء الحاسوب الأخرى مثلا من الذاكرة العشوائية (RAM) والتي هي أبطأ من المعالج بالطبع، فأن المعالج عليه الانتظار حتى تنتهي الذاكرة العشوائية من عملها وترسل إليه البيانات التي طلبها وعلى هذا الاساس فإن المعالج يهدر الكثير من الوقت في انتظار وحدة الذاكرة حتى تنتهي من عملها وترسل له البيانات المطلوبة. هنا يأتي دور الذاكرة المخبأة فعندما يحتاج المعالج إلى بيانات فإنه يقوم أولا بالبحث عنها داخل الذاكرة المخبأة فإذا وجدها فيأخذها من الذاكرة المخبأة وهذا أسرع بكثير طبعا، وتسمى هذه العملية

بإصابة الهدف وإذا لم يجد البيانات المطلوبة فتسمى فقدان الهدف وفي هذه الحالة يجب على المعالج طلب البيانات من الذاكرة الرئيسية والانتظار حتى يتم إرسال البيانات له.

4- تقنية تعدد البرامج والمشاركة الزمنية multiprogramming and time sharing:

يقصد بتعدد البرامج بأنها معالجة متوازية لبيانات ومعطيات عدة برامج تحصل في المعالج الدقيق نفسه خلال فترة زمنية واحدة, ولتنفيذ هذه التقنية تحتاج منظومة الحاسوب الى ذاكرة رئيسية ذات حجم كبير لان كل البرامج يتم تحميلها في الذاكرة الرئيسية تمهيداً لمعالجتها, ويتم تنفيذ العمليات بشكل مجدول حسب عمليات الإدخال والإخراج, ولا تحصل عملية مبادلة وانتقال من برنامج الى أخر لان كل البرامج موجودة في الذاكرة الرئيسية, ويفضل استخدام هذه التقنية في البرامج التي تكثر فيها عمليات الإدخال والإخراج كبرامج الحجز المصرفي, وهذه التقنية تعمل على زيادة سرعة معالجة وانجاز البرامج في المعالج الدقيق وبالتالي زيادة سرعة عمل منظومة الحاسوب.

لغرض توضيح هذه التقنية نفرض وجود برنامجين A و B يتم تنفيذهما بالطريقة الأولى بدون استخدام تقنية تعدد البرامج حيث يتم تحميل البرنامج الأول A من الذاكرة الرئيسية الى المعالج الدقيق ومن ثم معالجته وبعد ذلك يتم تحميل البرنامج الثاني B من الذاكرة الرئيسية الى المعالج الدقيق ومن ثم معالجته وهذا يستغرق وقت طويل, أما في الطريقة الثانية باستخدام تقنية تعدد البرمجة فسيتم تحميل كلا البرنامجين من الذاكرة الرئيسية الى المعالج الدقيق ليتم تنفيذهما سوياً كما في الشكل (9-6).



الشكل 6-9 إستخدام تقنية تعدد البرامج

أما بالنسبة للمشاركة الزمنية فيقصد بها إشتراك أكثر من جهاز في إستخدام بيانات الحاسوب وبرمجياته بشكل متزامن عن طريق تقسيم وقت المعالج الدقيق بين الاجهزة. ومن أهم مميزات المشاركة الزمنية إنه لا يوجد حاجة الى ذاكرة ذات حجم كبير لأن البرامج يتم تحميلها بشكل متعاقب الى الذاكرة الرئيسية حيث تبقى البرامج الأخرى موجودة في الذاكرة الثانوية لحين إستدعائها وإنتقالها الى الذاكرة الرئيسية عند حصول عملية مبادلة وانتقال من برنامج لآخر, ويفضل إستخدام المشاركة الزمنية في البرامج التي تكثر فيها العمليات الحسابية والجبرية كما هو الحال في برامج البطاقات المصرفية وشبكات الانترنت التي تسمح لآلاف المستخدمين للوصول الى نفس البرنامج الموجود على أجهزة الحاسوب المركزية في نفس الوقت حيث تتجدول عمليات التنفيذ حسب الفترة الزمنية وبشكل متزامن.

أسئلة الفصل السادس

س1: إذكر أهم المواصفات الفنية التي يمتاز بها المعالج الدقيق بنتيوم I ؟

س2: إذكر أهم المواصفات الفنية التي يمتاز بها المعالج الدقيق بنتيوم II ؟

س3: إذكر أهم المواصفات الفنية التي يمتاز بها المعالج الدقيق بنتيوم III ؟

س4: إذكر أهم المواصفات الفنية التي يمتاز بها المعالج الدقيق بنتيوم IIII ؟

س5: عرف كل مما يأتي:

1- تعدد المعالجات. 2- تعدد الأنوية. 3- تعدد البرامج. 4- المشاركة الزمنية.

س6: ما هي العناصر التي تتحكم بأداء المعالج الدقيق وكفاءته؟

س7: ما هي خصائص تقنية تعدد المعالجات معززاً إجابتك بمخطط بسيط يبين عمل هذه التقنية؟

س8: ما هو الفرق بين معالج يمتلك نواتين ومعالج يمتلك نواة واحدة؟ بين ذلك؟

س9: إرسم مخطط يبين تصميم المعالج الدقيق ثنائي النواة؟

س10: عدد مستويات الذاكرة المخبأة؟ ثم إرسم مخطط يبين هذه المستويات؟

س11: إشرح آلية عمل الذاكرة المخبأة في منظومة الحاسوب؟

س12: إشرح آلية تنفيذ برنامجين بإستخدام تقنية تعدد البرامج؟ معززاً إجابتك بالرسم؟

س13: ما هو الفرق بين تقنية تعدد البرامج والمشاركة الزمنية؟