

تقنيات جدولة العمليات في معالج الحاسوب وأنظمة التشغيل

أ.م. د سامي ذياب عبدالرزاق

كلية المعارف الجامعة

المستخلص

يتم بموجبه استعراض تقنيات جدولة العمليات والتطور الحاصل فيها ابتداءً من عرض مفهوم قواعد الأسبقية الساكنة و فلسفتها و القواعد الديناميكية و فلسفتها و تحديد ذات البعد الواحد منها و ذات الأبعاد المتعددة فيها، ثم مناقشة تقنيات جدولة العمليات في وحدة المعالجة المركزية للحاسوب (CPU) سواء كانت أحادية النواة (one core) أم متعددة النواة (Multi core) من جانب، وفي أنظمة التشغيل المستخدمة في الحاسوب (Operating Systems) سواء كانت أحادية البرامج (Monoprogramming) أم متعددة البرامج (Multiprogramming) من جانب آخر

Abstract

It will be display operations scheduling techniques and their development by explaining the (static & dynamic priority) concept & philosophy, then determine (uni-directional & multi-directional priority). then discuss Operations scheduling inside central processing unit of computer (uni-core & multi-core processor) and inside operating systems of computer (Monoprogramming & Multiprogramming).

د- أهمية البحث :

تظهر أهمية البحث من خلال استعراضه لفلسفة تصميم أنظمة تشغيل الحاسوب و طرق معالجة البيانات و الاوامر فيها وفقا للتطور الحاصل في هذا المجال ، مما يسهل طريقة اختيار نظام الحاسوب المناسب لطبيعة الاعمال المراد تنفيذها و حجمها .

هـ - منهج البحث : يعتمد البحث على المنهج الوصفي التحليلي لعرض فلسفة جدولة العمليات وفقا لقواعد الاسبقية و فلسفة انواع أنظمة تشغيل الحاسوب و جدولة تنفيذ الاوامر فيها ، و التطبيق الفعلي لبيان كفاءة نظم التشغيل المتوازية و كلفتها .

ثانيا : تتابع الاعمال وقواعد الاسبقية (Job Sequencing and Priority Rules)

يعبر تتابع الاعمال عن الترتيب الذي بموجبه يتم معالجة الاعمال في مراكز العمل . فمجرد تحديد تواريخ البدء و تواريخ الاكمال سوف لن يضمن انجاز الاعمال كما مجدول لها ، مما يتطلب اعتماد قواعد تستخدم لتحديد اسبقيات تتابع الاعمال على مراكز العمل ، فيتم اشتغال او انجاز (معالجة) (n) من الاعمال في (m) من مراكز العمل او المكائن . فيكون امامنا (ni) من البدائل التي يمكن ان تحدد الجدولة المثلى التي تملي على مديري العمليات باختيار القرار الامثل للجدولة و الذي يقود الى اكمال الاعمال بالاوقات المحددة لها قدر الامكان و حل اشكالات تنافس الاعمال المختلفة على الموارد المشتركة و تجاوز عطلات المكائن و تغيب العاملين و تحسين الكفاءة من حيث اوقت و التكلفة و الاسبقية (Stevenson ,93,743) و تعبر طرائق التتابع عن الترتيب الذي تنجز بموجبه الاعمال ، ويشار الى هذه الطرائق بقواعد الاسبقية لارسال الاعمال (Priority Rules For Dispatching Jobs) الى مراكز العمل ، اذ تقوم بتقديم ارشادات الى المشرفين عن التتابع الذي ينبغي ان تقدم بموجبه الاعمال لغرض الانجاز عندما تصبح مراكز العمل مهيئة لاستقبال عمل جديد (Heizer & Render,88,618) لقد تم تطوير قواعد مختلفة حسب الهدف لكل منها ، و يتفق الباحثون على القواعد الاكثر شيوعا و استخداما في الحياة العملية لتحديد اسبقيات تتابع الاعمال ، فمنها ما

ثالثاً : نظم تشغيل الحاسوب (operating systems - OS)

