

# البرمجة التفرعية

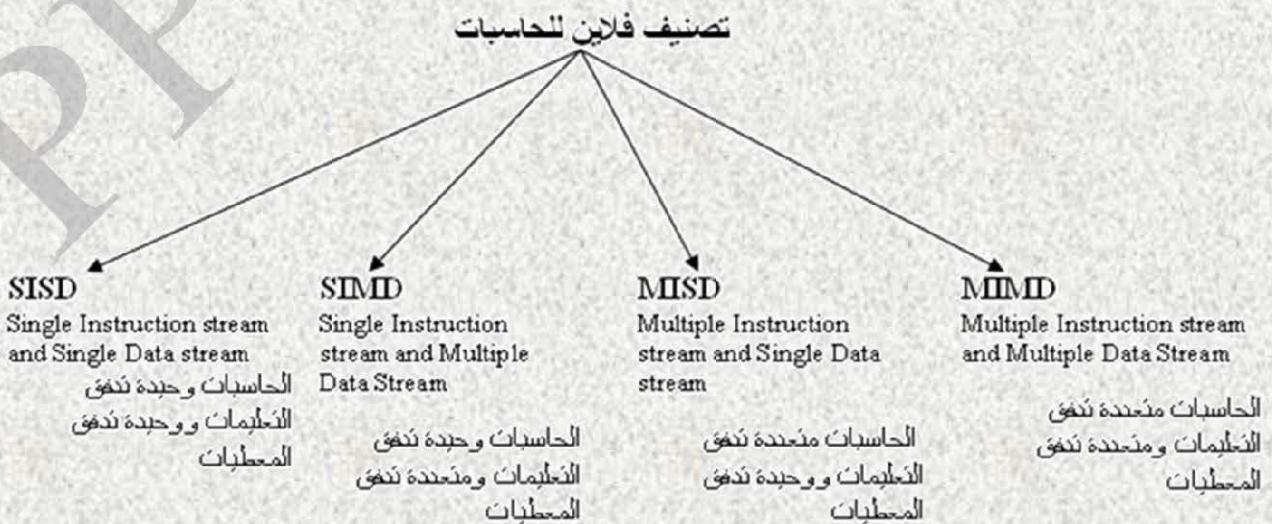
## Parallel Programming

### References

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

### الحواسيب التفرعية

- تصنیف الحاسوبات المتوازیة:
  - التصنیف الأکثر انتشارا هو ذلك التصنیف الذي اقترحه "میشيل فلاین" (Flynn) في عام 1966
    - يأخذ تصنیف فلاین بعين الاعتبار عاملین اثنین وهم:
      - كمية السریان (أو التدفق) للتعليمات
      - و كمية السریان للبيانات التي تتدفق للمعالج.
  - تصنیف فلاین :Flynn's Classification Scheme
    - يعتمد أساساً على كمية تدفق البيانات والتعليمات الموجودة في الآلة.
    - يقصد بالتدفق هنا على أنه تتبع أو تسلسل لعناصر (التعليمات أو المعطيات) كمانفذت بواسطة المعالج.
    - مثال، تقوم بعض الآلات بتنفيذ تدفق واحد من التعليمات، بينما يتم تنفيذ عدة تدفقات في آلات أخرى. وبنفس الطريقة في بعض الآلات ترجع تدفقاً واحداً من المعطيات، وألات أخرى ترجع تدفقات متعددة.
    - وعليه فإن فلاین يضع الآلة في تصنیف واحد من بين أربعة وذلك اعتماداً على وجود تدفق واحد أو تدفقات متعددة.



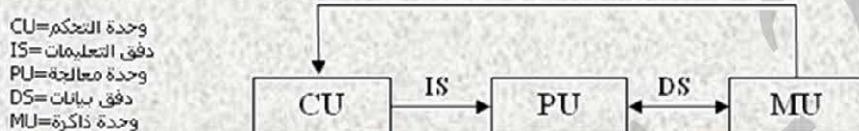
## الحاسبات وحيدة تدفق التعليمات ووحيدة تدفق المعطيات :SISD

- يندرج تحت هذا الصنف جميع الحاسبات التسلسلية المعتادة، مثل Apple DEC VAX و Macintosh

- 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 أساساً 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 دفقاً واحداً من التعليمات



## الحاسبات وحيدة تدفق التعليمات ومترددة تدفق المعطيات :SIMD

- تعتمد على وحدة تعليمات واحدة تصدر أوامر إلى عدة وحدات معالجة وعليه تحتاج إلى تدفق متعدد للمعطيات

- عادة تصدر وحدة التعليمات نفس الأمر إلى جميع وحدات المعالجة.

- مثال: جميع وحدات المعالجة تنفذ تعليمية الجمع ADD ثم يتم تنفيذ تعليمية التخزين STORE وهكذا

- يتميز هذا النوع من الحاسبات بوجود وحدة تحكم مركبة.

- تنفذ وحدات المعالجة المختلفة تعليمية واحدة من وحدة التحكم وتقوم بتنفيذ هذه التعليمية بشكل متزامن على معاملات مختلفة.

