

第零章 计算机概论

一、电脑：辅助人脑的工具

所谓的电脑就是一种计算机，而计算机其实是：“接受使用者输入指令与数据，经由中央处理器的数学与逻辑单元运算处理后，以产生或储存成有用的信息”。因此，只要有输入设备（不管是键盘还是触摸屏）及输出设备（例如电脑屏幕或直接由打印机打印出来），让你可以输入数据使该机器产生信息的，那就是一部计算机了。

那么计算机主要的组成元件是什么呢？下面我们以常见的个人电脑主机或服务器工作站主机 来作为说明好了

1.1 计算机硬件的五大单元

关于电脑的硬件组成部分，其实可以观察你的台式机来分析一下，依外观来说这家伙主要 可分为三部分，分别是：

- 输入单元：包括键盘、鼠标、读卡机、扫描仪、手写板、触摸屏等等一堆；
- 主机部分：这个就是系统单元，被主机机箱保护住了，里面含有一堆板子、CPU与内存 等；
- 输出单元：例如屏幕、打印机等等

我们主要通过输入设备如鼠标与键盘来将一些数据输入到主机里面，然后再由主机的功能处 理成为图表或文章等信息后，将结果传输到输出设备，如屏幕或打印机上面。

那主机里面含 有什么元件呢？如果你曾经拆开过电脑主机机箱，会发现其实主机里面最重要的就是一块主板，上面安插了**中央处理器（CPU）以及内存、硬盘（或记忆卡）**还有一些适配卡设备而已。

整部主机的重点在于**中央处理器（Central Processing Unit, CPU）**，CPU为一个具有特定功能的芯片，**里头含有微指令集**，如果你想要让主机进行什么特异的功能，就得要参考这颗 CPU 是否有相关内置的微指令集才可以。**由于CPU的工作主要在于管理与运算，因此在 CPU内又可分为两个主要的单元，分别是：算数逻辑单元与控制单元。**

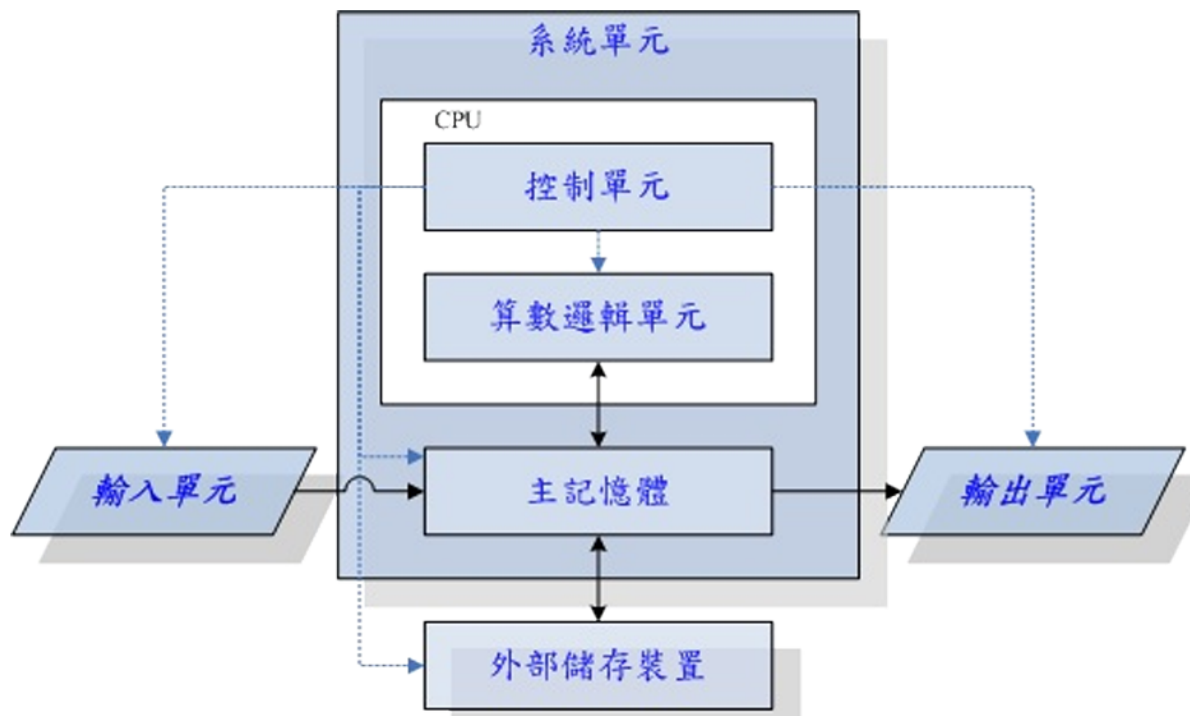
其中算数逻辑单元主要负责程序运算与逻辑判断，控制单元则主要在协调各周边元件与各单元间的工作。

既然CPU的重点是在进行运算与判断，那么要被运算与判断的数据是从哪里来的？

CPU读取的数据都是从内存来的！内存内的数据则是从输入单元所传输进来！而CPU处理完毕的数据也必须要先写回内存中，最后数据才从内存传输到输出单元。

为什么我们都会说，要加快系统性能，通常将内存容量加大就可以获得相当好的成效？如同下图以及上面的说明，因为所有的数据都要经过内存的传输，所以内存的容量如果太 小数据高速缓存就不足～影响性能相当大啊！尤其针对Linux作为服务器的环境下！这点要特别记忆喔！

综合上面所说的，我们会知道其实电脑是由几个单元所组成的，包括**输入单元、输出单元、CPU内部的控制单元、算数逻辑单元与内存五大部分**。这几个东西的相关性如下所示：



