The human computer
Debriefing: Control Unit

How realistic was this?

Not so bad.

One important thing to remember is that the main role of the control unit is to interpret the instructions stored in memory and ensure that the program runs smoothly. In particular, it decides where the next instruction is by updating the value of the program counter.

Another important thing to realize is that the control unit has no way to know whether what it does is correct or not. It doesn't need (and doesn't know how to) decode the numbers it passes from and to other parts of the computer. If the ALU suddenly decides that 1+1=3, the control unit couldn't care less. It will simply transfer this binary data to wherever it should go. More importantly, the error is nearly impossible to detect and recover from, since nothing resembles a binary value more than another binary value. If the JMP now links to a wrong cell, or the ADD tries to add an instruction to a number, so be it.

About speed: clearly, the speed of the control unit is critical to the speed of the overall computer. If it takes one minute to decode each instruction, the computer will be very slow. The time it takes the control unit to process all steps in one of its cycles (i.e. to read, decode and fully process one instruction) is called the CPU's clock rate. It is measured in *cycles per second* (measured in hertz). For instance, a 3GHz Pentium processor can process 3,000,000,000 cycles per second (how fast can YOU go ?). However, in practice, it is often the other parts of a computer that slow down the control unit. The processor may run very quickly through the steps of its cycles, but it could hand forever waiting for a very sloppy memory to return data when it tries to access it. Most time will be spent waiting for memory to respond rather than actually processing data. Especially limiting in nowadays computers is the speed of the cables (or "bus") that connect the control unit to the memory.

Of course, in reality, a control unit is more complex.

In the memory we role-played here, instructions were stored as a series of letters (M… O… V). In reality, instructions are not stored as words, but as a unique instruction code (the "Op-code"), which is defined by the constructor of the computer's CPU (e.g. there are Intel-defined codes for each instruction that can be run on an Intel Pentium processor).

Control systems in advanced computers may change the order of some instructions so as to improve performance.

It is noticeable that the sequence of operations that the control unit goes through to process an instruction is in itself like a short computer program - and indeed, in some more complex CPU designs, there is another yet smaller computer called a micro-sequencer that runs a microcode program that causes all of these events to happen.

While a computer may be viewed as running one gigantic program stored in its main memory, in some systems it is necessary to give the appearance of running several programs simultaneously (“multi-tasking”). This is achieved by having the computer switch rapidly between running each program in turn. One means by which this is done is with a special signal called an interrupt which can periodically cause the computer to stop executing instructions where it was and do something else instead. By remembering where it was executing prior to the interrupt, the computer can return to that task later. If several programs are running "at the same time", the interrupt generator might be causing several hundred interrupts per second, causing a program-switch each time. Since modern computers typically execute instructions several orders of magnitude faster than human perception, it may appear that many programs are running at the same time even though only one is ever executing in any given instant. This method of multitasking is sometimes termed "time-sharing" since each program is allocated a "slice" of time in turn.

Remember:

CPU (central processing unit) = control unity + arithmetic and logic unit (ALU) + registers + basic input and output devices

Most basic computer= CPU + memory

A CPU is usually constructed on a single integrated circuit called a micro processor (e.g. Intel Pentium III). CPU and memory are usually incorporated with other devices on an electronic circuit called the motherboard.

For more information, you can have a look at the role-sheets and debriefing information of the parts played by other people in the class.