I/O Interface

I/O devices such as keyboards and displays establish communication of computer with outside world.

Devices can be interfaced in two ways I/O MAPPED I/O and Memory mapped I/O. In I/O mapped I/O, device is identified with a unique device number and data are transferred thru IN/OUT instruction. Memory mapped I/O each device is identified with 16 bit address. I/O devices are considered to be a part of memoryand memory related instruction is used for data transfer.

An I/O interface must be able to

_ Determine whether or not it is being interfaced

_ Determine whether it has to send data to CPU or receive data from CPU

_ Send ready signal informing CPU that transfer is over

_ Send interrupt Requests to CPU and receive interrupt acknowledgement and send an interrupt type.

An Interface can be divided into two parts. A part that interfaces to the I/O device and a part that interfaces to the system bus. There must be drivers and receivers to maintain signal quality, logic for translating the interface control signals to proper handshaking signals, logic for decoding address that appear on the bus.

Handshaking signals are used to determine in which direction transfer has to take place whether from CPU or to CPU. It should determine whether it is a READ or WRITE operation. Interrupt signals also must be handled here.

Address decoder determine whether it is I/O mapped I/O or Memory mapped I/O from one of the bits. If the decoder finds that an interface is referenced it sends signal to the appropriate device.

Interfaces can be categorized according to the way I/O devices transfer data either in serial or parallel form.

Memory Interface

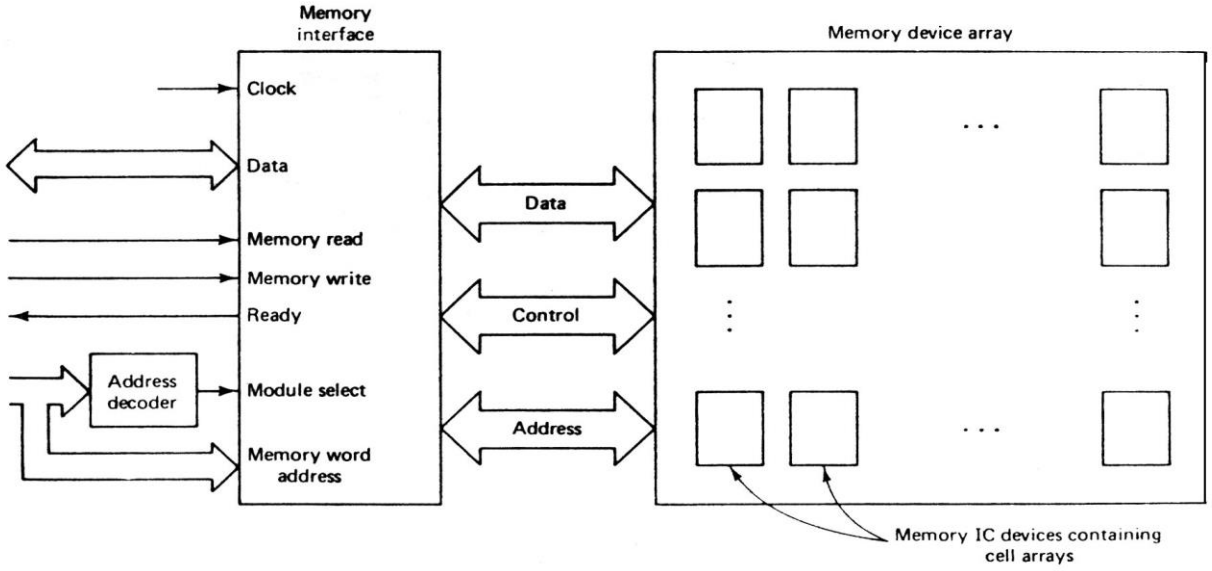


Figure 4.1: Memory Module Design

The memory of a computer consists of number of memory modules. Each module consists of an interface and an array of memory IC devices. Each IC device consists of an array of memory cells as shown in Figure 4.1. Each cell can store 1 bit.

Microprocessor communicates with memory through memory interface.