上面图示中的“系统单元”其实指的就是电脑机箱内的主要元件，而重点在于CPU与内存。特别要看的是实线部分的传输方向，基本上数据都是流经过内存再转出去的！至于数据会流进/流出内存则是CPU所发布的控制命令！而CPU实际要处理的数据则完全来自于内存（不管是程序还是一般文件数据）！这是个很重要的概念喔！这也是为什么当你的内存不足时，系统的性能就很糟糕！也是为什么现在人们买智能手机时，对于可用内存的要求都很高的原因

1.2 CPU的架构

前面说过的，CPU其实内部已经含有一些微指令，我们所使用的软件都要经过CPU内部的微指令集来达成才行。那这些指令集的设计主要又被分为两种设计理念，这就是目前世界上常见到的两种主要CPU架构，分别是：精简指令集（RISC）与复杂指令集（CISC）系统

- 精简指令集（RISC）

这种CPU的设计中，微指令集较为精简，每个指令的执行时间都很短，完成的动作也很单纯，指令的执行性能较佳；但是若要做复杂的事情，就要由多个指令来完成。

常见的RISC微指令集CPU主要例如：甲骨文（Oracle）公司的 SPARC系列、IBM公司的Power Architecture（包括 PowerPC）系列、与安谋公司（ARM Holdings）的ARM CPU系列

在应用方面，SPARC CPU 的电脑常用于学术领域的大型工作站中，包括银行金融体系的主要服务器也都有这类的电脑架构；至于PowerPC架构的应用上，例如索尼（Sony）公司出产的Play Station 3（PS3）就是使用PowerPC架构的Cell处理器；那安谋的ARM呢？你常使用的各厂牌手机、PDA、导航系统、网络设备（交换器、路由器等）等，几乎都是使用ARM架构的CPU喔！老实说，目前世界上使用范围最广的CPU可能就是ARM这种架构的

- 复杂指令集（CISC）

与RISC不同的，CISC在微指令集的每个小指令可以执行一些较低阶的硬件操作，指令数目多而且复杂，每条指令的长度并不相同。因为指令执行较为复杂所以每条指令花费的时间较长，但每个个别指令可以处理的工作较为丰富。

常见的CISC微指令集CPU主要有AMD、Intel、VIA等的x86架构的CPU。由于AMD、Intel、VIA所开发出来的x86架构CPU被大量使用于个人电脑（Personal computer）用途上面，因此，个人电脑常被称为x86架构的电脑！

那为何称为x86架构呢？这是因为最早的那颗Intel发展出来的CPU代号为8086，后来依此架构又开发出80286, 80386..., 因此这种架构的CPU就被称为x86架构了。

在2003年以前由Intel所开发的x86架构CPU由8位升级到16、32位，后来AMD依此架构修改新一代的CPU为64位，为了区别两者的差异，**因此64位的个人电脑CPU又被统称为x86_64的架构喔！**

所谓的位指的是CPU一次数据读取的最大量！64位CPU代表CPU一次可以读写64bits这么多的数据，32位CPU则是CPU一次只能读取32位的意思。因为CPU读取数据量有限制，因此能够从内存中读写的数据也就有所限制。所以，一般32位的CPU所能读写的最大数据量，大概就是4GB左右

那么不同的x86架构的CPU有什么差异呢？除了CPU的整体结构（如第二层高速缓存、每次运行可执行的指令数等）之外，**主要是在于微指令集的不同**。新的x86的CPU大多含有很先进的微指令集，这些微指令集可以加速多媒体程序的运行，也能够加强虚拟化的性能，而且某些微指令集更能够增加能源效率，让CPU耗电量降低呢！由于电费越来越高，购买电脑时，除了整体的性能之外，节能省电的CPU特色也可以考虑喔！

1.3 其他单元的设备

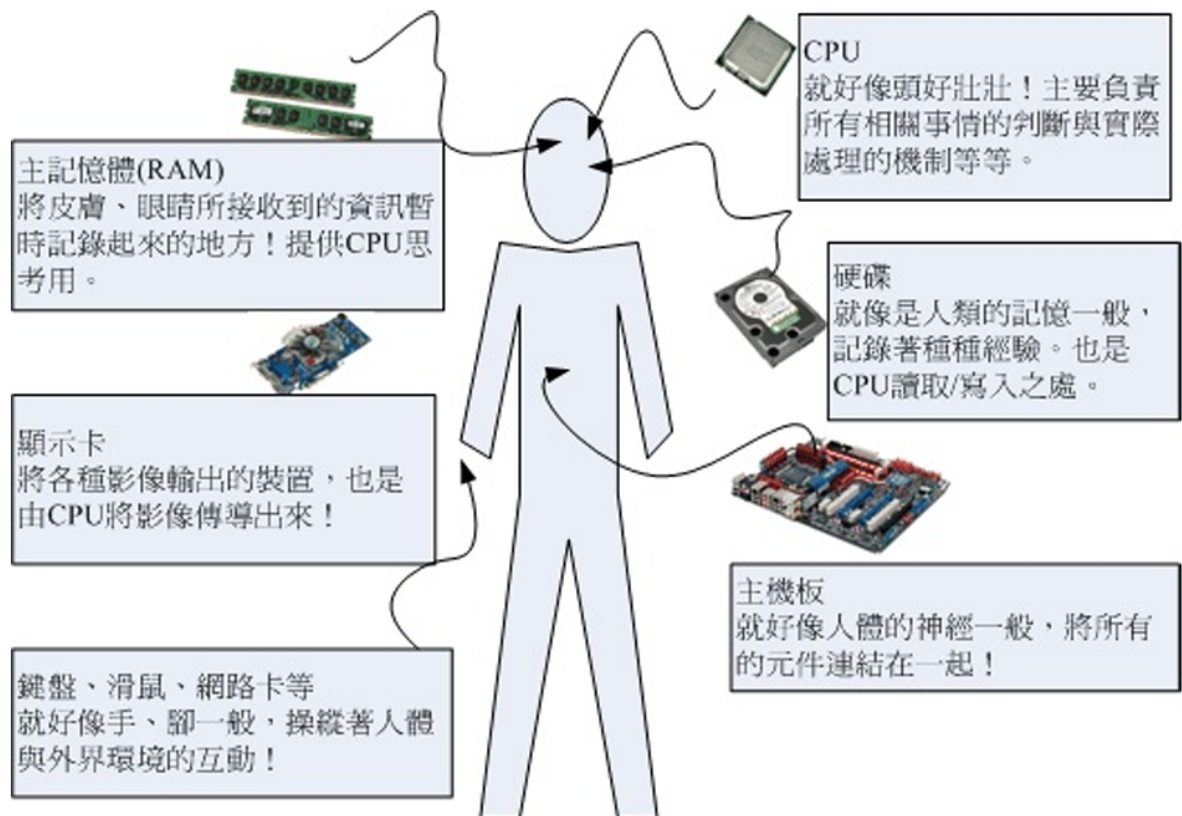
五大单元中最重要的控制、算术逻辑被整合到CPU的封装中，但系统当然不可能只有CPU啊！那其他三个重要电脑单元的设备还有哪些呢？