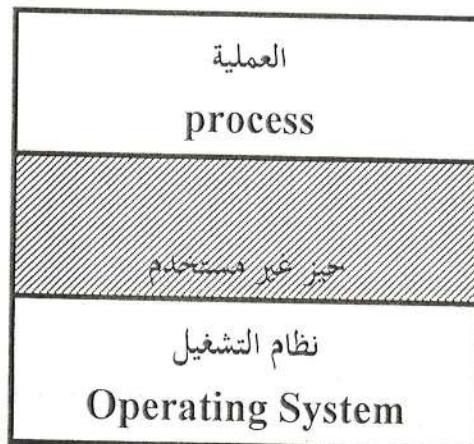
نظام التشغيل هو مجموعة من البرمجيات المسؤولة عن إدارة موارد وبرمجيات الحاسوب، ويمثل وسيط بين المستخدم وموارد الحاسوب، هو مفضلة لتشغيل برامج المستخدم، يقوم بالمهام الأساسية مثل إدارة وتخصيص مصادر الحاسوب (الذاكرة، القرص الصلب، الوصول للأجهزة الطرفية الملحقة..إلخ) ، ترتيب أولوية التعامل مع الأوامر، التحكم في أجهزة الإدخال والإخراج، تسهيل الشبكات، وإدارة الملفات

(نظام_تشغيل/ <http://ar.wikipedia.org/wiki>) .

أ- نظم التشغيل أحادية البرامج (Monoprogramming)

هو من أقدم أنظمة التشغيل ، كان يستخدم مع بدايات الحاسبات ومازال مستخدماً في بعض المواقع. تكون إدارة الذاكرة فيه بسيطة جداً وذلك لان الأسلوب المتبع فيه هو إتاحة جميع موارد الحاسب لعملية واحدة ، بحيث تكون باقي العمليات في حالة خمود حتي تنتهي العملية من التنفيذ ، اذ يتم تسكين عملية واحدة في الذاكرة الرئيسية بحيث أن لايزيد حيز العملية عن الحيز المتاح في الذاكرة وعلي هذا الأساس يتم تقسيم الذاكرة الرئيسية الي ثلاثة أجزاء: جزء خاص بالعملية المراد تنفيذها ، وجزء خاص بنظام التشغيل، وعادة يكون هنالك حيز غير مشغول وهو الجزء الثالث، كما في (شكل رقم ١)

(http://www.uop.edu.jo/e-courses/www.kutub.info_1396).



(http://ar.wikipedia.org/wiki/تاريخ_انظمة_التشغيل)

المهمة (task) هي "المسار التنفيذي لمجموعة من عناوين الذاكرة ، أذ يتم تحميل عنوان موقع الذاكرة الذي يحوي اول اوامر البرنامج في سجل العناوين (Address Register) في وحدة المعالجة المركزية (CPU) ، لتبدأ تنفيذ العمليات تباعا (Processes)

اما العملية (process) هي عبارة عن برنامج أو جزء منه قيد التنفيذ.

(<http://www.go4expert.com/forums/showthread.php?t=681>)

اطلق على نظام (time-sharing) تسمية متعدد المهام (Multitasking) أذ يسمح نظام التشغيل لاكثر من برنامج ان تحمل إلى الذاكرة وتنفذ في مدة زمنية متداخلة بتجزئة الزمن time slicing حيث اذ يقوم معالج مركزي بالتبديل بينها ، ولكي يستطيع نظام التشغيل من إدارة العمليات فانة يخصص لكل عملية ما يأتي:

- مساحة عنوان : "Address Space" عبارة عن المساحة المحجوزة بالذاكرة الرئيسة للعملية الحالية.
- شفرة البرنامج : "Program Code" عبارة عن الايعازات المطلوب تنفيذها.
- مؤشر التكديس . Stack Pointer
- عداد البرامج . Program Counter
- السجل Register
- الكومة "النثريات" Heap "

بنقلها الى المعالج، وفي حالة حدوث اي مقاطعة او طلب ادخال او اخراج من المستخدم يقوم المجدول متوسط المدى "Med Term Scheduler" بنقلها من حالة التنفيذ إلى الانتظار (Stallings,2005) .

ويمكن تصنيف العمليات الى انواع حسب :

• موقع التنفيذ : يتضمن الاتي

- عمليات نمط المعالجة: CPU Bound Process : و هي العمليات التي تقضي معظم وقتها في

المعالج

- عمليات نمط الادخال و الاخراج I/O Bound Process : و هي العمليات التي تقضي معظم وقتها

في عمليات الادخال و الاخراج

يجب أن تقوم المجدولات (Scheduler) باختيار خليط من النوعين السابقين لكي يتم استغلال المعالج

بشكل مثالي.

