

البرمجة التفرعية

Parallel Programming

References

- Peter S. Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann Publishers is an imprint of Elsevier 2011

الحواسيب التفرعية

- تصنيف الحاسبات المتوازية:

○ التصنيف الأكثر انتشاراً هو ذلك التصنيف الذي اقترحه "ميشيل فلاين" (Flynn) في عام 1966

- يأخذ تصنيف فلاين بعين الاعتبار عاملين اثنين وهما:

- كمية السريان (أو التدفق) للتعليمات
- و كمية السريان للبيانات التي تتدفق للمعالج.

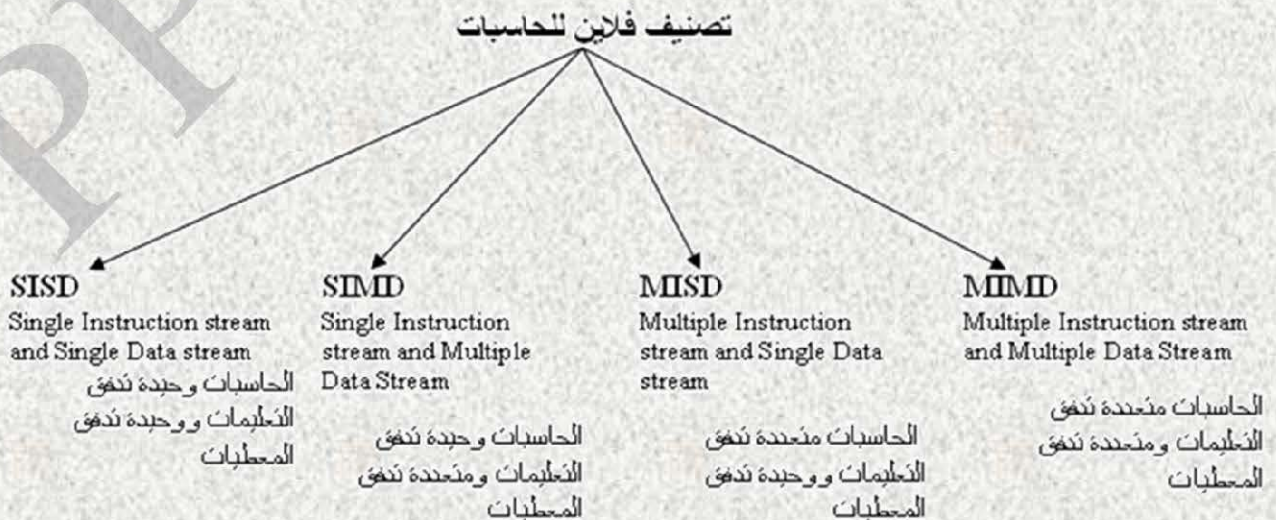
○ تصنيف فلاين Flynn's Classification Scheme:

- يعتمد أساساً على كمية تدفق البيانات والتعليمات الموجودة في الآلة.

- يقصد بالتدفق هنا على أنه تتابع أو تسلسل لعناصر (التعليمات أو المعطيات) كما نفذت بواسطة المعالج.

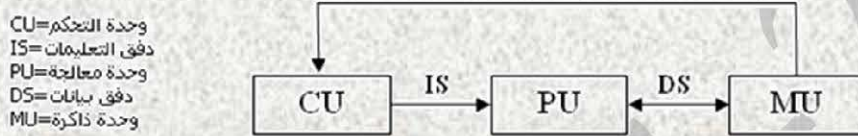
- مثال، تقوم بعض الآلات بتنفيذ تدفق واحد من التعليمات، بينما يتم تنفيذ عدة تدفقات في آلات أخرى. وبنفس الطريقة فبعض الآلات ترجع تدفقاً واحداً من المعطيات، وآلات أخرى ترجع تدفقات متعددة.

- وعليه فإن فلاين يضع الآلة في تصنيف واحد من بين أربعة وذلك اعتماداً على وجود تدفق واحد أو تدفقات متعددة.



