THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 4, April 1979
Printed in U.S.A.

B.S.T.J. BRIEF

MAC-4: A Single-Chip Microcomputer

By W. F. CHOW and W. W. TROUTMAN

(Manuscript received January 30, 1979)

MAC-4 is a single-chip microcomputer designed to fill a rising need in many Bell System applications. Its instruction set promotes program memory efficiency and emphasizes bit manipulation. MAC-4 is implemented in CMOS technology, resulting in a low power dissipation which can be further reduced by using a special halt instruction. These features make it valuable as a controller in a telephone environment and for telephone loop power applications. Figure 1 is a photograph of the MAC-4 chip.

The MAC-4 contains a program memory (ROM), a data memory (RAM), and many input/output (I/O) features. Both the ROM and the RAM can be expanded through a simple layout modification. The ROM can vary in size between 1024 and 3840 4-bit nibbles, while the RAM size varies between 80 and 192 nibbles.

ARCHITECTURE

MAC-4 consists of five basic functional sections: the internal control, arithmetic-logic unit, special registers and address arithmetic unit, memories, and I/O circuits. The internal control coordinates and executes the sequence of events for an instruction. The arithmetic-logic unit performs the arithmetic/logic operation. All the I/O ports and special registers are addressable in units of 4-bit nibbles. The address width is 12 bits, giving a total directly addressable space of 4096 nibbles.

The user addressable special registers consist of a 12-bit program counter, a 12-bit DMA pointer, an 8-bit stack pointer, two 8-bit memory pointers, an 8-bit processor control register, an 8-bit I/O control register, a 4-bit extended memory pointer, a 4-bit page pointer, and a

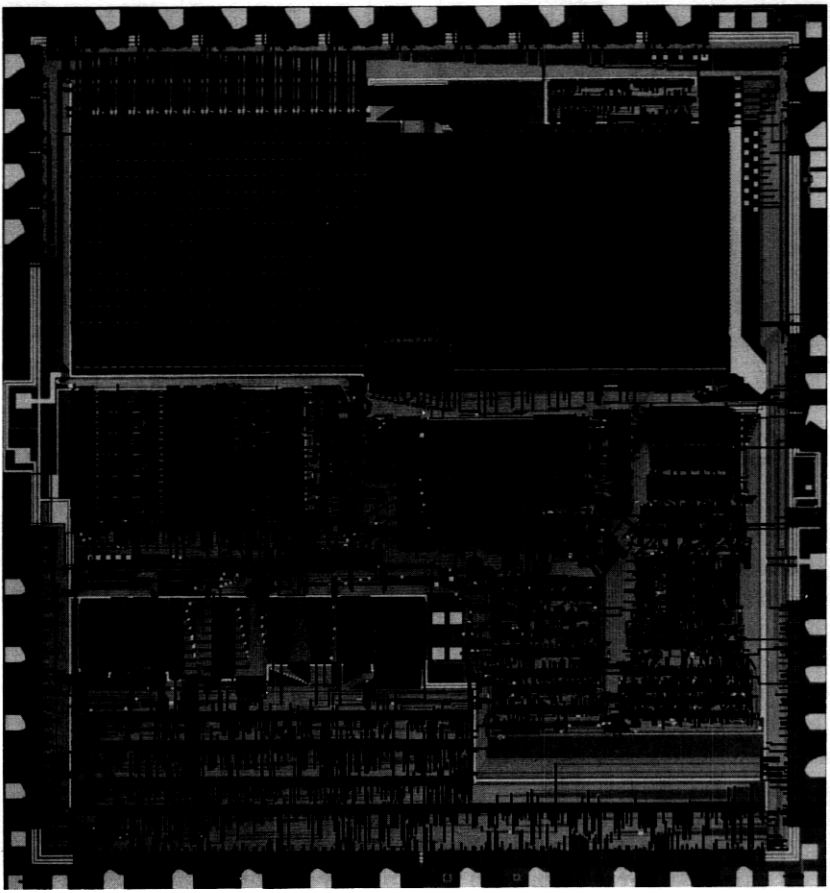

Fig. 1—MAC-4 chip.