• حسب التعاونية :يتضمن الاتي :

-عمليات غير متعاونة (مستقلة) : و هي العمليات التي لا تتأثر و لا تؤثر في تنفيذ العمليات

الاخري

- عمليات متعاونة : و هي العمليات التي تتأثر و تؤثر في تنفيذ العمليات الاخرى

٢- الانظمة المتوازية (parallel systems)

ان الحاجة إلى سرعة أكبر وكفاءة أكثر قادت إلى تصميم الأنظمة المتوازية التي تتلخص في وجود أكثر من

وحدة معالجة مركزية على نفس الجهاز، كل منها تُستخدم لخدمة برنامج أو جزء من برنامج ما ، يعني

استخدم التوازي لسنوات عدة، ولاسيما في الحواسيب عالية الأداء، وازداد الاهتمام به أكثر في الآونة الأخيرة بسبب العوائق المادية التي تحول دون توسيع حجم العمليات المعالجة (Adve et a 2008). فصار استهلاك أجهزة الحاسوب للطاقة (ومن ثم توليد الحرارة) مصدرا للقلق في السنوات الأخيرة، فإن الحواسيب المتوازية أصبحت الامتواج المهيمن في هندسة الحواسيب، ولاسيما في نماذج المعالجات متعددة النواة. ان كتابة البرامج الموجهة للحواسيب المتوازية أكثر صعوبة من البرامج ذات المهام التسلسلية، لأن توازي المهام يسمح بظهور أنواع جديدة محتملة من الأخطاء البرمجية، والتي من أشهر أمثلتها مشكلة حالة تعارض الاتصالات والتزامن بين المهام الفرعية المختلفة (Patterson & Hennessy, 1998,715) .

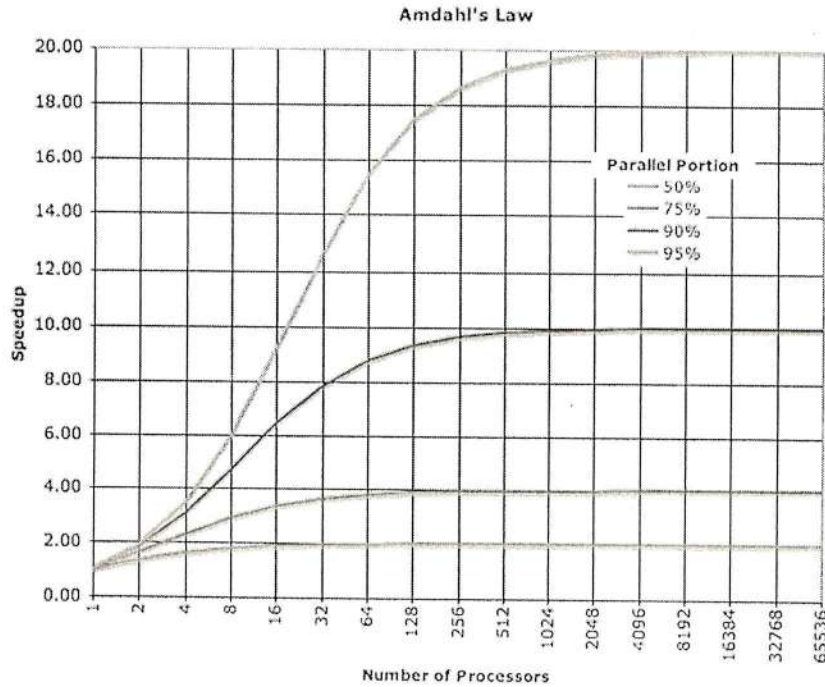
ان رفع سرعة تنفيذ العمليات كان الهم الرئيسي لتحسين أداء الحواسيب منذ منتصف ثمانيات القرن العشرين وحتى عام 2004 ، إذ أن مدة تنفيذ برنامج تساوي عدد الاوامر مضروبا في متوسط الوقت اللازم لتنفيذ تعليمة واحد. فالحفاظ على كل شيء ثابت مع زيادة تردد الساعة الداخلية يخفض من متوسط الوقت المستغرق لتنفيذ تعليمة ، فيقلل وقت التنفيذ لجميع البرامج ذات العمليات المحدودة (Patterson & Hennessy, 2002,43).

$$P = C \times V^2 \times F \quad : \text{ بحسب استهلاك الشرائح للطاقة من خلال المعادلة}$$

اذ: P - الطاقة، و C - عدد المكثفات المستعملة أثناء عمل دورة كاملة (تناسب مع عدد الترانزستورات التي تتغير مدخلاتها)، و V - الجهد، أما F - تردد المعالج (دورة في الثانية).