其实在主机机箱内的设备大多是通过**主板**（main board）连接在一块，主板上有个链接沟通所有设备的芯片组，这个芯片组可以将所有单元的设备链接起来，好让CPU可以对这些设备下达命令。其他单元的重要设备主要有：

- 系统单元：如图 0.1.2 所示，系统单元包括 CPU 与内存及主板相关元件。而主板上头其实还有很多的连接接口与相关的适配卡，包括近期常使用的 PCI-E 10G 网卡、磁盘阵列卡、还有显卡等等。尤其是显卡，这东西对于玩3D游戏来说是非常重要的一环，他与显示的精致度、色彩与分辨率都有关系
- 存储单元：包括内存（main memory, RAM）与辅助内存，其中辅助内存其实就是大家常听到的“储存设备”啰！包括硬盘、软盘、光盘、磁带等等的
- 输入、输出单元：同时涵盖输入输出的设备最常见的大概就是触摸屏了。至于单纯的输入设备包括前面提到的键盘鼠标之外，目前的体感设备也是重要的输入设备喔！至于输出设备方面，除了屏幕外，打印机、音效喇叭、HDMI电视、投影机、蓝牙耳机等等，都算喔

电脑是如何运行的呢？

1.4 运行流程



- CPU=脑袋瓜子：每个人会做的事情都不一样（微指令集的差异），但主要都是通过脑袋瓜子来进行判断与控制身体各部分的活动；
- 内存=脑袋中放置正在被思考的数据的区块：在实际活动过程中，我们的脑袋瓜子需要有外界刺激的数据（例如光线、环境、语言等）来分析，那这些互动数据暂时存放的地方就是内存，主要是用来提供给脑袋瓜子判断用的信息。
- 硬盘=脑袋中放置回忆的记忆区块：跟刚刚的内存不同，内存是提供脑袋目前要思考与处理的信息，但是有些生活琐事或其他没有要立刻处理的事情，就当成回忆先放置到脑袋的记忆深处吧！那就是硬盘！主要目的是将重要的数据记录起来，以便未来将这些重要的经验再次的使用；
- 主板=神经系统：好像人类的神经一样，将所有重要的元件连接起来，包括手脚的活动都是脑袋瓜子发布命令后，通过神经（主板）传导给手脚来进行活动啊！
- 各项周边设备=人体与外界沟通的手、脚、皮肤、眼睛等：就好像手脚一般，是人体与外界互动的重要关键！
- 显卡=脑袋中的影像：将来自眼睛的刺激转成影像后在脑袋中呈现，所以显卡所产生的数据来源也是CPU控制的。
- 电源供应器（Power）=心脏：所有的元件要能运行得要有足够的电力供给才行！这电力供给就好像心脏一样，如果心脏不够力，那么全身也就无法动弹的！心脏不稳定呢？那你的身体当然可能断断续续的～不稳定！

由这样的关系图当中，我们知道整个活动中最重要的就是脑袋瓜子！而脑袋瓜子当中与现在正在进行的工作有关的就是CPU与内存！任何外界的接触都必须要有脑袋瓜子中的内存记录下来，然后给脑袋中的CPU依据这些数据进行判断后，再发布命令给各个周边设备！如果需要用到过去的经验，就得由过去的经验（硬盘）当中读取啰！

也就是说，整个人体最重要的地方就是脑袋瓜子，同样的，**整部主机当中最重要的就是CPU与内存**，而CPU的数据来源通通来自于内存，如果要由过去的经验来判断事情时，也要将经验（硬盘）挪到目前的记忆（内存）当中，再交由CPU来判断喔！这点得要再次的强调啊！

1.5 电脑上面常用的计算单位（容量、速度等）

电脑的运算能力除了CPU微指令集设计的优劣之外，但主要还是由速度来决定的。至于存放在电脑储存设备当中的数据容量也是有单位的

- 容量单位

电脑对数据的判断主要依据有没有通电来记录信息，所以理论上对于每一个纪录单位而言，它只认识0与1而已。**0/1这个二进制的单位我们称为bit**。但bit实在太小了，所以在储存数据时每份简单的数据都会使用到8个bits的大小来记录，因此定义出**Byte** 字节这个单位，他们的关系为：

1 Byte = 8 bits

不过同样的，Byte还是太小了，在较大的容量情况下，使用Byte相当不容易判断数据的大小，举例来说，1000000Bytes这样的显示方式你能够看得出有几个零吗？所以后来就有一些常见的简化单位表达式，例如K代表1024Byte，M代表1024K等。而这些单位在不同的进位制下有不同的数值表示，下面就列出常见的单位与进位制对应：

进位制	Kilo	Mega	Giga	Tera	Peta	Exa	Zetta
二进制	1024	1024K	1024M	1024G	1024T	1024P	1024E
十进制	1000	1000K	1000M	1000G	1000T	1000P	1000E