4-bit condition register. The program counter stores the current program address. The DMA pointer stores the starting address of the data block for direct memory access operations, or as a count-up value when counting events. The 8-bit stack pointer stores the address of RAM where the return address of a routine call or an interrupt service will be stored. The two memory pointers are used to store operand addresses. The page pointer provides the middle 4-bit nibble of a 12-bit address. The 4-bit extended memory pointer is used together with other 8-bit pointers to provide the upper nibble of the 12-bit address. This offers flexibility by allowing access to ROM as a source, if this is desired.

MAC-4 has a 4-bit-wide data bus. It can, however, process operands of 4-, 8-, 12-, or 16-bit width. The operand width is determined by a 2-bit field in the processor control register, which is under user control.

Of the remaining six bits in the processor control register, four determine optional autoload and autoincrement functions and the remaining two are spares. An autoload operation replaces a specific pointer by the direct address, and an autoincrement operation adds the operand width to the specific pointer.

I/O and special registers appear in the memory address, where they can be used as source and destination for all instructions; this eliminates the need for special I/O instructions. There is no accumulator, but the addressing modes allow all the RAM space, as well as the memory pointer registers, to function as accumulators. This allows true memory-to-memory data transfer for dyadic operations. There are 43 instructions which can be grouped into four categories: the monadic, the dyadic, the program transfer, and the miscellaneous as shown in Table I.

I/O STRUCTURE

Since MAC-4 was conceived mainly as a controller, it is designed with a flexible I/O structure. Thirty-four out of the 40 pins are dedicated to I/O functions. The input/output modes are selected by the 8-bit I/O control register. The lower four bits control the flow of data among the inport, the I/O bus, and the internal data bus. The upper four bits control the data for DMA in/out, DMA serial/parallel, event counter enable, and interrupt enable.

MAC-4 includes a nonmaskable DMA. The user's program sets the appropriate bits of the I/O control register to handle serial or parallel

Table I—MAC-4 instruction set

Monadic	Dyadic
Add one	Add
Rotate right through carry	Add with carry
Rotate left through carry	Subtract
Subtract one	Subtract with borrow
Complement	Logical AND
Clear	Logical OR
Set	Logical exclusive OR
Set bit	Move bit
Clear bite	
Move	
Program Transfer	**Miscellaneous**
Branch or jump	Short call
Branch or jump if equal	Halt
Branch or jump if not equal	Return
Branch or jump if less than	Interrupt return
Decrement and branch or jump if not zero	PCR override
	Call
Branch or jump if bit is set	
Branch or jump if bit is cleared	
Branch or jump if zero	
Branch or jump if not zero	
Test for zero under mask and jump	

data. A DMA request, generated by pulling the DMA request pin low, causes the processor to complete the current instruction and then relinquishes control to the DMA controller. An asynchronous clock strobes the data in or out. The DMA pointer is incremented at every clock pulse for parallel operation or at every fourth clock pulse for serial operation. An interrupt is generated at the end of the DMA, when the DMA request pin goes high; the program then goes to a DMA service routine.

The DMA pointer can also serve as a 12-bit event counter. Each asynchronous clock pulse increments the pointer. A transition from FFF to 000 results in an overflow that generates an interrupt.

MAC-4 has a single-level maskable interrupt that can be activated by pulsing the pin low while the interrupt enable bit in the I/O control register is set. The interrupt takes place at the end of the current instruction. By using maskable interrupt, DMA, and event count features, three distinct interrupts can be generated.

The user-addressable I/O latches provide 16 pins that may be individually designated as either inputs or outputs. Each latch can be set or reset individually or in groups of four. A 4-bit input port relays data into internal memory directly or indirectly through the user PLA encoder. The user PLA is a mask-programmable matrix which can be used for fast code translation. In addition, data may be fed from the 4-bit input port through the PLA encoder directly to the 8-bit I/O bus. Data can also be written to or read from the I/O bus directly. The I/O bus can be used in a bus-structured environment by using four pins normally employed by the I/O latches for strobing data in or out and for handshaking in a master-slave, or multiprocessor, environment.

ACKNOWLEDGMENTS

MAC-4 is the result of teamwork by the members of the Low End Microprocessor Design Group of the Integrated Circuit Laboratory and the members of the Microcomputer Development Group of the Software and Microprocessor Technology Laboratory. Special thanks are due to D.C. Stanzione for his effort in supervising the Microcomputer Development Group at the beginning of this project.