■ الحاسبات وحيدة تدفق التعليمات ووحيدة تدفق المعطيات SISD:

- يندرج تحت هذا الصنف جميع الحاسبات التسلسلية المعتادة، مثل Apple Macintosh و DEC VAX
- تقوم بجلب التعليمات من الذاكرة ثم تقوم بتنفيذها باستخدام قيم المعطيات المشار إليها من الذاكرة وذلك بشكل تسلسلي
- يعرف بتصميم "فون نيومان" Von Neumann
- في البداية تم اعتماد هذا النوع من الحاسبات وعليه فالكثير من لغات البرمجة والمترجمات، ونظم التشغيل، وطرق البرمجة، كلها تعتمد أساسًا على هذا التصنيف
- تصنف كل أنواع الحاسبات التسلسلية بالإضافة إلى بعض الحاسبات التفرعية التي تعتمد على المعالجات الشعاعية مثل: Cray-1 على الرغم من أن تعليمات التشغيل تعمل على قيم بيانات شعاعية إلا أن لديها دَفَقٌ واحد من التعليمات



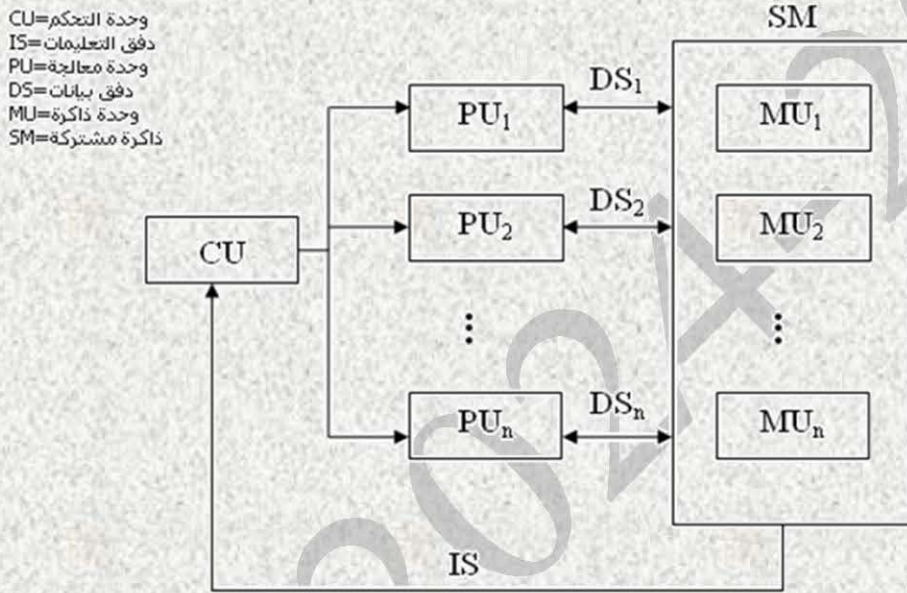
■ الحاسبات وحيدة تدفق التعليمات ومتعددة تدفق المعطيات SIMD:

- تعتمد على وحدة تعليمات واحدة تصدر أوامر إلى عدة وحدات معالجة وعليه نحتاج إلى تدفق متعدد للمعطيات
- عادة تصدر وحدة التعليمات نفس الأمر إلى جميع وحدات المعالجة .
- مثال: جميع وحدات المعالجة تنفذ تعليمة الجمع ADD ثم يتم تنفيذ تعليمة التخزين STORE وهكذا
- يتميز هذا النوع من الحاسبات بوجود وحدة تحكم مركزية .
- تنفذ وحدات المعالجة المختلفة تعليمة واحدة من وحدة التحكم وتقوم بتنفيذ هذه التعليمة بشكل متزامن على معاملات مختلفة.
- تكون هذه الحاسبات متزامنة، وغالبًا ما تملك ذاكرة مشتركة بين الوحدات.
- لتسهيل عملية الولوج المتوازي إلى الذاكرة
- يُلجأ إلى تقسيمها إلى بنوك مما يسمح باستخلاص عدة معاملات في نفس الوقت
- يتم الاتصال بين وحدات المعالجة المختلفة وبنوك الذاكرة عبر شبكة الربط.