一般来说，文件大小使用的是二进制的方式，所以 1GBytes 的文件大小实际上为：
1024x1024x1024Bytes 这么大！速度单位则常使用十进制，例如 1GHz 就是
1000x1000x1000Hz 的意思

- 速度单位

CPU的运算速度常使用MHz或者是GHz之类的单位，**这个Hz其实就是秒分之一**。而在网络传输方面，由于网络使用的是bit为单位，因此网络常使用的单位为Mbps是**Mbits per second**，亦即是**每秒多少 Mbit**。举例来说，大家常听到的 20M/5M 传输速度，如果转成文件大小的Byte时，其实理论最大传输值为：每秒2.5MByte/每秒625KByte的下载/上传 速度喔

例题：假设你今天购买了500GB的硬盘一颗，但是格式化完毕后却只剩下460GB左右的容量，这是什么原因？答：因为一般硬盘制造商会使用十进制的单位，所以500GByte代表为50010001000*1000Byte之意。转成文件的容量单位时使用二进制（1024为底），所以就成为466GB左右的容量了。

二、数据表示方式

事实上我们的电脑只认识0与1，记录的数据也是只能记录0与1而已，所以电脑常用的数据是二进制的。但是我们人类常用的数值运算是十进制，文字方面则有非常多的语言。

那么电脑如何记录与显示这些数值/文字呢？就得要通过一系列的转换才可以啦！下面我们就来谈谈数值与文字的编码系统

2.1 数字系统

早期的电脑使用的是利用通电与否的特性的真空管，如果通电就是1，没有通电就是0，后来沿用至今，我们称这种只有0/1的环境为二进制。所谓的十进制指的是逢十进一位，因此在个位数归为零而十位数写成1。所以所谓的二进制，就是逢二就前进一位的意思

那二进制怎么用呢？我们先以十进制来解释好了。如果以十进制来说，3456的意义为：

$$3456 = 3 \times 10^3 + 4 \times 10^2 + 5 \times 10^1 + 6 \times 10^0$$

同样的，将这个原理带入二进制的环境中，我们来解释一下1101010的数值转为十进制的话，结果如下：

$$1101010 = 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 64 + 32 + 0 \times 16 + 8 + 0 \times 4 + 2 + 0 \times 1 = 106$$

那如果有十进制数值转为二进制的环境时，该如何计算？刚刚是乘法，现在则是除法就对了！我们同样的使用十进制的106转成二进制来测试一下好了：

2	106	0	106/2的餘數
2	53	1	53/2的餘數
2	26	0	26/2的餘數
2	13	1	13/2的餘數
2	6	0	6/2的餘數
2	3	1	3/2的餘數
	1		

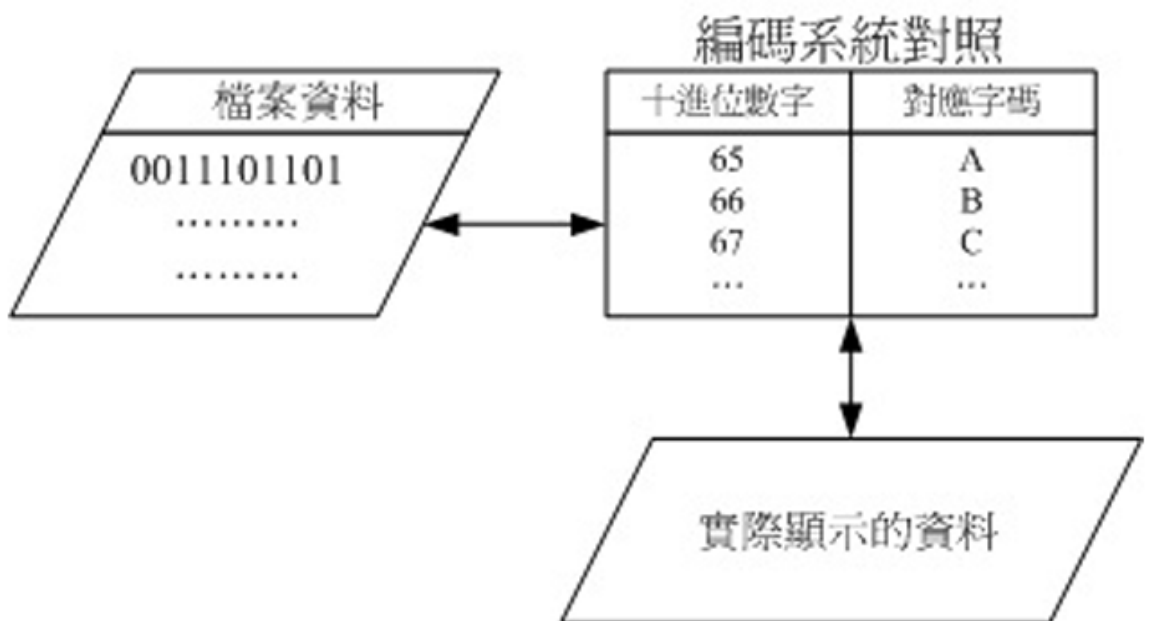
最后的写法就如同上面的红色箭头，由最后的数字向上写，因此可得到1101010的数字啰！这些数字的转换系统是非常重要的，因为电脑的加减乘除都是使用这些机制来处理的！

