Machine Instruction Characteristics

يتم تحديد تشغيل المعالج من خلال التعليمات التي يقوم بتنفيذها والتي يشار اليها باسم machine instructions او computer instructions . ويشار الى مجموعة التعليمات المختلفة التي يمكن للمعالج تنفيذها باسم instruction set .

Elements of a Machine Instruction

يجب ان تحتوي كل تعليمات على المعلومات التي يطلبها المعالج للتنفيذ.

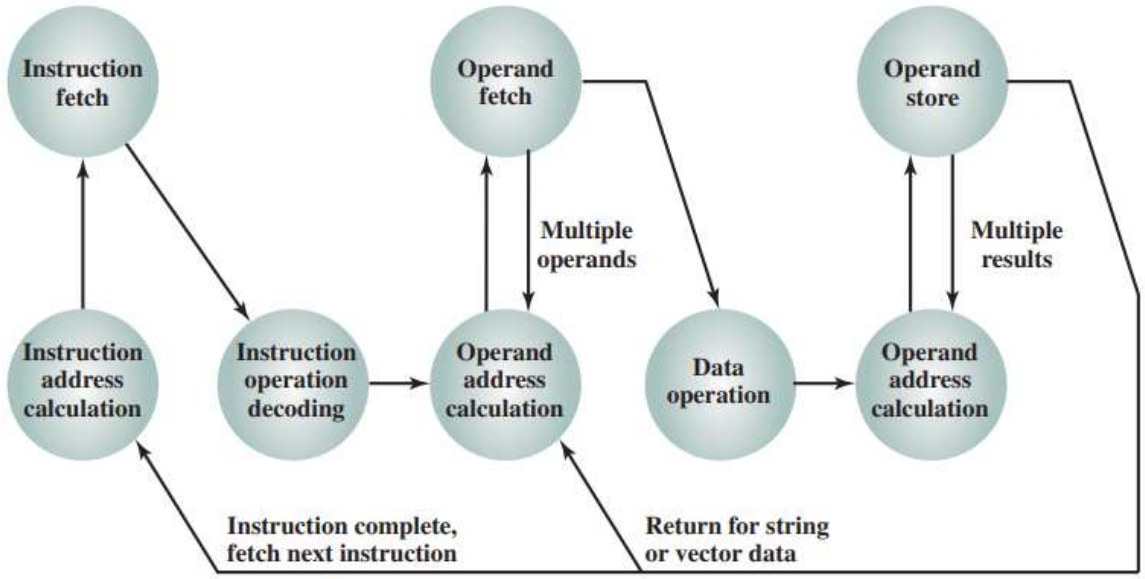


Figure 12.1 Instruction Cycle State Diagram

هاذي الصورة توضح الخطوات المتبعة في تنفيذ التعليمات ويحدد معها الـ machine instruction منها :

- Operation code : رمز العملية يحدد العملية (Operation) التي سيتم تنفيذها على سبيل المثال (ADD,I/O) يتم تحديد العملية بواسطة الـ Binary Code الذي يعرف بـ operation code او opcode .
- Source operand reference : مرجع معامل المصدر قد تشتمل العملية على معامل واحد او اكثر المعاملات (operands) هي التي تمثل input او مدخلات للعملية (operation).
- Result operand reference : مرجع معامل النتيجة قد تنتج العملية نتيجة.
- Next instruction reference : مرجع التعليمات التالية يخبر المعالج بمكان جلب التعليمات التالية بعد اكتمال تنفيذ هذه التعليمة.

يمكن ان يكون عنوان التعليمات التالية التي سيتم جلبها اما عنوانا حقيقا (real address) او عنونا افتراضيا (virtual address) هذا يعتمد على البنية. بشكل عام يكون التمييز واضحا بالنسبة لبنية مجموعة التعليمات (instruction set architecture). في معظم الحالات تتبع التعليمات التالية التي سيتم جلبها مباشرة بعد التعليمات الحالية() . في هذة الحالات لا توجد اشارة واضحة الى التعليمات التالية (next instruction). عند الحاجة الى مرجع صريح (explicit reference) يجب توفير الذاكرة الرئيسية (main memory address) او (Virtual Memory Address) .

