

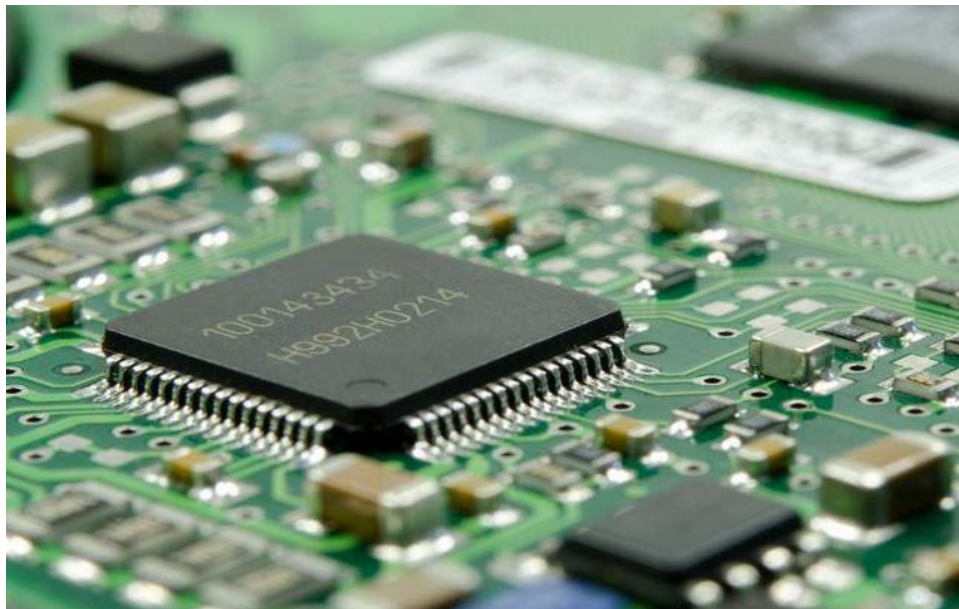


جامعة القادسية
كلية التربية



Lecture 24

Computer Architecture



Prepared by:

Firas Abdulrahman Yosif

Branching

A **branch** is an instruction in a computer program that can cause a computer to begin executing a different instruction sequence and thus deviate from its default behavior of executing instructions in order. Branch may also refer to the act of switching execution to a different instruction sequence as a result of executing a branch instruction. Branch instructions are used to implement control flow in program loops and conditionals (i.e., executing a particular sequence of instructions only if certain conditions are satisfied).

الفرع عبارة عن تعليمات في برنامج كمبيوتر يمكن أن تجعل الكمبيوتر يبدأ في تنفيذ تسلسل تعليمات مختلف وبالتالي ينحرف عن سلوكه الافتراضي في تنفيذ التعليمات بالترتيب. قد يشير الفرع أيضًا إلى عملية تبديل التنفيذ إلى تسلسل تعليمات مختلف نتيجة لتنفيذ تعليمات الفرع. تُستخدم تعليمات الفرع لتنفيذ التحكم في التدفق في حلقات البرنامج والشروط الشرطية (أي تنفيذ تسلسل معين من التعليمات فقط في حالة استيفاء شروط معينة).

A branch instruction can be either an **unconditional branch**, which always results in branching or a **conditional branch**, which may or may not cause branching depending on some condition. Also, depending on how it specifies the address of the new instruction sequence (the target address).

يمكن أن تكون تعليمات الفرع إما فرعًا غير مشروط ، والذي ينتج عنه دائمًا تفرع أو فرع شرطي ، والذي قد يتسبب أو لا يتسبب في التفرع حسب بعض الشروط. أيضًا ، اعتمادًا على كيفية تحديد عنوان تسلسل التعليمات الجديد (العنوان الهدف).

A branch instruction is generally classified as *direct*, *indirect* or *relative*. It means the instruction contains the target address, specifies where the target address is to be found (e.g., a register or memory location), or specifies the difference between the current and target addresses. A branch instruction computes the target address in one of four ways:

- A target address is the sum of a constant and the address of the branch instruction itself.
- The target address is the absolute address given as an operand to the instruction.
- The target address is the address found in the Link Register.
- The target address is the address found in Count Register.

يتم تصنيف تعليمات الفرع عمومًا على أنها مباشرة أو غير مباشرة أو نسبية. يعني أن التعليمات تحتوي على العنوان الهدف ، وتحدد مكان العثور على عنوان الهدف (على سبيل المثال ، موقع السجل أو الذاكرة) ، أو تحدد الفرق بين العنوانين الحالي والهدف. تحسب تعليمات الفرع العنوان الهدف بإحدى الطرق الأربع:

1. عنوان الهدف هو مجموع ثابت وعنوان تعليمات الفرع نفسه.