2.2 文字编码系统

既然电脑都只有记录0/1而已，甚至记录的数据都是使用Byte/bit等单位来记录的，那么文字该如何记录啊？

事实上文字文件也是被记录为0与1而已，而这个文件的内容要被取出来查阅时，必须要经过一个编码系统的处理才行。

所谓的“编码系统”可以想成是一个“字码对照表”，他的概念有点像下面的图示：



当我们要写入文件的文字数据时，该文字数据会由编码对照表将该文字转成数字后，再存入文件当中。

同样的，当我们要将文件内容的数据读出时，也会经过编码对照表将该数字转成对应的文字后，再显示到屏幕上。

现在你知道为何浏览器上面如果编码写错时，会出现乱码了吗？这是因为编码对照表写错，导致对照的文字产生误差之故啦！

常用的英文编码表为ASCII系统，这个编码系统中，每个符号（英文、数字或符号等）都会占用1Bytes的记录，因此总共会有 $2^8=256$ 种变化。

至于中文字当中的编码系统早期最常用的就是big5这个编码表了。每个中文字会占用2Bytes，理论上最多可以有 $2^{16}=65536$ ，亦即最多可达6万多个中文字。但是因为big5编码系统并非将所有的位都拿来运用成为对照，所以并非可达这么多的中文字码的。目前big5仅定义了一万三千多个中文字，很多中文利用big5是无法成功显示的

big5码的中文字编码对于某些数据库系统来说是有问题的，某些字码例如“许、盖、功”等字，由于这几个字的内部编码会被误判为单/双引号，在写入还不成问题，在读出数据的对照表时，常常就会变成乱码。

不只中文字，其他非英语系国家也常常会有这样的问题出现啊！为了解决这个问题，由国际组织ISO/IEC跳出来制订了所谓的**Unicode编码系统**，我们常常称呼的**UTF8或万国码的编码**就是这个。因为这个编码系统打破了所有国家的不同编码，因此目前网际网络社会大多朝向这个编码系统在走。

三、软件程序运行

一般来说，目前的电脑系统将软件分为两大类，一个是系统软件，一个是应用程序。

我们还是得要了解一下什么是程序，尤其是机器程序，了解了之后再探讨一下为什么现今的电脑系统需要“操作系统”这玩意儿呢！

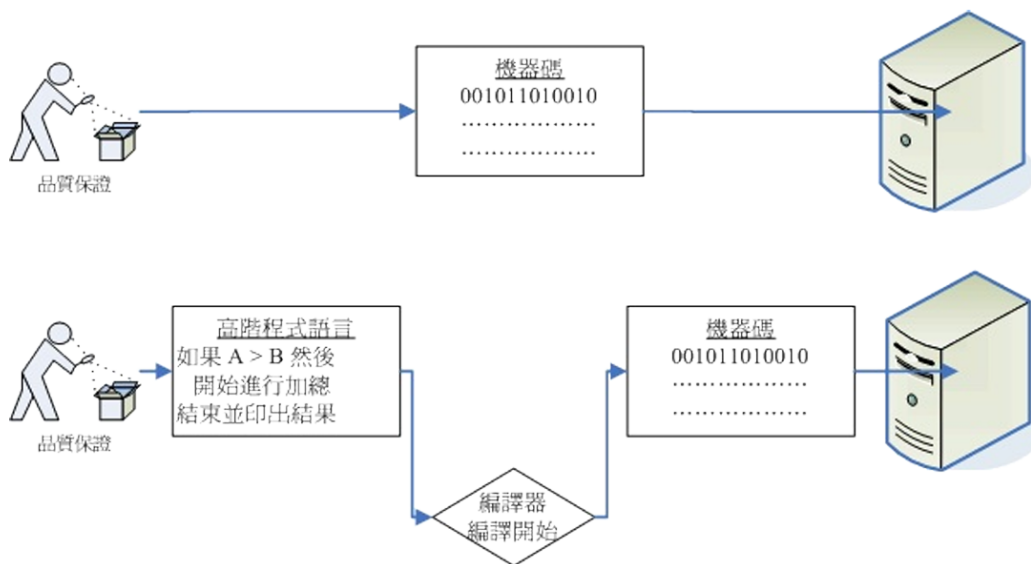
3.1 机器程序与编译程序

我们前面谈到电脑只认识0与1而已，而且电脑最重要的运算与逻辑判断是在CPU内部，而CPU其实是具有微指令集的。因此，我们需要CPU帮忙工作时，就得要参考微指令集的内容，然后撰写让CPU读的懂的指令码给CPU执行，这样就能够让CPU运行了。

不过这样的流程有几个很麻烦的地方，包括：

- 需要了解机器语言：机器只认识0与1，因此你必须要学习直接写给机器看的语言！这个地方相当的难呢！
- 需要了解所有硬件的相关功能函数：因为你的程序必须要写给机器看，当然你就得要参考机器本身的功能，然后针对该功能去撰写程序码。例如，你要让DVD影片能够放映，那就得要参考DVD光驱的硬件信息才行。万一你的系统有比较冷门的硬件，光是参考技术手册可能会昏倒~
- 程序不具有可携性：每个CPU都有独特的微指令集，同样的，每个硬件都有其功能函数。因此，你为A电脑写的程序，理论上是没办法在B电脑上面运行的！而且程序码的修改非常困难！因为机器码，并不是人类看得懂程序语言啊！
- 程序具有专一性：因为这样的程序必须要针对硬件功能函数来撰写，如果已经开发了一支浏览器程序，想要再开发文件管理程序时，还是得从头再参考硬件的功能函数来继续撰写，每天都在和“硬件”挑战！

那怎么解决啊？为了解决这个问题，电脑科学家设计出一种让人类看得懂程序语言，然后创造一种“编译器”来将这些人类能够写的程序语言转译成为机器能看得懂机器码，如此一来我们修改与撰写程序就变得容易多了！目前常见的编译器有C，C++，Java，Fortran等等。机器语言与高阶程序语言的差别如下所示



图

3.1 编译器的角色

从上面的图示我们可以看到高阶程序语言的程序码是很容易察看的！上面已经将程序码（英文）写成中文说这样比较好理解！

所以这样已经将程序的修改问题处理完毕了。问题是，在这样的环境下面我们还是得要考虑整体的硬件系统来设计程序！

举例来说，当你需要将运行的数据写入内存中，你就得要自行分配一个内存区块出来让自己的数据能够填上去，所以你还得要了解到内存的位址是如何定位的，为了要克服硬件方面老是需要重复撰写控制码的问题，所以就有操作系统（Operating System, OS）的出现了！什么是操作系统呢？下面就来谈一谈先！