يمكن ان تكون معاملات المصدر او النتيجة (Source and result operands) في واحدة من اربع مناطق :

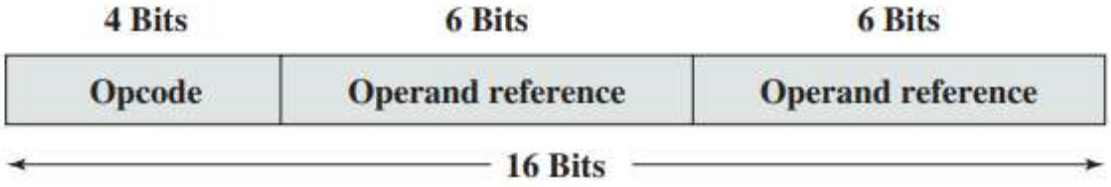
- Main or virtual memory : في الذاكرة الرئيسية او الذاكرة الافتراضية مثل الحال مع next instruction references يجب ان توفير addresses في main & virtual memory .
- Processor register : سجل المعالج مع استثناءات نادرة جدا يحتوي المعالج على سجل واحد او اكثر يمكن referenced اليه بواسطة تعليمات الجهاز (machine instructions) اذا كان يوجد سجل واحد فقط تكون الاشارة اليه ضمنية (implicit) اما

ان كان اكثر فسيتم تعين اسم او رقم لكل سجل ويجب ان تحتوي التعليمات على رقم السجل المطلوب او الاسم.

- Immediate : المباشر قيمة المعامل موجود في حقل في التعليمات التي يتم تنفيذها.
- I/O device : جهاز الادخال والاخراج يجب ان تحدد التعليمات وحدة الادخال والاخراج والجهاز الخاص بالتشغيل.

Instruction Representation

داخل الكمبيوتر يتم تمثيل كل تعليمات من خلال سلسلة من البتات. يتم تقسيم التعليمات الى حقول تتوافق مع العناصر المكونة للتعليمات. مثال على ذالك من خلال صورة :



في معظم الـ instruction sets يتم استخدام اكثر من تنسيق واحد اثناء تنفيذ التعليمات تتم قراءة التعليمات من سجل instruction register (IR) في المعالج.

يجب ان يكون المعالج قادرا على استخراج البيانات من حقول التعليمات المختلفة لاجراء العمليات المطلوبة. من الصعب على المبرمج والقارئ للـ textbooks للتعامل مع هذا التمثيلات الثنائية (binary representations) لتعليمات الالة. لذالك هكذا اصبح استخدام التمثيل الرمزي لتعليمات الالة. مثالا على ذالك :

Table Q1(a): IAS instruction set

Opcode		Assembly	RTL Format
Bin	Hex		
0000 1010	0A	LOAD MQ	AC <= MQ
0000 1001	09	LOAD MQ,M(X)	MQ <= M(X)
0010 0001	21	STOR M(X)	M(X) <= AC
0000 0001	01	LOAD M(X)	AC <= M(X)
0000 0010	02	LOAD -M(X)	AC <= -M(X)
0000 0011	03	LOAD |M(X)|	AC <= |M(X)|
0000 0100	04	LOAD -|M(X)|	AC <= -|M(X)|
0000 1101	0D	JUMP M(X,L)	PC <= M(X,L)
0000 1110	0E	JUMP M(X,R)	PC <= M(X,R)
0000 1111	0F	JUMP+ M(X,L)	AC ≥0? PC <= M(X,L)
0001 0000	10	JUMP+ M(X,R)	AC ≥0? PC <= M(X,R)
0000 0101	05	ADD M(X)	AC <= AC + M(X)
0000 0111	07	ADD |M(X)|	AC <= AC + |M(X)|
0000 0110	06	SUB M(X)	AC <= AC - M(X)
0000 1000	08	SUB |M(X)|	AC <= AC- |M(X)|
0000 1011	0B	MUL M(X)	AC <= msb(M(X)*MQ) MQ <= lsb(M(X)*MQ)
0000 1100	0C	DIV M(X)	MQ <= quo(AC÷M(X)) AC <= rem(AC÷M(X))
0001 0100	14	LSH	AC <= (AC <<0)
0001 0101	15	RSH	AC <= (0 >> AC)
0001 0010	12	STOR M(X,Laddr)	M(X,Laddr) <= AC(28:39)
0001 0011	13	STOR M(X,Raddr)	M(X,Raddr) <= AC(28:39)

الـ Opcodes يتم تمثيلها بالاختصارات التي تسمى mnemonics وهي التي تشير الى العملية.

مثال على ذالك :

ADD	Add
SUB	Subtract
MUL	Multiply
DIV	Divide
LOAD	Load data from memory
STOR	Store data to memory

والمعاملات الـ (Operands) ايضا يتم تمثيلها على شكل رمزي على سبيل المثال :

ADD R,Y

قد يعني اضافة القيمة في موقع البيانات Y الى محتويات سجل R . في هذا المثال يشير الى Y عنوان موقع في الذاكرة ويشير الى R عنوان موقع في سجل معين.