- ونظراً لتنفيذ نفس العملية على الوحدات المختلفة فمن الممكن اعتبار الحاسبات SIMD كحاسوب وحيد المعالج الذي يقوم بتنفيذ التعليمات على أجزاء مختلفة من المعطيات

- يلائم هذا النوع العمليات على الأشعة وعلى المصفوفات وغالباً ما يستخدم من أجل عمليات الحساب العلمي حيث تؤخذ المعطيات من الذاكرة وينفذ عليها أمر واحد
- مثال: عملية ضرب العدد B بالشعاع (مصفوفة) A(I) والنتيجة يوضع في C(I)
- تحصل عملية ضرب العدد بجميع عناصر الشعاع بعملية واحدة على كل المعاملات

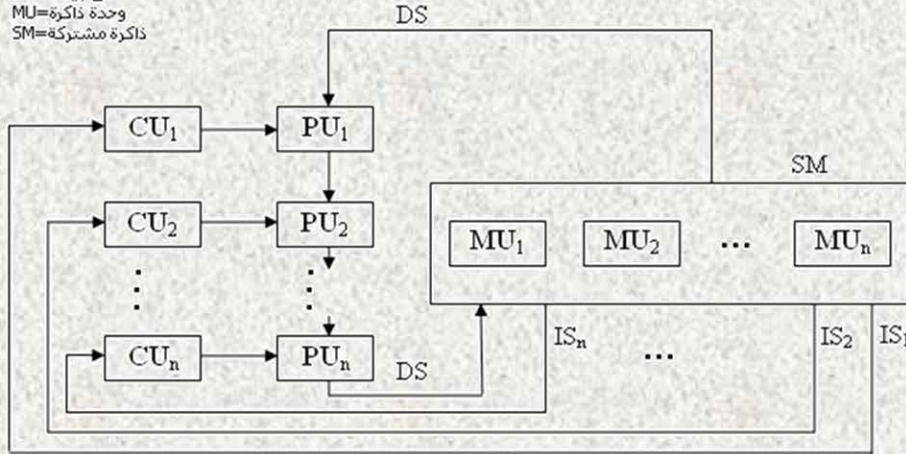
- تستخدم في هذه الحاسبات آلاف المعالجات، وتكون عادة متوسطة الأداء أو بسيطة جداً، وجودتها وأدائها العالي ناتج عن العدد الكبير للمعالجات المستخدمة



■ الحاسبات متعددة تدفق التعليمات ووحيدة تدفق المعطيات MISD:

- يتم تنفيذ عدة تعليمات مختلفة على معامل واحد خلال الدورة الزمنية للحاسب.
- يتميز هذا النوع بوجود عدد من المعالجات التي تعمل بشكل مستقل عن بعضها البعض.
- يتضمن كل معالج وحدة تحكم خاصة به تساعد على تنفيذ المهام الجزئية الموكلة إليه.
- يوجد القليل من الحاسبات المتوازية من هذا نوع حيث تعتبر قليلة الاستعمال فهي تعتمد مبدأ العمل التسلسلي