3.2 操作系统

如同前面提到的，在早期想要让电脑执行程序就得要参考一堆硬件功能函数，并且学习机器语言才能够撰写程序。同时每次写程序时都必须重新改写，因为硬件与软件功能不见得都一致之故。那如果我能够把所有的硬件都驱动，并且提供一个发展软件的参考接口来给工程师开发软件的话，那发展软件不就变的非常的简单了？那就是操作系统啦！

- 操作系统核心（Kernel）

操作系统（Operating System, OS）其实也是一组程序，这组程序的重点在于**管理电脑的所有活动以及驱动系统中的所有硬件**。电脑没有软件只是一堆废铁，那么操作系统的功能就是让CPU可以开始判断逻辑与运算数值、让内存可以开始载入/读出数据与程序码、让硬盘可以开始被存取、让网卡可以开始传输数据、让所有周边可以开始运行等等。总之，硬件的所有动作都必须要通过这个操作系统来达成就是了。

上述的功能就是操作系统的核心（Kernel）了！你的电脑能不能做到某些事情，都与核心有关！只有核心有提供的功能，你的电脑系统才能帮你完成！

举例来说，你的核心并不支持 TCP/IP 的网络协定，那么无论你购买了什么样的网卡，这个核心都无法提供网络能力的！

但是单有核心我们使用者也不知道能作啥事的～因为核心主要在管控硬件与提供相关的能力（例如存取硬盘、网络功能、CPU资源取得等），这些管理的动作是非常的重要的，如果使用者能够直接使用到核心的话，万一使用者不小心将核心程序停止或破坏，将会导致整个系统的崩溃！**因此核心程序所放置到内存当中的区块是受保护的！并且开机后就一直常驻在内存当中**

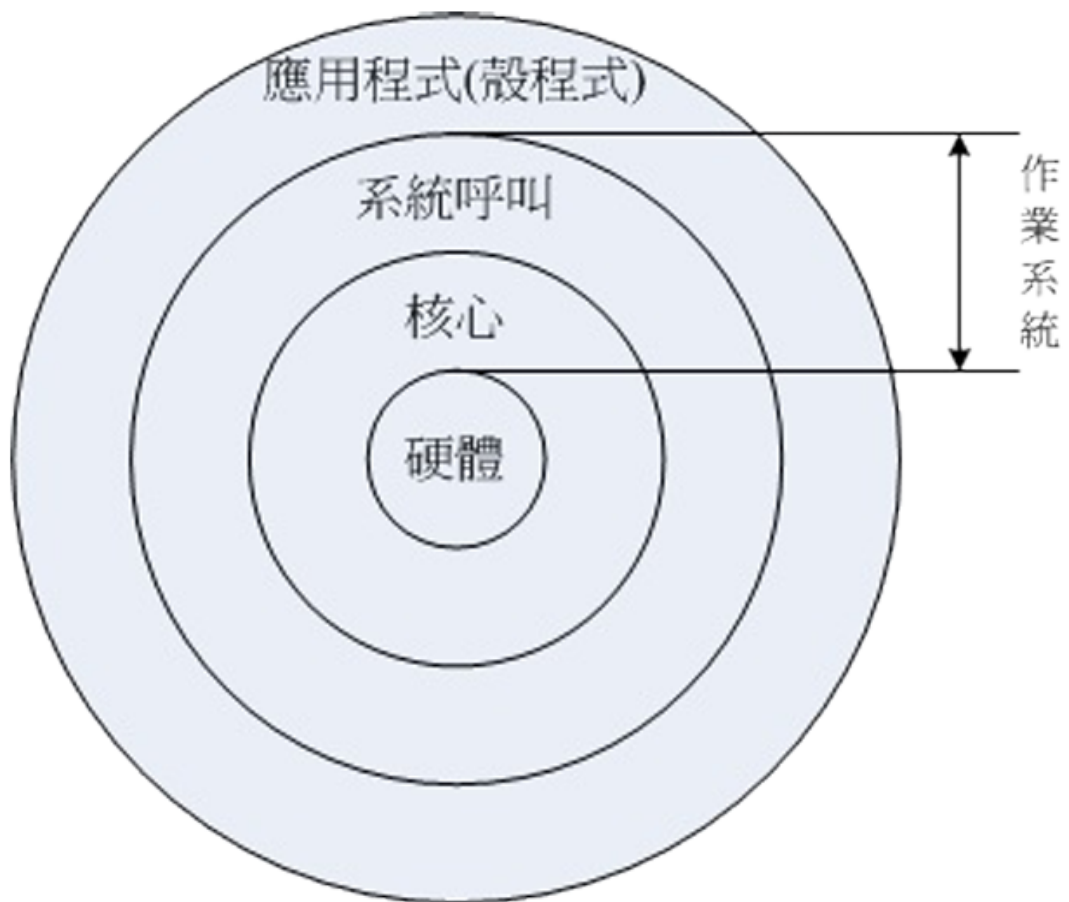
所以整部系统只有核心的话，我们就只能看着已经准备好运行（Ready）的电脑系统，但无法操作他！好像有点望梅止渴的那种感觉啦！这个时候就需要软件的帮忙了！

- 系统调用

既然我的硬件都是由核心管理，那么如果我想要开发软件的话，自然就得要去参考这个核心的相关功能！如此一来不是从原本的参考硬件函数变成参考核心功能，还是很麻烦啊！

有没有更简单的方法啊！为了解决这个问题，**操作系统通常会提供一整组的开发接口给工程师来开发软件！**

工程师只要遵守该开发接口那就很容易开发软件了！举例来说，我们学习C程序语言只要参考C程序语言的函数即可，不需要再去考虑其他核心的相关功能，因为**核心的系统调用接口会主动的将C程序语言的相关语法转成核心可以了解的任务函数**，那核心自然就能够顺利运行该程序了！如果我们将整个电脑系统的相关软/硬件绘制成图的话，他的关系有点像这样