لاحظ ان الـ operation تتم على محتويات موقع ما وليس على address .

يحتوي كل symbolic opcode كود تشغيل رمزي على تمثيل ثنائي ثابت (fixed binary representation) ويحدد المبرمج موقع كل معامل رمزي (symbolic operand) على سبيل المثال :

X = 500

Y = 600

و simple program سقبل هذا الادخال الرمزي ويحول اكواد التشغيل ومراجع المعاملات الى نموذج ثنائي ويبني التعليمات الالة الثنائية.

Instruction Types

مثال في high-level language instruction يمكن عنه التعبير على سبيل المثال بلغة C .

X = X + Y

الـ statement هاذي تدل على اضافة قيمة Y الى قيمة المخزنة في X و وضع النتيجة في X.

كيف يمكن تحقيق ذالك بـ machine instructions لنفترض ان

X في موقع بيانات 700 وY في موقع بيانات 701

يمكن لهاذي العملية ان تتم بثلاثة تعليمات :

1. قم بعمل تحميل (Load) لـ Register بمحتويات موقع 700.
2. وقم باضافة محتويات موقع 701 الى السجل.
3. وقم بتخزين المحتويات السجل في موقع الذاكرة 700.

مثل ما هو واضح تعبر اللغة عالية المستوى عن العمليات بشكل (concise algebraic) باستخدام المتغيرات.

تعبر لغة الالة عن العمليات بشكل اساسي من movement of data او من registers .

في هذا المثال البسيط دعونا نفكر في انواع التعليمات التي يجب ان تتضمن في الكمبيوتر العملي. الكمبيوتر يجب ان يحتوي على مجموعة من التعليمات التي يسمح للمستخدم بصياغة اي مهمة لمعالجة البيانات. اي برنامج مكتوب بـ high-level language يجب ترجمته الى لغة الالة ليتم تنفيذه. وبالتالي فان مجموعة تعليمات لالة يجب ان تكون كافية للتعبير عن اي من التعليمات من لغة عالية المستوى. يمكننا تصنيف انواع التعليمات هكذا :

1. Data processing : معالجة البيانات مثل التعليمات الحسابية والمنطقية.
2. Data storage : تخزين التعليمات نقل البيانات من او الى السجل و او موقع الذاكرة.
3. Data movement : مثال i/o instruction .
4. Control : مثل الـ test و branch instructions .

توفر التعليمات الحسابية (Arithmetic instructions) قدرات حسابية لمعالجة البيانات الرقمية. التعليمات المنطقية (Logic (Boolean) instructions) تعمل على bits من الـ word كبتات وليس ارقام. وبتالي فهي توفر امكانات لمعالجة اي نوع اخر من انواع البيانات قد يرغب المستخدم في توظيفها يتم تنفيذ هذه العمليات بشكل اساسي على البيانات الموجودة في سجلات المعالج. ولذالك يجب ان تكون هناك تعليمات الذاكرة لنقل البيانات بين الذاكرة والسجلات. وهناك حاجة الى I/O instructions لنقل البرامج والبيانات الى الذاكرة وارجاع نتائج الحسابات الى المستخدم. الـ Test instructions تستخدم لاختبار قيمة data word او

حالة حساب status of a computation . والـ Branch instructions تستخدم للتفرع الى مجموعة مختلفة من التعليمات اعتمادا على قرار متخذ في سياق البرنامج.

Number of Addresses

احدى الطرق التقليدية لوصف بنية المعالج هي من حيث عدد العناوين الموجودة في كل التعليمات. اصبح هذا البعد اقل اهمية مع التعقيد المتزايد لتصميم المعالج. لكن مفيد.

ما هو العدد الاقصى لعدد العناوين التي قد يحتاجها في التعليمات ؟ معظم التعليمات الحسابية والمنطقية تطلب operands . التعليمات الحسابية اما تكون اما احادية الـ operand او اكثر. لكن على الاقل نحتاج الى عناونين كحد اقصى للاشارة الى source operands . يجب تخزين نتيجة العملية مما يقترح ايضا عنوان ثالث يحدد destination operand وفي النهاية يجب جلب التعليمة (next instruction) التالية وعنوانها مطلوب. يشير هذا المنطق الى انه من الممكن ان تكون التعليمة المطلوبة تحتوي على اربع مراجع للعناوين : two source operands و one destination operand و address of the next instruction .

في معظم البنيات تحتوي العديد من التعليمات على واحد او اثنان او ثلاثة عناوين مع كون عنوان التعليمة التالية ايضا (يحصل عليها من الـ program counter). تحتوي معظم البنيات ايضا على عدد قليل من التعليمات ذات الاغراض الخاصة مع المزيد من المعاملات.

