Technical Report

MINICOMPUTER ON A CHIP

by
Hash Patel

Hardware cost limits the use of minicomputers for dedicated applications to systems bearing a price tag of at least $10,000 or more. The economies offered by low cost, high performance, commercially available microprocessors have opened up entirely new fields of dedicated computer applications which may not have been feasible or cost effective.

However, presently available microprocessors do not fare well in cost performance trade off with hardware and software in areas currently dominated by minicomputers. The microprocessor based system needs extensive software development to enhance its capability and be a useful system design tool. Because of the lack of adequate software support, a microprocessor based system may require longer development time than minicomputer based systems. In addition, the need of specialized software, general utility programs, mathematical routines and executive system software can increase the cost and lengthen the system development time by at least an order of magnitude. In order to justify these higher development costs, the microprocessor based system must have a sufficiently large production volume.

Intersil has accepted the challenge of bridging the cost/performance gap of minicomputer vs. microprocessor by offering the performance of a true minicomputer in a unique, single chip, low cost microprocessor, the IM6100.

By emulating an existing minicomputer architecture, the IM6100 overcomes many of the problems associated with the current microprocessor technology. The IM6100 is a single chip, 12 bit, parallel microcomputer which is capable of executing the instruction set of Digital Equipment Corporation's PDP-8 minicomputer. The IM6100 is designed by using state of the art "silicon gate complementary MOS" technology which offers high packing density with good speed performance, noise immunity and extremely low power dissipation. The internal logic structure is fully static in nature and allows the clock to be stopped between instructions, cycles or minor cycles. In addition, it requires single +5 volt supply. All the signals are fully TTL compatible. The design has been optimized to minimize external packages, thus, reducing the overall system cost in comparison with other microprocessors that require multiple supply voltages and support chips.

The ability of complementary MOS technology to withstand larger variation of supply voltages and temperature range automatically qualifies the IM6100 for military applications, so far ignored by other microprocessor developers.

The functional block diagram of the IM6100 is shown in figure 1. It has six 12 bit registers, a programmable logic array (PLA) to generate control signals, an arithmetic and logic unit (ALU) and timing circuitry.

As shown in the block diagram, the IM6100 allows two separate pins to connect an external crystal, thus eliminating the need of expensive and cumbersome clock generators or clocking mechanisms. The internal circuitry is designed to operate at any speed between DC and 2MHz. The internal timing circuitry will divide the crystal frequency by two and provide the base timing of 500ns clock frequency with 50% duty cycle for system operation, with a 4MHz crystal.