电脑系统主要由硬件构成，然后核心程序主要在管理硬件，提供合理的电脑系统资源分配（包括CPU资源、内存使用资源等等），因此只要硬件不同（如x86架构与RISC架构的CPU），核心就得要进行修改才行。而由于核心只会进行电脑系统的资源分配，所以在上头还需要有应用程序的提供，使用者才能够操作系统

为了保护核心，并且让程序设计师比较容易开发软件，因此**操作系统除了核心程序之外，通常还会提供一整组开发接口，那就是系统调用层**。软件开发工程师只要遵循公认的系统调用参数来开发软件，该软件就能够在该核心上运行。所以你可以发现，软件与核心有比较大的关系，与硬件关系则不大！硬件也与核心有比较大的关系！至于与使用者有关的，那就是应用程序啦

简单的说，上面的图示可以带给我们下面的概念：

- 操作系统的核心层直接参考硬件规格写成，所以同一个操作系统程序不能够在不一样的硬件架构下运行。举例来说，个人电脑版的Windows 8.1不能直接在ARM架构（手机与平板硬件）的电脑下运行。

- 操作系统只是在管理整个硬件资源，包括CPU、内存、输入输出设备及文件系统文件。如果没有其他的应用程序辅助，操作系统只能让电脑主机准备妥当（Ready）而已！并无法运行其他功能。所以你现在知道为何Windows上面要达成网页影像的运行还需要类似 Photoshop之类的软件安装了吧？
- 应用程序的开发都是参考操作系统提供的开发接口，所以该应用程序只能在该操作系统上面运行而已，不可以在其他操作系统上面运行的。现在您知道为何去购买线上游戏的光盘时，光盘上面会明明白白的写着该软件适合用于哪一种操作系统上了吧？也该知道某些游戏为何不能够在Linux上面安装了吧？

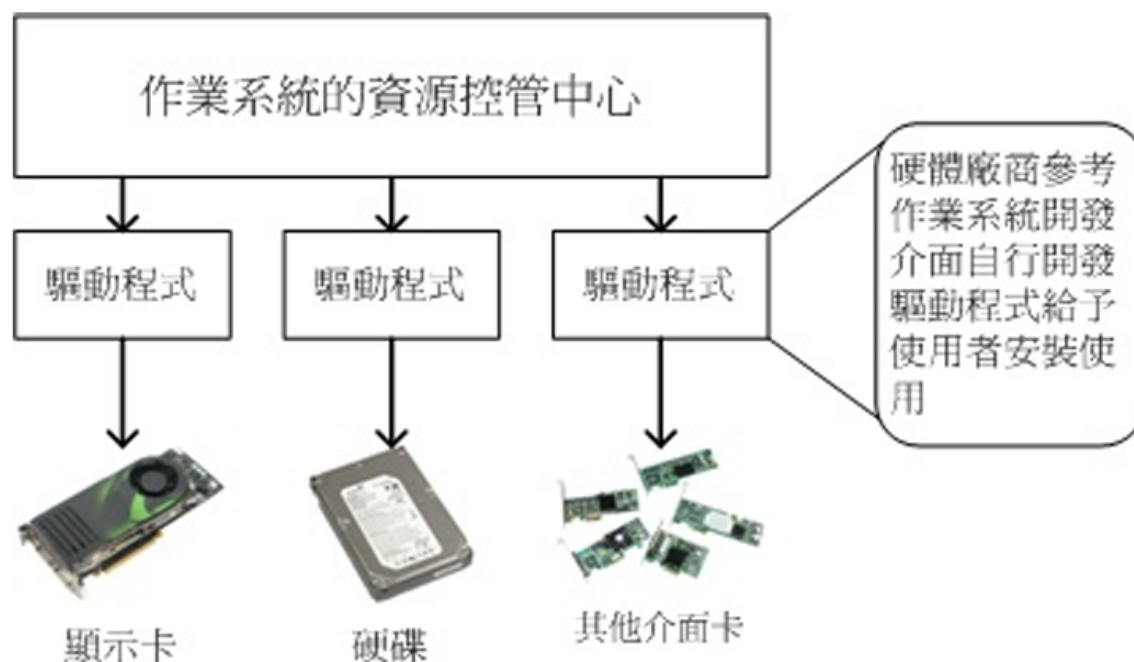
核心功能

既然核心主要是在负责整个电脑系统相关的资源分配与管理，那我们知道其实整部电脑系统最重要的就是CPU与内存，因此，核心至少也要有这些功能的：

- 系统调用接口（System call interface）刚刚谈过了，这是为了方便程序开发者可以轻易的通过与核心的沟通，将硬件的资源进一步的利用，于是需要有这么个简易的接口来方便程序开发者
- 程序管理（Process control）总有听过所谓的“多任务环境”吧？一部电脑可能同时间有很多的工作跑到CPU等待运算处理，核心这个时候必须要能够控制这些工作，让CPU的资源作有效的分配才行！另外，良好的CPU调度机制（就是CPU先运行那个工作的排列顺序）将会有效的加快整体系统性能呢。
- 内存管理（Memory management）控制整个系统的内存管理，这个内存控制是非常重要的，因为系统所有的程序码与数据都必须要先存放在内存当中。通常核心会提供虚拟内存的功能，当内存不足时可以提供内存交换（swap）的功能
- 文件系统管理（Filesystem management）文件系统的管理，例如数据的输入输出（I/O）等等的工作啦！还有不同文件格式的支持啦等等，如果你的核心不认识某个文件系统，那么您将无法使用该文件格式的文件喽！例如：Windows 98就不认识NTFS文件格式的硬盘；
- 设备的驱动（Device drivers）就如同上面提到的，硬件的管理是核心的主要工作之一，当然喽，设备的驱动程序就是核心需要做的事情啦！好在目前都有所谓的“可载入模块”功能，可以将驱动程序编辑成模块，就不需要重新编译核心啦！
- 事实上，驱动程序的提供应该是硬件厂商的事情！硬件厂商要推出硬件时，应该要自行参考操作系统的驱动程序开发接口，开发完后将该驱动程序连同硬件一同贩卖给使用者才对！举例来说，当你购买显卡时，显卡包装盒都会附上一片光盘，让你可以在进入Windows之后进行驱动程序的安装啊