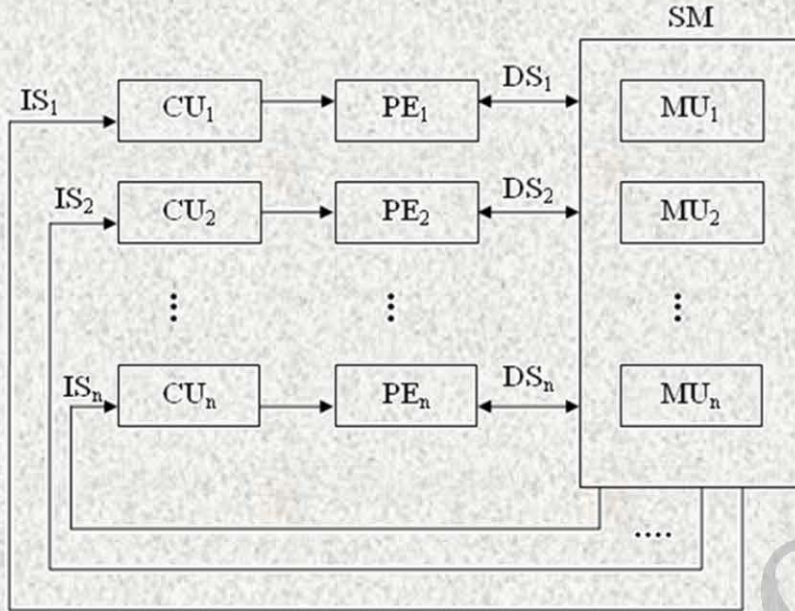
CU=وحدة التحكم
IS=دفع التعليمات
PU=وحدة معالجة
DS=دفع بيانات
MU=وحدة ذاكرة
SM=ذاكرة مشتركة



■ الحاسبات متعددة تدفق التعليمات ومتعددة تدفق المعطيات MIMD:

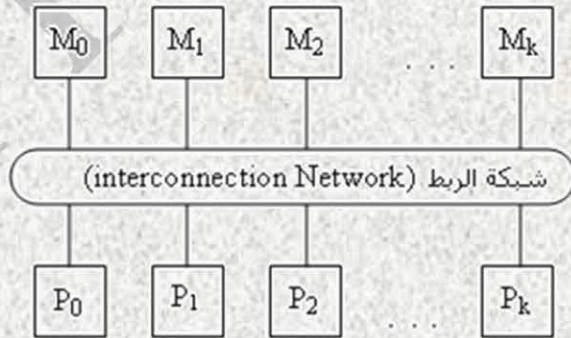
- يتم تنفيذ عدة تعليمات على معاملات مختلفة خلال الدورة الزمنية للحاسب .
- يتميز بوجود عدد من المعالجات التي تعمل بشكل مستقل عن بعضها البعض.
- يتضمن كل معالج وحدة تحكم خاصة به تساعد على تنفيذ المهام الجزئية الموكلة إليه.
- هذا النوع غير متزامنة وبالتالي فالحوادث التي تجري على معالج ما لا ترتبط بالحوادث التي تجري على المعالجات الأخرى .
- يمكن فرض نوع من التزامن بين هذه المعالجات عند الضرورات البرمجية لذلك
 - باستخدام بعض التعليمات الأولية المخصصة للترزامن
 - أو عن طريق العداد
 - ويمكن أن يتم أيضاً عن طريق البرمجيات ونظم التشغيل
- أن كل معامل يعالج من قبل معالج ما بأمر ما، وفي الوقت ذاته تعالج جميع المعاملات حسب كافة الأوامر ونتيجة هذه المعالجة تخزن في الذاكرة.
- عند المعالجة لا ينتظر أي معالج نتيجة من معالج آخر لأن جميع المعالجات تعمل بشكل غير متزامن وعلى التوازي
- تعتمد المعالجة التفرعية على مستوى الخوارزميات حيث يقسم البرنامج إلى مهمات جزئية مستقلة جزئياً فيما بينها، وتنفذ كل مهمة على معالج من معالجات الحاسب
- الأكثر استخداماً في الوقت الحالي وتعتمد عليها الحاسبات الحديثة
- يوجد لهذا التصنيف فئتين فرعيتين هامتين وهما:
 - الذاكرة المشتركة (Shared memory).
 - تمرير الرسائل (message passing).