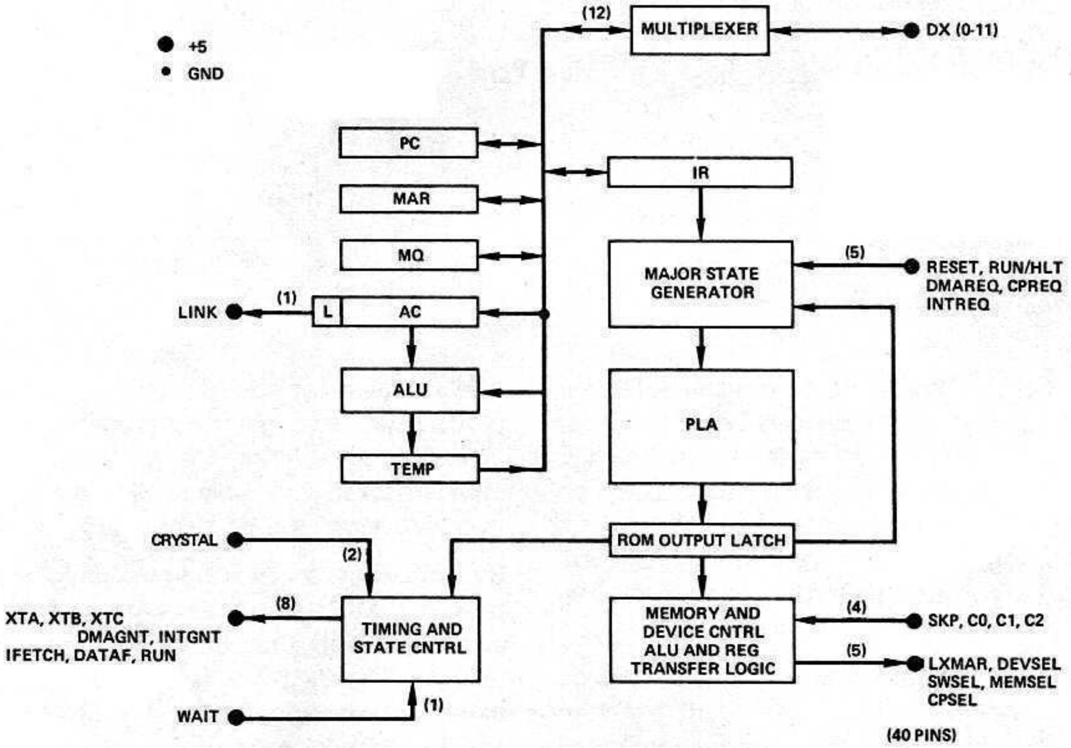

Figure 1. Functional Block Diagram

The 12 bit data bus handles address, data and instruction transfers between the CPU and the memory or the peripheral devices on a time multiplexed basis. All data transfers within the CPU are handled by the internal bidirectional bus. The accumulator (AC) is a 12 bit register in which arithmetic and logical operations are performed. All programmed information transfers between memory and peripheral devices are passed through the accumulator.

The program counter (PC) controls the program sequence. It contains the address of the memory location from which the next instruction is fetched. During the instruction fetch, contents of the program counter are transfered to the memory address register. The program counter is normally incremented by one after an instruction fetch, however, the contents of the program counter may be modified by branch instruction. The memory address register (MAR) contains the address of the memory location that is currently selected for reading or writing. The instruction register (IR) contains the instruction that is currently being executed by the CPU. The 12 bit temporary register (TEMP) latches the result of the arithmetic-logic unit before it is sent to the destination register to avoid race conditions. Link is used as a carry flip flop for 2's complement arithmetic; a carry out of accumulator complements the link. The link may be rotated as part of the accumulator and can be cleared, set, complemented, and tested under the program control. The multiplier quotient register (MQ) is a program accessible register for temporary data storage. MAR, IR and TEMP are also used as internal registers for microprogram control.

During an instruction fetch, the instruction to be executed is loaded into the instruction register. The PLA is used for correct sequencing of the CPU for the appropriate instruction. Once an instruction is executed, the major state generator scans the various request lines through an internal priority network. The state of the priority network decides whether the machine is going to fetch the next instruction in sequence or service one of the request lines.

Figure 2 describes the interface between CMOS memory and the IM6100. One may observe that the IM6100 is architectured to minimize the external circuitry to interface with memory and peripheral devices. All basic timing and control signals are produced by the processor. Slow memory and peripheral devices may be conveniently interfaced by using the WAIT signal.

The interrupt structure of the IM6100 is compatible with that of the PDP-8. The memory handling capacity of the IM6100 is also compatible with PDP-8. It can address up to 4K of basic memory directly. The memory capacity can be expanded up to 32K by external hardware. The IM6100 will require 5 microseconds to add two 12 bit words as compared to 3 microseconds for a PDP-8. The IM6100 has a 1 cycle direct memory access (DMA) similar to other microprocessors.