ان الزيادة في التردد يزيد كمية الطاقة المستهلكة ، مما أجبر شركة إنتل في مايو 2004 بالغاء إنتاج معالجاتها Jayhawk و Tejas ، اذ تعد نهاية سياسة رفع التردد كأنموذج شائع لمعمارية الحواسيب

(Flynn, 2004).



شكل رقم (٤) تمثيل رسومي لقانون أمدال

قانون غوس تافسون هو قانون آخر في هندسة الحاسبات، ويرتبط ارتباطاً وثيقاً بقانون أمدال. ويمكن

$$S(P) = P - \alpha(P - 1) \quad \text{صياغته على النحو الآتي:}$$

اذ P هو عدد المعالجات، S هو مقدار التسريع و α هو الجزء غير القابل للموازاة من هذه العملية. يفترض

قانون أمدال أن للمشكلة حجماً ثابتاً وأن حجم المقطع المتسلسل مستقل عن عدد المعالجات، في حين أن

القانون غوستافسون لم يتم بهذه الافتراضات (ACM, 1988, 532-533).

• حالة السباق والاستبعاد المتبادل، والتباطؤ في التوازي

في برنامج مواز غالباً ما تسمى المهام الفرعية خيوطاً (threads). بعض هندسة الحواسيب المتوازية

تستخدم نسخ صغيرة و خفيفة من الخيوط المعروفة باسم الألياف، بينما تستخدم هندسات أخرى نسخ أكبر

تسمى بالعمليات. ومع ذلك فإن "الخيوط" مقبولة عموماً كمصطلح عام عن المهام الفرعية. غالباً ما تحتاج

سوف يقوم خيط واحد بقفل المتغير س، في حين أن الخيوط الأخرى ستكون مستبعدة، وغير قادرة على المضي قدما إلى أن يتم تحرير المتغير س مرة أخرى. وهكذا يتم ضمان تنفيذ البرنامج بشكل صحيح. لكن هذه العملية سوف تبطئ هذا البرنامج بشكل كبير. إن قفل عدة متغيرات باستخدام أقفال غير موحدة يدخل البرنامج في طريق مسدود. كثير من البرامج المتوازية تتطلب تزامنا بين مهامها الفرعية، وهذا يتطلب استخدام حاجز. والذي عادة ما ينفذ باستخدام قفل برمجي. فئة من الخوارزميات، تعرف بـ إقفال- تحرير، و انتظار-تحرير، فهي تتجنب استخدام الأقفال والحواجز تماما. ولكن هذا النهج هو صعب التنفيذ غالبا ويتطلب تصميم تركيبية بيانات بشكل صحيح.

لا تؤدي كل عمليات الموازاة إلى التسريع. فكلما قسمت المهام إلى خيوط أكثر فاكتر، فإن تلك الخيوط تستهلك جزء زائد من الوقت في التواصل مع بعضها. فإن الوقت الزائد من عملية الاتصال يهيمن على الوقت الذي يستغرقه حل المشكلة، وكذلك عملية الموازاة تزيد بدلا من أن تقلل من مقدار الوقت المطلوب للانتهاء. يعرف هذا بتباطؤ التوازي. (parallel slowdown).

• توازي الحبيبات الناعمة والحبيبات الخشنة والمريك

تصنف التطبيقات وفقا لمدى حاجة المهام الفرعية للمزامنة أو التواصل مع بعضها البعض. إذ يظهر التطبيق توازي الحبيبات الناعمة إذا كانت المهام الفرعية تتواصل عدة مرات في الثانية الواحدة، أو توازي الحبيبات الخشنة في عدم التواصل لعدة مرات في الثانية الواحدة، ويصنف بالتوازي المريك في ندرة أو انعدام التواصل. و هذا ينشر بأن التطبيقات التي تعد من صنف التوازي المريك هي أسهل للموازاة (Lamport,)