- تكون هذه الحاسبات متزامنة، غالباً ما تملك ذاكرة مشتركة بين الوحدات.

- لتسهيل عملية اللوچ المتوازي إلى الذاكرة

- يُلْجأ إلى تقسيمها إلى بنوك مما يسمح باستخلاص عدة معاملات في نفس الوقت

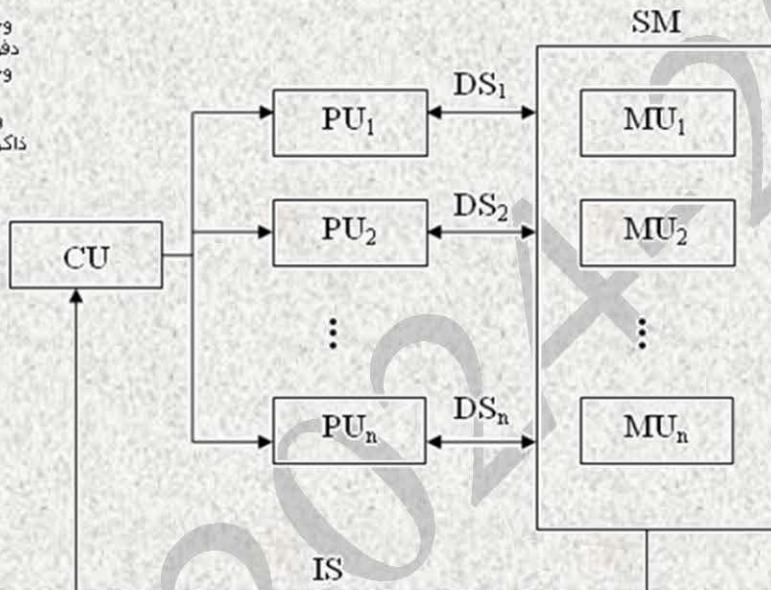
- يتم الاتصال بين وحدات المعالجة المختلفة وبنوك الذاكرة عبر شبكة الربط.

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

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

- تستخدم في هذه الحاسوبات آلاف المعالجات، وتكون عادةً متوسطة الأداء أو بسيطة جدًا، ووجودها وأدائها العالي ناتج عن العدد الكبير للمعالجات المستخدمة

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

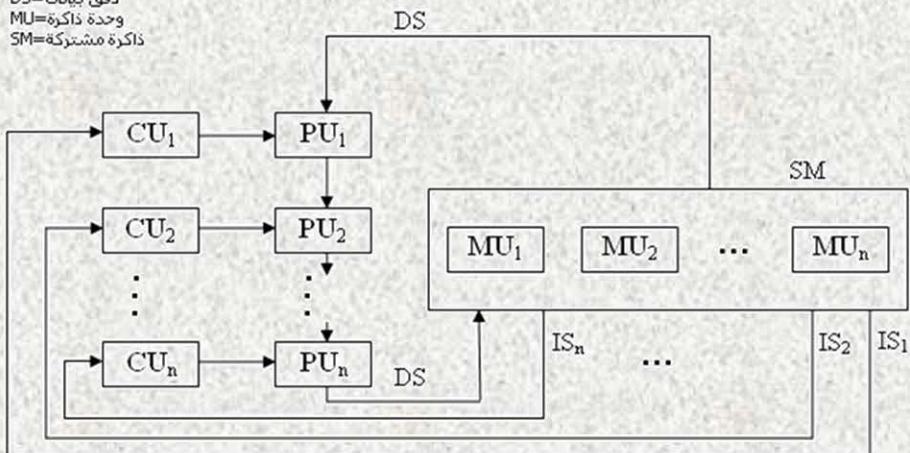


#### ▪ الحاسوبات متعددة تدفق التعليمات ووحيدة تدفق المعطيات: MISD

- 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 في تنفيذ المهام الجزئية الموكلة إليه.

- يوجد القليل من الحاسوبات المتوازية من هذا نوع حيث تعتبر قليلة الاستعمال في تعتمد مبدأ العمل التسلسلي

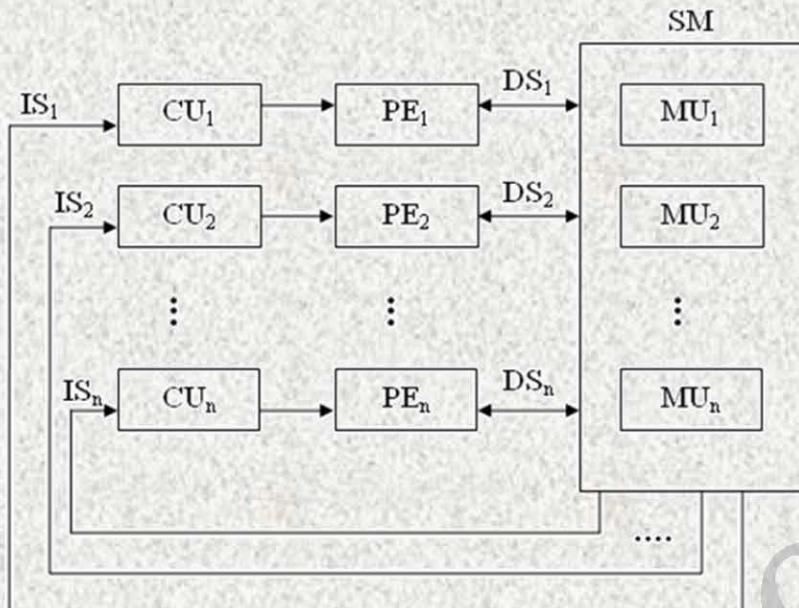
CU=وحدة التحكم  
 IS=دفق التعليمات  
 PU=وحدة معالجة  
 DS=دفق بيانات  
 MU=وحدة ذاكرة  
 SM=ذاكرة مشتركة