2. العنوان الهدف هو العنوان المطلق المعطى كمعامل للتعليمات.

3. عنوان الهدف هو العنوان الموجود في Link Register.

4. العنوان الهدف هو العنوان الموجود في Count Register.

The target address can be computed sufficiently ahead of the branch to pre-fetch instructions along the target path using the first two methods.

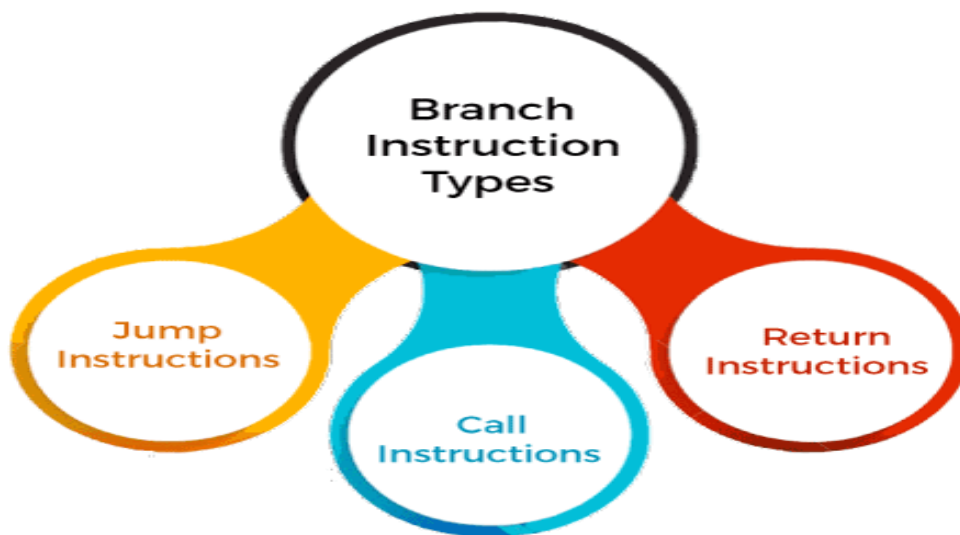
Using the third and fourth methods, pre-fetching instructions along the branch path is also possible provided the Link Register or Count Register is loaded sufficiently ahead of the branch instruction.

يمكن حساب عنوان الهدف بشكل كافٍ قبل الفرع لجلب التعليمات مسبقاً على طول المسار الهدف باستخدام الطريقتين الأوليين.

باستخدام الطريقتين الثالثة والرابعة ، يمكن أيضاً جلب تعليمات الجلب المسبق على طول مسار الفرع بشرط تحميل Link Register أو Count Register بشكل كافٍ قبل تعليمات الفرع.

- **Types of Brach Instructions**

There are three types of branching instructions in computer organization:



1. Jump Instructions

The jump instruction transfers the program sequence to the memory address given in the operand based on the specified flag. Jump instructions are further divided into two parts, Unconditional Jump Instructions and Conditional Jump Instructions.

تنقل تعليمات القفز تسلسل البرنامج إلى عنوان الذاكرة الوارد في المعامل بناءً على العلامة المحددة. تنقسم تعليمات القفز إلى قسمين ، تعليمات القفز غير المشروط وتعليمات القفز الشرطي.

- **Unconditional Jump Instructions:** Transfers the program sequence to the described memory address.
- **Conditional Jump Instructions:** Transfers the program sequences to the described memory address only if the condition is satisfied.

2. Call Instructions

The call instruction transfers the program sequence to the memory address given in the operand. Before transferring, the address of the next instruction after CALL is pushed onto the stack. Call instructions are also two types: Unconditional Call Instructions and Conditional Call Instructions.

تعليمات الاستدعاء تنقل تسلسل البرنامج إلى عنوان الذاكرة المعطى في المعامل. قبل النقل ، يتم دفع عنوان الإرشاد التالي بعد CALL على المكس. تعليمات الاتصال هي أيضًا نوعان: تعليمات الاتصال غير المشروطة وإرشادات الاتصال المشروطة

- **Unconditional Call Instructions:** It transfers the program sequence to the memory address given in the operand.
- **Conditional Call Instructions:** Only if the condition is satisfied, the instructions execute.

3. Return Instructions

The return instruction transfers the program sequence from the subroutine to the calling program. Return instructions are two types: Unconditional Jump Instructions and Conditional Jump Instructions.

تنقل تعليمات الإرجاع تسلسل البرنامج من الروتين الفرعي إلى برنامج الاستدعاء. تعليمات الإرجاع
نوعان: تعليمات القفز غير المشروط وتعليمات القفز الشرطي.