• أنواع التوازي

- التوازي على مستوى البت

منذ بدايات تكنولوجيا دوائر التكامل الفائقة (VLSIC) (very-large-scale integration circuits) في تصنيع رقائق الحاسوب في سبعينات القرن العشرين وحتى عام ١٩٨٦، كانت تتم عملية التسريع في هندسة الحاسبات من خلال مضاعفة حجم كلمة الحاسوب (مقاسة بالبت) ، والتي بدورها تزيد كمية المعلومات التي يمكن للمعالج أن يعالجها في كل دورة. إذ ان زيادة حجم الكلمة يقلل من عدد الاوامر التي يجب على المعالج تنفيذها لأداء عملية على متغيرات حجمها أكبر من طول الكلمة. فعندما نريد جمع رقمين صحيحين مكونين من ١٦ بت على معالج ٨ بت، فإن المعالج يقوم بجمع اول ٨ بت من كل رقم باستخدام امر الجمع، ثم يجمع ثاني ٨ بت مع الإعارة من العملية السابقة، ومن ثم فإن معالج ٨ بت يحتاج لامرين لإكمال عملية واحدة، بينما معالج ١٦ بت سيكون قادراً على إكمال العملية بامر واحدة.

تم استبدال المعالجات ٤-بت بمعالجات ٨ بت ثم ١٦ بت ، ثم ٣٢ بت. وفي وقت قريب (٢٠٠٣-٢٠٠٤)، ظهرت هندسة x64 ، والتي تحوي على معالجات ٦٤ بت وشاع استخدامها (Culler et al, 1999, 15) .

- التوازي على مستوى الاوامر

ان جوهر برنامج الحاسوب هو تدفق الاوامر التي ينفذها المعالج، ويمكن إعادة ترتيب هذه الاوامر ودمجها في مجموعات يتم تنفيذها بالتوازي دون تغيير نتيجة هذا البرنامج. وهذا ما يسمى بالموازاة على مستوى الاوامر . اذ تطورت فكرة الموازاة على مستوى الاوامر وهيمنت على هندسة الحاسبات من منتصف ثمانينات وحتى منتصف تسعينات القرن العشرين (Culler et al,1999,15) .

IF	ID	EX	MEM	WB					
IF	ID	EX	MEM	WB					
	IF	ID	EX	MEM	WB				
	IF	ID	EX	MEM	WB				
		IF	ID	EX	MEM	WB			
		IF	ID	EX	MEM	WB			
			IF	ID	EX	MEM	WB		
			IF	ID	EX	MEM	WB		
				IF	ID	EX	MEM	WB	
				IF	ID	EX	MEM	WB	

شكل رقم (٦ ب)

سوبرسكيلر معالج بخمس مراحل متوالية، قادر على تمرير امرين في كل دورة.

- التوازي على مستوى البيانات

هي عملية الموازاة المتواجدة في جمل التكرار في البرنامج، و التي تركز على توزيع هذه البيانات عبر عقد الحاسوب المختلفة بحيث تتم معالجتها بشكل متواز ، وغالبا ما تؤدي موازاة التكرار بصورة مماثلة إلى تسلسل العمليات أو المهام التي تنجز بناء على عناصر بيانات ذات هياكل كبيرة، فالعديد من التطبيقات العلمية والهندسية تستعمل توازي البيانات (Culler et al., 1999, 125).

- توازي المهام

هي صفة البرامج المتوازية التي تحوي حسابات مختلفة تماما يمكن أن تنفذ في المجموعة نفسها أو في مجموعة مختلفة من البيانات، وهذا يتناقض مع موازاة البيانات، حيث إذ يتم تنفيذ نفس العملية الحسابية على المجموعة نفسها أو مجموعات مختلفة من البيانات ، وهولا يتناسب عادة مع حجم المشكلة (Culler et al., 1999, 125).