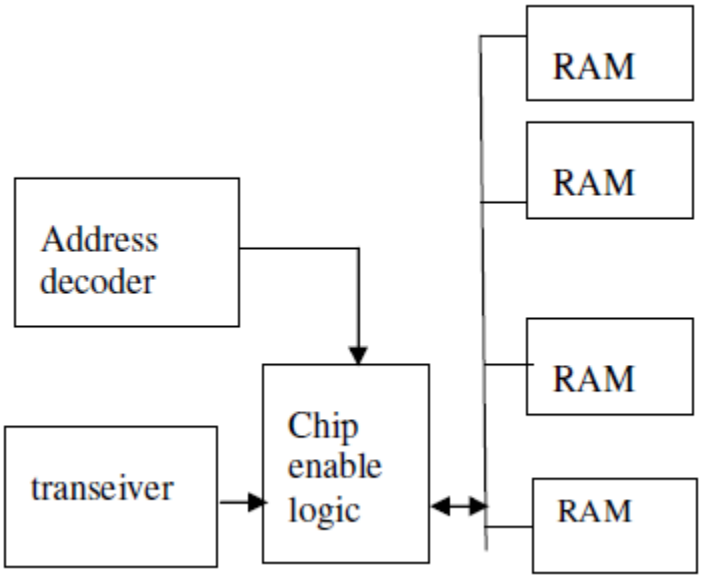
The primary function of memory interface is that the microprocessor must be able to read from or write to a given register of memory chip. The microprocessor must be able to select the memory chip, send control signals for read or write operation.

Certain signals to indicate whether a Memory read or write operation has to be performed.

Whenever a communication with memory is required, a set of signals has to be sent by CPU.

Chip select logic which is used to select the particular chip based on the signal it receives from the

Transceiver



Physical Memory Mapped I/0 and Port Addressed I/O

CPU controlled I/O comes in two ways. The difference is simply whether we use the normal memory addresses for I/O, this is referred to as physical memory mapped I/O, or whether we set up a totally separate section of memory addresses which can ONLY be used for I/O, called port addressed I/O or I/O mappedI/O. Thus the latter has two separate addressing areas for the CPU, one for normal (physical) memory, one
for I/O, while the former has only one which serves both purposes.

The other difference between these two techniques is that with memory mapped I/O the same processor instructions that are used for transferring data to and from the processor registers and memory, or for testing the content of a memory register etc, are used for the I/O operation, whereas port-addressed I/O has specific instructions (load and store) which can only be used for I/O operations, and have different mnemonics and codes. In the latter case data must always be transferred into a processor register, and in the latter instructions like ADD, SUB (subtract) and CMP (compare) can refer to the input/output registers.

The peripheral controller serves to present the data at the appropriate memory locations, memory mapped or port addressed as may be. It will also provide control registers to

determine whether data can be inputted or outputed, that is whether the external connections are for input or output of data, since they are often shared. The processor cannot read and write at the same location , it must do one or the other.

There will also be a 'status' register which have various 'flags' which signal to the processor that some data has been placed for reading. Some of these may also be observable as connections to the outside world, to show, for example, that the processor has supplied data for output. The registers are the same size as the

computer memory locations, usually eight-bit,-byte sized , so that they usually occupy one memory address each. The actual number of addresses depends on the complexity and functions provided by the peripheral controller

Memory Mapped I/O

Memory I/O devices are mapped into the system memory map along with RAM and ROM. To access a hardware device, simply read or write to those 'special' addresses using the normal memory access instructions. The advantage to this method is that every instruction which can access memory can be used to manipulate an I/O device. The disadvantage to this method is that the entire address bus must be fully decoded for every device. For example, a machine with a 32-bit address bus would require logic gates to resolve the state of all 32 address lines to properly decode the specific address of any device. This increases the cost of adding hardware to the machine.

Port Mapped I/O or I/O Mapped I/O

I/O devices are mapped into a separate address space. This is usually accomplished by having a different set of signal lines to indicate a memory access versus a port access. The advantage to this system is that less logic is needed to decode a discrete address and therefore less cost to add hardware devices to a machine.

Hand Shaking

Handshaking, or two-way handshaking, is one type of strobed operation. It typically involves two handshaking lines: an output line to indicate when the board is ready and an input line that indicates when the peripheral device is ready. Depending on the timing and the property of the handshaking lines, there are different handshaking protocols.