وحدة التحكم CU
دفع التعليمات IS
عنصر معالجة PE
دفع بيانات DS
وحدة ذاكرة MU
ذاكرة مشتركة SM



• الذاكرة المشتركة MIMD Shared Memory:

- أي معالج يتضمن وحدة تعليمات و وحدة حسابية تمكنه من أن يقرأ من أو يكتب في ذاكرة مشتركة.
- ترتبط المعالجات مع وحدات الذاكرة بواسطة شبكة الربط، والتي يمكن أن تتخذ عدة أشكال تبعاً لنوع الآلة.
- شبكة الربط يمكن أن تثبت على شكل حلقة أو شبكة Mesh
- يمكن أن يحصل تعارض أو تضارب في الذاكرة عندما يحاول معالجان الكتابة في نفس الجزء من الذاكرة في نفس الوقت،
- وأيضاً ربما تتداخل المعالجات مع بعضها عند الكتابة في نفس خلية الذاكرة المشتركة مما يتسبب في إيقاف وفشل العملية الحسابية.
- كي لا يصبح التداخل مشكلة يجب أن تزود الآلة بأقفال أو أي آلية للترامن وذلك لضمان وجود معالج واحد فقط يتعامل مع خلية الذاكرة المشتركة في الوقت الواحد



- لكل آلات الذاكرة المشتركة MIMD سمات مشتركة.
- يزود نموذج الذاكرة المشتركة مخزن (Buffer) للمبرمج .

- وعليه استخدام نموذج الذاكرة المشتركة أسهل للبرنامج من نموذج تمرير الرسالة

■ العيب الواضح المشترك هو النقاط الساخنة (hot spot)

- تحاول المعالجات أن تكتب على نفس خلية الذاكرة . و لأن المعالجات يجب أن تنتظر حتى تكون خلية الذاكرة جاهزة فإن ذلك يمكن أن يعيق أداء النقاط الساخنة
- العيب الآخر هو أن المبرمج والمترجم compiler ونظام التشغيل يجب أن تقرر كيف يُقسّم البرنامج على عدة معالجات

• تمرير الرسائل MIMD Message Passing:

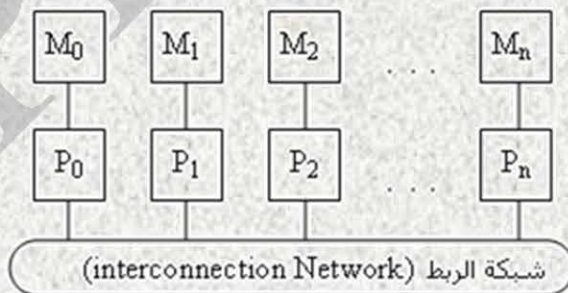
- كل معالج له ذاكرة داخلية خاصة به، ولكي تتواصل المعالجات فيما بينها فإنها ترسل رسائل إلى كل معالج عن طريق شبكة الربط
- بشكل مشابه لنموذج الذاكرة المشتركة، شبكة الربط يمكن أن تأخذ عدة أشكال مختلفة والأكثر استخداماً هي شبكة المكعب متعدد الأبعاد حيث تتوضع المعالجات في زوايا المكعب
- يتميز أسلوب توجيه الرسالة بسرعه مما يشكل أمراً هاماً لأداء الآلات من هذا النوع
- الحسنة:

- لا يوجد ذاكرة مشتركة، وهذا يعني اختفاء المشاكل الناتجة عن التداخل والتضارب في الذاكرة والتي يعاني منها نموذج الذاكرة المشتركة.

■ النفاذ أو الوصول إلى الذاكرة سريع

○ الأهداف:

- يعمل كأداة اتصال لعبور قيم المعطيات بين المعالجات
- يعمل كآلية تزامن للخوارزميات المتوزية



- بما أن المعالجات لا تشترك في ذاكرة واحدة فإن نموذج تمرير الرسائل يمكن أن يحتوي على عدد كبير جداً من المعالجات، بخلاف نموذج الذاكرة المشتركة

○ العيب الرئيسي هو الحمل الزائد الملقى على المبرمج،

- ليس كافيًا أن يقوم المبرمج بتقسيم البرنامج على المعالجات، بل عليه أيضًا أن يوزع المعطيات.
- برمجة الأجهزة من نوع تمرير الرسائل تتطلب من المبرمجين أن يعيدوا النظر في الخوارزميات كي يكون استخدام الآلة أكثر كفاءة