Instruction Set Design

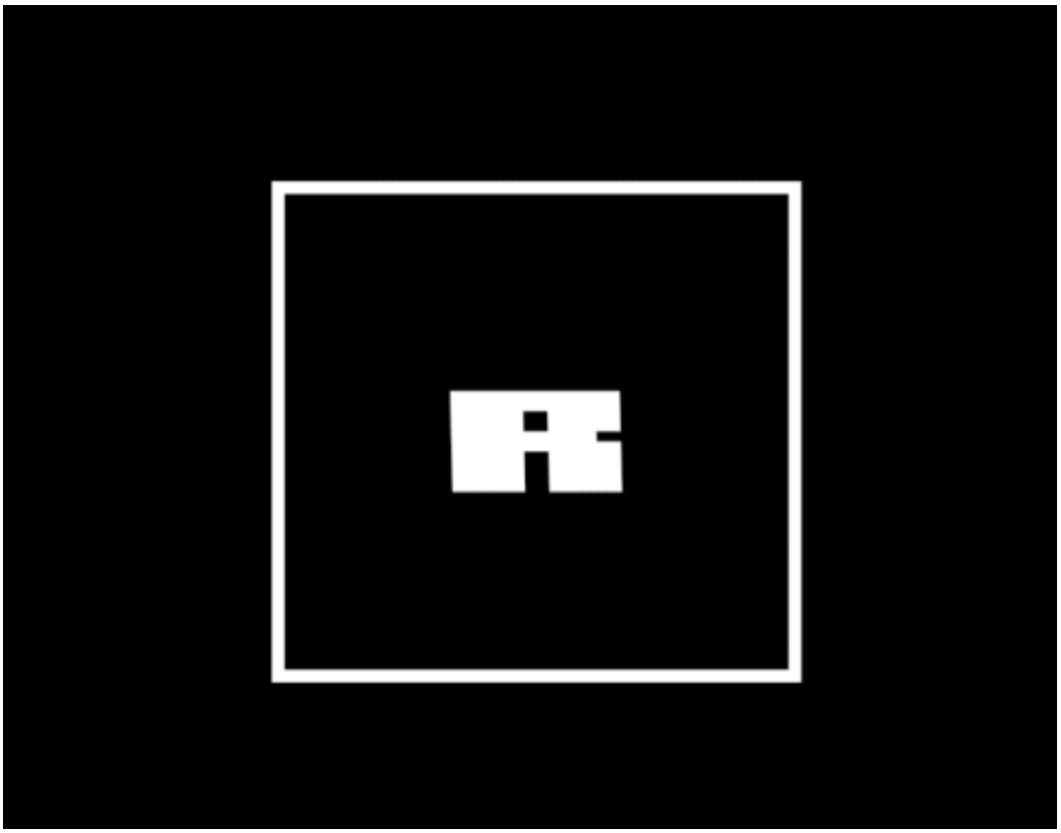
احد اكثر الجوانب المثيرة للاهتمام والاكثر تحليلا في تصميم الكمبيوتر هو تصميم instruction set . يعد تصميم مجموعة من التعليمات معقدا للغاية لانه يؤثر على العديد من

جوانب نظام الكمبيوتر . تحدد مجموعة التعليمات العديد من الوظائف التي يؤديها المعالج وبالتالي يكون لها تاثير كبير على تنفيذ المعالج. الـ instruction set هي مجموعة من التعليمات وهي وسيلة للمبرمج للتحكم في المعالج. وبتالي يجب مراعاة متطلبات المبرمج عند تصميم مجموعة التعليمات. قد يفاجئك معرفة ان بعض القضايا الاساسية المتعلقة بتصميم مجموعات التعليمات لا تزال محل نزاع. اهم قضايا التصميم الاسياسية هي :

- Operation repertoire : كم عدد العمليات التي يجب ان تتوفر ومدى تعقيدها.
- Data types : الانواع المختلفة من البيانات التي يتم تنفيذ العمليات عليها.
- Instruction format : طول التعليمات بالبت وعدد العناوين وحجم حقول المختلفة الخ.
- Registers : عدد سجلات المعالج التي يمكن referenced اليها بالتعليمات واستخدامها.
- Addressing : الـ mode او modes التي يتم من خلالها تحديد operand .

هذه القضايا مترابطة الى حد كبير ويجب اخذها في الاعتبار معا عند تصميم مجموعة من التعليمات.

AhmadAlFareed



Twitter : https://twitter.com/dr_retkit

YouTube : https://www.youtube.com/@retkit1823