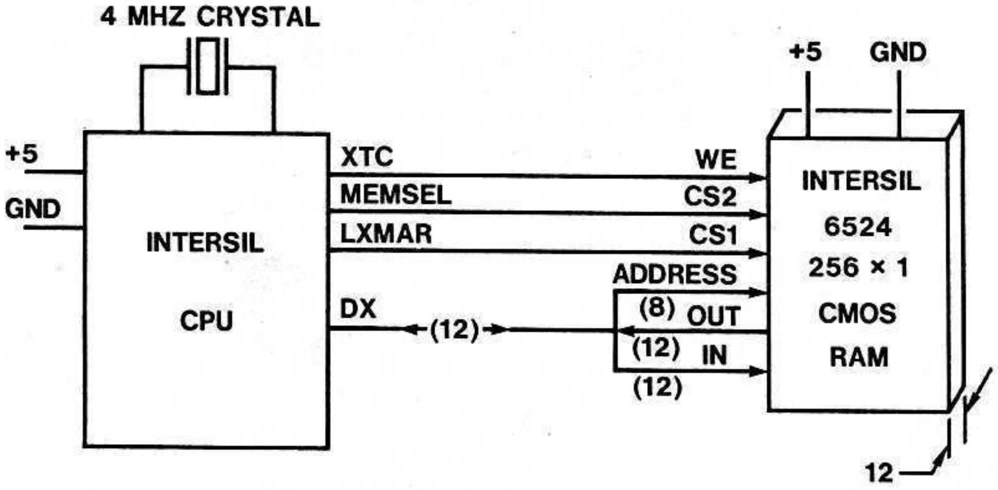

Figure 2. CPU-CMOS RAM Interface

As shown in figure 3, the IM6100 interfaces conveniently with PDP-8 compatible programmed I/O peripherals such as teletype, papertape reader, papertape punch, etc.

DEC's PDP-8 has perhaps the most well known machine organization and instruction set with more software support than any existing minicomputer system. A microcomputer architectured similar to the PDP-8 is immediately recognizable and immediately usable in most applications. Since system designers are already familiar with conventional minicomputer hardware, they can develop new systems around the IM6100 with a minimum of time and effort and still realize clean, economical and elegant solutions to digital design problems.

The microcomputer technology has evolved around 4/8 (single chip) or 16 (multi-chip) bit machines. The IM6100 is the first microprocessor incorporating a 12 bit CPU on one chip. The currently available 4/8 bit machines, though adequate for many applications as far as the word length is concerned, have awkward and time consuming memory reference instructions. There are many applications in which one must resort to double precision arithmetic in 8 bit machines to achieve the desired numerical accuracy. A 12 bit machine provides straightforward memory referencing and sufficient numerical accuracy without the correspondingly large memory overhead associated with the 16 bit architectures.

In summary, since the IM6100 utilizes the complementary MOS technology and recognizes the popular minicomputer instruction set, it will enable the innovative system designer to implement microcomputer based systems including battery operated and portable systems, economically and with considerably less design time.

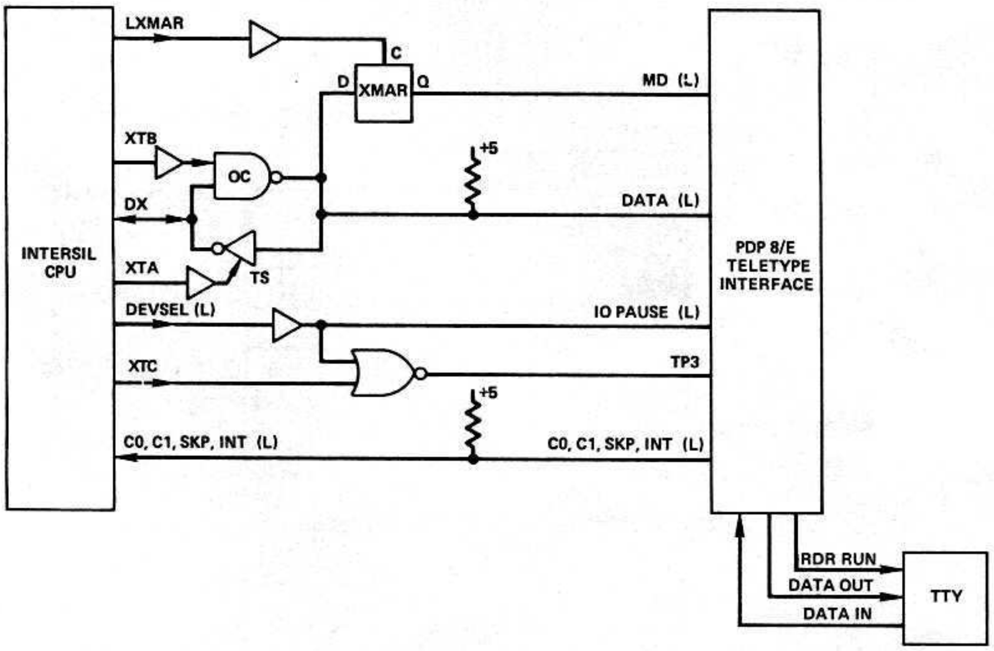

Figure 3. Example of a PDP 8/E Programmed I/O Peripheral Interface