تستفيد نظم الحاسوب من مخزن صغير وسريع بالقرب من المعالج لتخزين مؤقت لنسخ من قيم الذاكرة يسمى (cache memory). أما أنظمة الحواسيب المتوازية فتواجه صعوبات مع هذه المخازن التي قد تخزن القيمة نفسها في أكثر من مكان واحد، مع إمكانية تنفيذ غير صحيح للبرنامج ، فتتطلب نظام تخزين متناسق يتتبع القيم المخزونة مؤقتا ويعمل على تطهير استراتيجي لها، وبالتالي ضمان تنفيذ صحيح للبرنامج . (Patterson and Hennessy, 1998, 713) . يمكن تنفيذ الاتصالات بين الذاكرة والمعالجات وفيما بين المعالجات باستعمال أجهزة وطرق متعددة، عبر ذاكرة مشتركة أو عارضة تبديل (crossbar switch) أو ناقل مشترك أو شبكة ربط لعدد كبير من انواع الشبكات كالطوق والنجمة والشجرة ، بأكثر من معالج في العقدة الواحدة . إذ تحتاج الحواسيب المتوازية (المبنية على أساس شبكات الربط) إلى نوع من التوجيه لتمكينها من تمرير الرسائل بين العقد التي لا توجد علاقة مباشرة بينها . ومن المرجح أن تكون الوسيلة المستخدمة للاتصال بين المعالجات هي التسلسل الهرمي في الأجهزة الكبيرة متعددة المعالجات.

ب - فئات الحواسيب المتوازية

تصنف الحواسيب المتوازية على وفق المستوى الذي يدعم فيه الجهاز عملية التوازي كالآتي :

١ - حواسيب متعددة النواة (multicore)

يتضمن معالج الحاسوب متعدد النواة عدة وحدات تنفيذ (النوى) على الشريحة نفسها. تختلف هذه المعالجات عن معالجات سوبرسكيلر (superscalar) ، إذ تتمكن المعالجات متعددة النواة من تمرير اوامر متعددة لكل دورة عبر قنوات اوامر متعددة . وكل نواة في المعالج متعدد النواة يمكنها أن تكون سوبرسكيلر . ففي كل دورة ، يمكن لكل نواة تمرير اوامر متعددة من قناة واحدة. و من امثلة المعالجات متعددة النواة (intel dual core) و (intel core2 duo) و (intel i3) و (intel i5) .

• الحواسيب الشبكية (Grid computing)

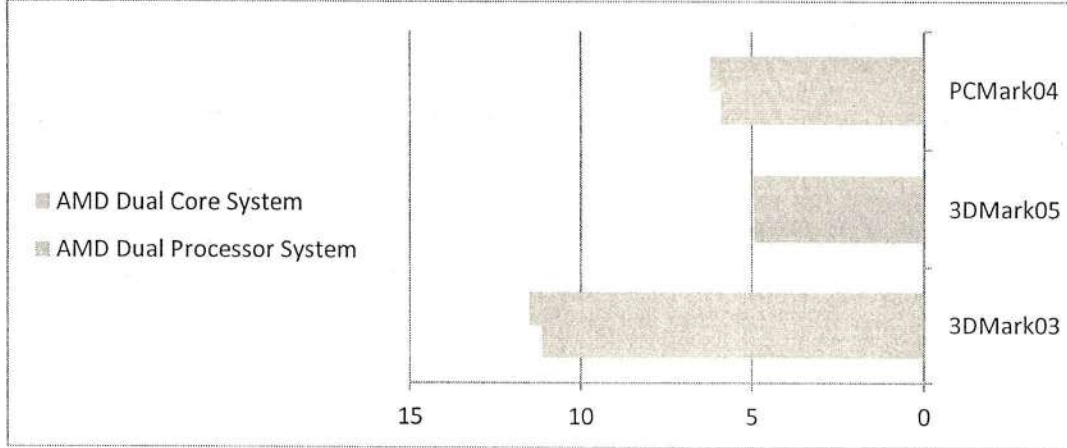
هو الشكل الأكثر انتشارا للحواسيب المتوازية ، يستخدم اتصال أجهزة الحاسوب عبر الإنترنت للعمل على معالجة مشكلة معينة. وبسبب انخفاض سرعة تدفق البيانات والبطء الناتج عن التباعد العالي للحواسيب على شبكة الإنترنت، فإن الحواسيب الشبكية تعالج مشاكل التوازي الحرج فقط . و قد تم إنشاء عدة برامجيات للحواسيب الشبكية، أشهرها SETI@home و Folding@Home معظم برامجيات الحواسيب الشبكية تستخدم برامجيات وسيطة (middleware) ، وهي برامجيات تعمل بين نظام التشغيل والبرامجيات الأخرى لإدارة موارد الشبكة وتوحد الواجهة البرمجية. ان أكثر البرامجيات الوسيطة للحواسيب الشبكية شيوعا هي بنية باركلي التحتية المفتوحة للحواسيب الشبكية (Berkeley Open Infrastructure for Network Computing (BOINC)). (Kirkpatrick & Scott ,2003, 668 - 669.)

