

How computer hardware and software work together



**ASSIGN
BUSTER**

How does computer software and hardware work together? This is kind of a complex question. I think the most important thing connecting software and hardware would probably be the register. Commands or services requested by the software are encoded into binary values of 0's and 1's and are written into a register.

For example, some register may hold a value of "01101100". This encoding has a total of 8 bits. Each value of either 0 or 1 is called a bit, and 8 bits are called a byte. At the hardware level, the computer hardware does not see these 0's and 1's as numbers, but as voltage levels.

For example, a high voltage of 3 volts may represent a 1 and a low voltage of near 0 volts may represent a 0. These high and low voltages are fed into a series of logic gates which then, through the correct logic design, will go about with producing the results needed by that particular software command. For example, if the software is requesting a calculation, the hardware logic gates may add the first two bits of the encoding together, produce a result of 1 ($0+1 = 1$), and then write this result into another register somewhere for the software to go fetch and read. Or, if the encoding is a request for some kind of service, the logic gates may in turn access or write into some other registers which would trigger other logic gates to initiate this service.

But again, this is a very complex process with many variations depending on what kind of technologies the designers of the system choose to use. What is a motherboard you ask? The motherboard is a printed circuit board (PCB) that contains and controls the components that are responsible for

processing data. The motherboard contains the CPU, memory, and basic controllers for the system. Motherboards are often sold with a CPU.

The motherboard has a Real-time clock (RTC), ROM BIOS, CMOS RAM, RAM sockets, bus slots for attaching devices to a bus, CPU socket or slot, cache RAM slot or sockets, jumpers, keyboard controller, interrupts, internal connectors, and external connectors. The bus architecture and type of components on it determine a computer's performance. The motherboard with its ribbon cables, power supply, CPU, and RAM is designated as a "bare bones" system. The motherboard contains a system's clock to synchronize the operation of the bus and other components. Jumpers on the motherboard allow a user to set different clock rates to work with the CPU.

Other jumpers control other components on the motherboard. 386 motherboards have an extra slot to add a math coprocessor (ALU – arithmetic and logic unit). 486's and Pentiums have a math coprocessor (ALU) already built into the chip. Operating systems offer a number of services to application programs and users. Applications access these services through application programming interfaces.

or system calls. By invoking these interfaces, the application can request a service from the operating system, pass parameters, and receive the results of the operation. Users may also interact with the operating system with some kind of a software user interface like typing commands by using command line interface or using a graphical user interface. For hand-held and desktop computers, the user interface is generally considered part of the operating system. On large multi-user systems like Unix and Unix-like

systems, the user interface is generally implemented as an application program that runs outside the operating system.