操作系统与驱动程序

老实说，驱动程序可以说是操作系统里面相当重要的一环了！不过，硬件可是持续在进步当中的！包括主板、显卡、硬盘等等。那么比较晚推出的较新的硬件，例如显卡，我们的操作系统当然就不认识喽！那操作系统该如何驱动这块新的显卡？为了克服这个问题，操作系统通常会提供一个开发接口给硬件开发商，让他们可以根据这个接口设计可以驱动他们硬件的“驱动程序”，如此一来，只要使用者安装驱动程序后，自然就可以在他们的操作系统上面驱动这块显卡了。



由上图我们可以得到几个小重点：

- 操作系统必须要能够驱动硬件，如此应用程序才能够使用该硬件功能；
- 一般来说，操作系统会提供开发接口，让开发商制作他们的驱动程序；
- 要使用新硬件功能，必须要安装厂商提供的驱动程序才行；
- 驱动程序是由厂商提供的，与操作系统开发者无关。

所以，如果你想要在某个操作系统上面安装一张新的显卡，那么请要求该硬件厂商提供适当的驱动程序吧！为什么要强调“适当的驱动程序”呢？因为驱动程序仍然是依据操作系统而开发的，所以，给Windows用的驱动程序当然不能使用于Linux的环境下了

3.3 应用程序

应用程序是参考操作系统提供的开发接口所开发出来软件，这些软件可以让使用者操作，以达到某些电脑的功能利用。举例来说，办公室软件（Office）主要是用来让使用者办公用的；影像处理软件主要是让使用者用来处理影音数据的；浏览器软件主要是让使用者用来上网浏览用的等等。

需要注意的是，应用程序是与操作系统有关系的，如同上面的图示当中的说明喔。因此，如果你想要购买新软件，请务必参考软件上面的说明，看看该软件是否能够支持你的操作系统啊！举例来说，如果你想要购买线上游戏光盘，务必参考一下该光盘是否支持你的操作系统，例如是否支持Windows XP/Windows Vista/MAC/Linux 等等。不要购买了才发现该软件无法安装在你的操作系统上喔

我们拿常见的微软公司的产品来说明。你知道Windows 8.1, Office 2013之间的关系了吗？

- Windows 8.1 是一套操作系统，他必须先安装到个人电脑上面，否则电脑无法开机运行；
- Windows 7 与Windows 8.1 是两套不同的操作系统，所以能在Win 7上安装的软件不见得可在Win 8.1上安装；
- Windows 8.1安装好后，就只能拥有很少的功能，并没有办公室软件；
- Office 2013是一套应用程序，要安装前必须要了解他能在哪些操作系统上面运行。

五、重点回顾

- 计算机的定义为：“接受使用者输入指令与数据，经由中央处理器的数学与逻辑单元运算处理后，以产生或储存成有用的信息”；

- 电脑的五大单元包括：输入单元、输出单元、控制单元、算数逻辑单元、存储单元五大部分。其中CPU占有控制、算术逻辑单元，存储单元又包含内存与辅助内存；
- 数据会流进/流出内存是CPU所发布的控制命令，而CPU实际要处理的数据则完全来自于内存；
- CPU依设计理念主要分为：精简指令集（RISC）与复杂指令集（CISC）系统；
- 关于CPU的频率部分：外频指的是CPU与外部元件进行数据传输时的速度，倍频则是CPU内部用来加速工作性能的一个倍数，两者相乘才是CPU的频率速度；
- 新的CPU设计中，已经将北桥的内存控制芯片整合到CPU内，而CPU与内存、显卡沟通的总线通常称为系统总线。南桥就是所谓的输入输出（I/O）总线，主要在联系硬盘、USB、网卡等周边设备；
- CPU每次能够处理的数据量称为字组大小（word size），字组大小依据CPU的设计而有32位与64位。我们现在所称的电脑是32或64位主要是依据这个CPU解析的字组大小而来的！
- 个人电脑的内存主要元件为动态随机存取内存（DynamicRandomAccessMemory, DRAM），至于CPU内部的第二层高速缓存则使用静态随机存取内存（StaticRandomAccessMemory, SRAM）；
- BIOS（Basic Input Output System）是一套程序，这套程序是写死到主板上面的一个内存芯片中，这个内存芯片在没有通电时也能够将数据记录下来，那就是只读存储器（Read Only Memory, ROM）；
- 目前主流的外接卡接口大多为PCIe接口，且最新为PCIe3.0，单信道速度高达1GBytes/s常见的显卡连接到屏幕的接口有HDMI/DVI/D-Sub/Display port等等。HDMI可同时传送影像与声音。
- 传统硬盘的组成为：圆形盘片、机械手臂、磁头与主轴马达所组成的，其中盘片的组成为扇区、磁道与柱面；
- 磁盘连接到主板的接口大多为SATA或SAS，目前台式机主流为SATA 3.0，理论极速可达600MBytes/s。
- 常见的文字编码为ASCII，繁体中文编码主要有Big5及UTF8两种，目前主流为UTF8操作系统（Operating System, OS）其实也是一组程序，这组程序的重点在于管理电脑的所有活动以及驱动系统中的所有硬件。
- 电脑主要以二进制作为单位，常用的磁盘容量单位为Bytes，其单位换算为1Byte= 8bits。最纯净的操作系统仅在驱动与管理硬件，而要使用硬件时，就得需要通过应用软件或者是壳程序（shell）的功能，来调用操作系统操纵硬件工作。目前称为操作系统的，除了上述功能外，通常已经包含了日常工作所需要的应用软件在内了。

六、本章习题