خامسا : الجانب التطبيقي :

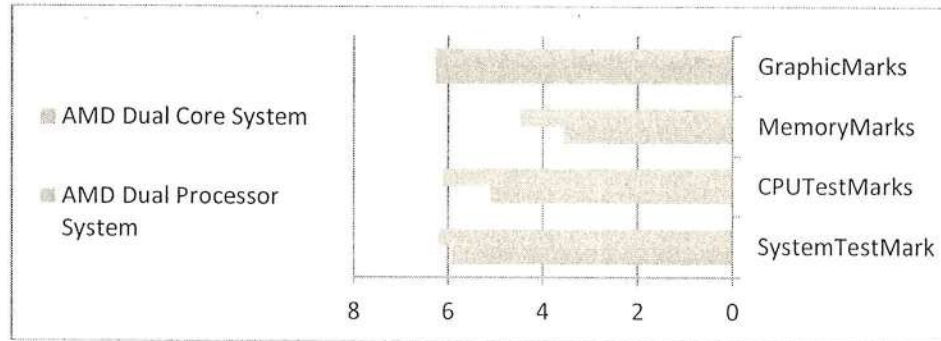
يتضمن هذا الجانب مناقشة عملية لآلية اختيار الحاسوب المناسب لعمل خادم شبكة (Network Server) او منصة اعمال متخصصة (Workstation) تستخدم لمعالجة الرسومات والفيديو او انظمة الوقت الحقيقي . و تتضمن عينة البحث حاسوبين بمعالجين ثنائية النواة (Dual core) و مقارنتهما بحاسوبين ثنائية المعالجات (Dual processor) من انتاج شركتي (AMD & Intel). اذ سيتم مقارنة العلامات التي تسجلها برامج قياس الكفاءة (PCMark,3DMark) حيث تسجل هذه البرامج علامات كفاءة الجهاز من حيث تعامله من التطبيقات ثلاثية الابعاد ، والاستغلال الامثل للمعالج ، والاستغلال الامثل لوحدة الذاكرة . اضافة الى مقارنة بين الكلفة الخاصة بكل نوع .

نتائج برامج فحص الكفاءة (PCMarks , 3DMarks) فهي كالآتي

Overall Scores



PCMark04 Scores

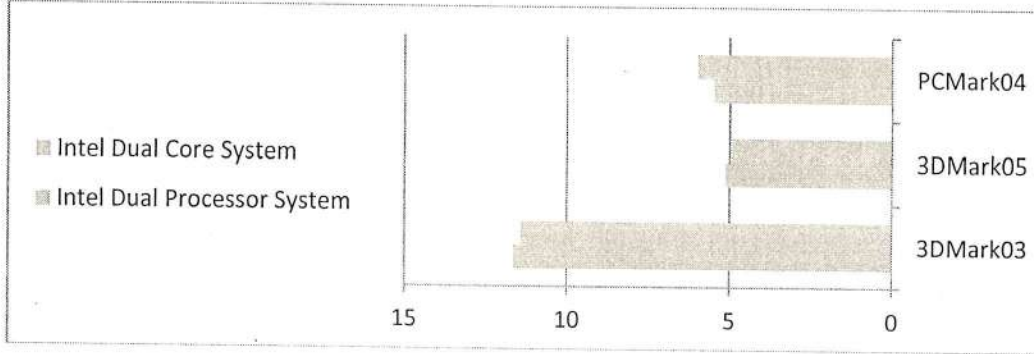


يلاحظ من المخططات اعلاه :