الحواسيب متعددة تدفق التعليمات ومتعددة تدفق المعطيات :MIMD

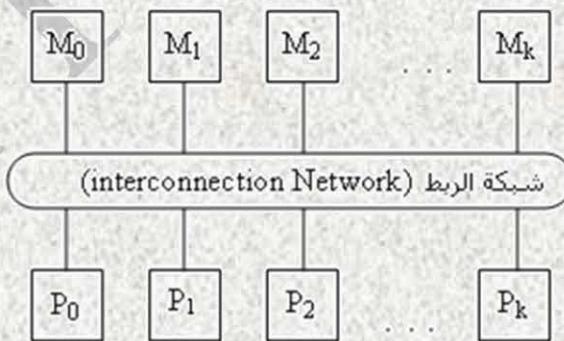
- يتم تنفيذ عدة تعليمات على معملات مختلفة خلال الدورة الزمنية للحاسِب.
- يتميز بوجود عدد من المعالجات التي تعمل بشكل مستقل عن بعضها البعض.
- يتضمن كل معالج وحدة تحكم خاصة به تساعدُه على تنفيذ المهام الجزئية الموكلة إليه.
- هذا النوع غير متزامنة وبالتالي فالحوادث التي تجري على معالج ما لا ترتبط بالحوادث التي تجري على المعالجات الأخرى.
- يمكن فرض نوع من التزامن بين هذه المعالجات عند الضرورات البرمجية لذلك
  - باستخدام بعض التعليمات الأولية المخصصة للتزامن
  - أو عن طريق العتاد
  - ويمكن أن يتم أيضًا عن طريق البرمجيات ونظم التشغيل
- أن كل معامل يعالج من قبل معالج ما بأمر ما، وفي الوقت ذاته تعالج جميع المعملات 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 مهام جزئية مستقلة جزئياً فيما بينها، وتتنفيذ كل مهمة على معالج من معالجات الحاسِب الأكثر استخداماً في الوقت الحالَة وتعتمد عليها الحاسِبات الحديثة
- يوجد لهذا التصنيف فتَيَن فرعيتين هامتين وهما:
  - الذاكرة المشتركة (Shared memory).
  - تمرير الرسائل (message passing).

CU=وحدة التحكم  
 IS=دفق التعليمات  
 PE=عصر معالجة  
 DS=دفق بيانات  
 MU=وحدة ذاكرة  
 SM=ذاكرة مشتركة



#### 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 تتدخل 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 للتزامن 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 .

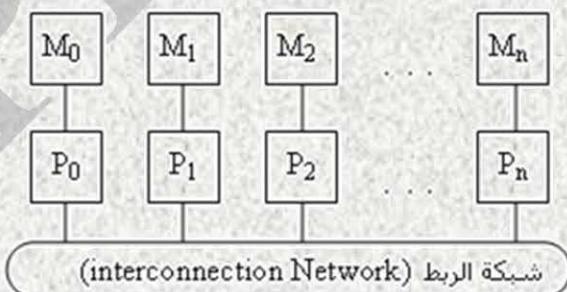
- وعليه استخدام نموذج الذاكرة المشتركة أسهل للبرنامج من نموذج تمرير الرسالة
- العيب الواضح المشترك هو النقاط الساخنة (hot spot)
- تحاول المعالجات أن تكتب على نفس خلية الذاكرة . و لأن المعالجات يجب أن تنتظر حتى تكون خلية الذاكرة جاهزة فإن ذلك يمكن أن يعيق أداء النقاط الساخنة
- العيب الآخر هو أن المبرمج والمترجم compiler ونظام التشغيل يجب أن تقرر كيف يقسم البرنامج على عدة معالجات

تمرير الرسائل:MIMD Message Passing

- كل معالج له ذاكرة داخلية خاصة به، ولكي تتوافق المعالجات فيما بينها فإنها توسل 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 بسرعته مما يشكل أمراً هاماً للأداء الآلات من هذا النوع
- الحسنات:
- لا يوجد ذاكرة مشتركة، وهذا يعني اختفاء المشاكل الناتجة عن التداخل والتضليل 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 ترامن للخوارزميات المتوزعة



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

PP 2024-2025