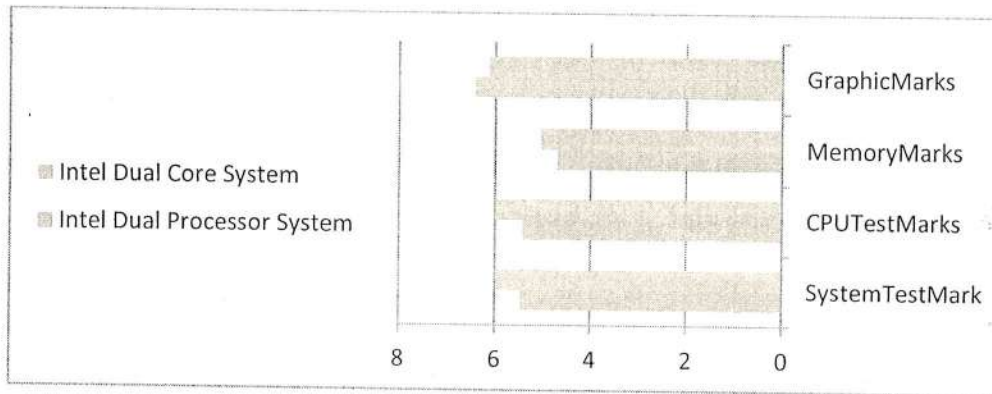
- ١- ان الاختلافات بين النموذجين طفيفة جدا من ناحية التأثير في الاداء .
- ٢- توجد افضلية نسبية للحاسوب ثنائي النواة (AMD Dual Core) قياسا بالحاسوب ثنائي المعالجات (AMD Dual Processor) و ذلك بسبب اعتمادها على ذاكرة مشتركة بين النواتين في الحاسوب الواحد مما يسرع تداول البيانات و العمليات بين النواتين ، و يشترط في النوع الثاني وجود ذاكرتين منفصلتين كل واحدة منها تتعامل مع المعالج الخاص بها وهذا يسبب بطأ تداول البيانات والعمليات بين المعالجين .
- ٣- ان كلفة الحاسوب ثنائي النواة قدرها (\$ 424) وهي اقل من كلفة الحاسوب ثنائي المعالجات و التي قدرها (\$ 710)

نتائج برامج فحص الكفاءة (3DMarks , PCMarks) فهي كالآتي

Overall Scores



PCMark04 Scores



يلاحظ من المخططات اعلاه :

- ١- ان الاختلافات بين النموذجين طفيفة جدا من ناحية التأثير في الاداء .
- ٢- توجد افضلية نسبية للحاسوب ثنائي النواة (Intel Dual Core) قياسا بالحاسوب ثنائي المعالجات (intel Dual Processor) و ذلك بسبب اعتمادها على ذاكرة مشتركة بين النواتين في الحاسوب الواحد مما يسرع تداول البيانات و العمليات بين النواتين ، و افضلية للحاسوب ثنائي المعالج (Intel Dual Processor) من ناحية تعامله مع الرسومات .
- ٣- ان كلفة الحاسوب ثنائي النواة قدرها (\$ 380) وهي اقل من كلفة الحاسوب ثنائي المعالجات و التي قدرها (\$ 685)

بالتوازي في ان واحد دون الحاجة الى زيادة سرعة المعالجات ، وكذلك اعتماد التوازي في استغلال موارد الحاسوب من ذاكرة و اساليب اتصال داخلي ، مما يزيد كفاءة معالجة و تنفيذ الاوامر داخل الحاسوب .

٦- تعالج الحواسيب المتوازية مشاكل شبكات الاتصالات و ذلك باعتماد اساليب ربط عنقودية و اخرى عالية التوازي للحواسيب تنسجم مع الزيادة الهائلة للاوامر داخل الشبكة ذاتها .

٧- ان كفاءة الحاسوب لا تعتمد فقط على سرعة المعالج فيه و انما تعتمد على طريقة معالجة البيانات و الاوامر فيه .

٨- ان استخدام الحواسيب متعددة النواة من شركتي intel & AMD هو اكفاً من اعتماد الحواسيب متعددة المعالجات للشركتين ، وهو اقل كلفة منه ايضا .

٩- ان استخدام الحواسيب متعددة المعالجات من شركة (intel) هي اكفاً في معالجة الرسومات من الحواسيب متعددة النواة المتشابهة من حيث المواصفات للشركة ذاتها.

ب- التوصيات

١. ضرورة الاهتمام بجدولة الاعمال على المكان و داخل الحاسوب و حسب القواعد المناسبة بالاستفادة من التطور التقني و التكنولوجي لضمان سرعة و دقة انجازها .
٢. التركيز على فهم فلسفة انظمة الحاسوب المتوازية و العمل بموجبها كونها تواكب التطور الحاصل و المتوقع مستقبلا لأنظمة الحاسوب و المعالجة فيه .
٣. اعتماد الحواسيب متعددة النواة لشركتي (AMD & Intel) لكفاءتهما و قلة الكلفة فيهما قياسا بالحواسيب متعددة المعالجات و المتشابهة من حيث المواصفات .