SEPTEMBER 1974
Philip Roybal
National Semiconductor

A MICROPROGRAM DEVELOPMENT SYSTEM

ABSTRACT

An important criterion for determining if a processor is suited to a given job is the instruction set—the options the processor offers for moving data among its logical elements.

The ability to change the instruction set, thus tailoring the effective architecture to the job, would allow a single processor configuration to be used for widely varying applications; and to be as efficient at each of them as a special-purpose processor. The results would be increased throughput and better utilization of memory. Now, for the first time, these advantages may be applied to LSI processors through microprogramming. The tools available for user-microprogramming National Semiconductor's family of IMP Microprocessors are presented in this paper.

INTRODUCTION

All digital processors are made up of some set of logical elements which handle data. The number and type of these elements, and their possible interconnections, determine the power of the processor for a given application. During the evolution of the stored-program computer, the form and function of these elements were gradually developed to handle particular processing chores, and new capabilities were made available to the programmer. Usually however, the programmer could have no direct influence on the actual architecture of his computer. He might provide inputs to the designer for the next generation of processors; but once he had his own, he was stuck with it, for better or worse.

We are now at the stage where there is considerable agreement on the basic elements that constitute a processor, and the programmer has so much capability in the individual logic elements that he rarely finds things he simply cannot do for lack of a functional block. However, he often finds that the permissible ways of passing data among the various blocks, i.e., the instruction set of the processor, are not well-adapted for his needs. If he could change the architecture of the processor to fit different types of applications, he could realize a considerable savings in program memory, as well as an increase in processor speeds *(Figure 1).*

Consider the processor requirements of two different users: one who is building a CRT terminal, and the other who is building an automatic navigational instrument. In the first case, the processor will spend most of its time passing bits in and out, at a high rate of speed. To do this efficiently, it will need instructions that facilitate block data transfers, character file searches, and parity checking. On the other hand, the navigation processor probably handles a relatively small amount of input over a long interval. But it must do involved trigonometric calculations on the data that it gets. It therefore needs instructions such as Multiply and Divide; and a set of trig instructions (sin, cos, etc.) would be extremely useful.

The user attempting to solve both of the above problems with a single processor formerly had to make some compromises in efficiency on one job or the other. Now, for the first time, the technique of microprogramming allows

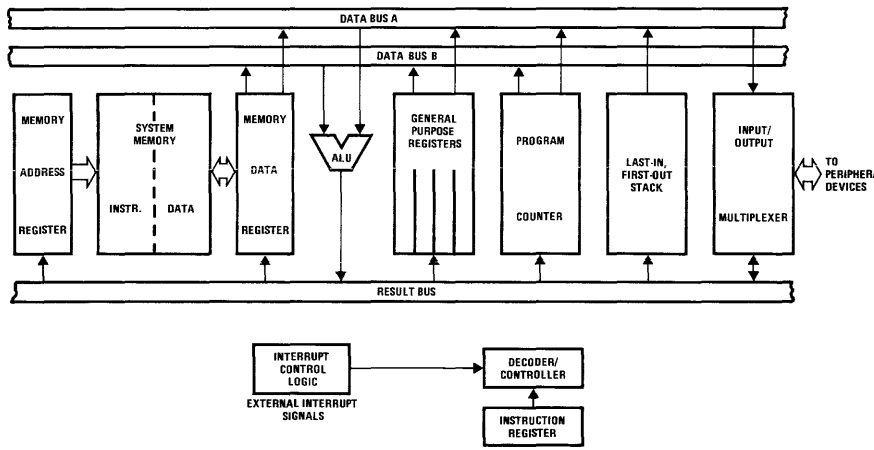

FIGURE 1. Typical Processor Block Diagram. These logical elements are present in most digital processors. The data paths through the processor are generally defined by wiring during system design, and these paths often severely restrict the application of the processor. Special-purpose processors differ from each other not so much in the logical elements present, but in how those elements are interconnected to handle data.

B15M94/Printed in U.S.A.

him to take a standard processor and tailor it to either application with a high degree of efficiency. FACE, in the Microprogram Development System, is the hardware that makes this possible.

IMP MICROPROCESSOR ARCHITECTURE

The National Semiconductor family of microprocessors is designed around two PMOS/LSI building blocks which can be assembled in different combinations to yield anything from a simple 4-bit processor up to a powerful, 32-bit system. The building blocks from which these systems are constructed are the Register, Arithmetic and Logic Unit (RALU), and the Control Read Only Memory (CROM). These chips are designed so that multiple CROM chips may be used to control one to eight RALUs, and yet a single CROM may control up to eight RALUs. There are many possible combinations.

The RALU is a four-bit-wide "slice" of a processor, containing seven general-purpose registers, a status flag register, a 16-word LIFO stack, an Arithmetic and Logic Unit (ALU), an I/O multiplexer, and three data buses. In contrast to the typical processor however, data flow among RALU elements is completely general—any element can talk to any other element. The flow of data, and thus the instruction set, is determined by a second chip: the CROM *(Figure 2)*.

The CROM is a mask-programmed read-only memory, coupled with control and sequencing logic. It contains the microprogram control commands that direct the RALUs and define the instruction set of the system. NS microprocessor systems are supplied with one or more standard CROMs, which contain the basic instruction set of the processor.

In operation, an instruction fetched from memory is passed to a register within the CROM. The contents of this instruction register serve as a starting address for a microcontrol routine in the 100 x 23 bit Microprogram Control Store ROM, the heart of the CROM. This microroutine provides a sequence of control outputs which cause the RALUs and other processor elements to

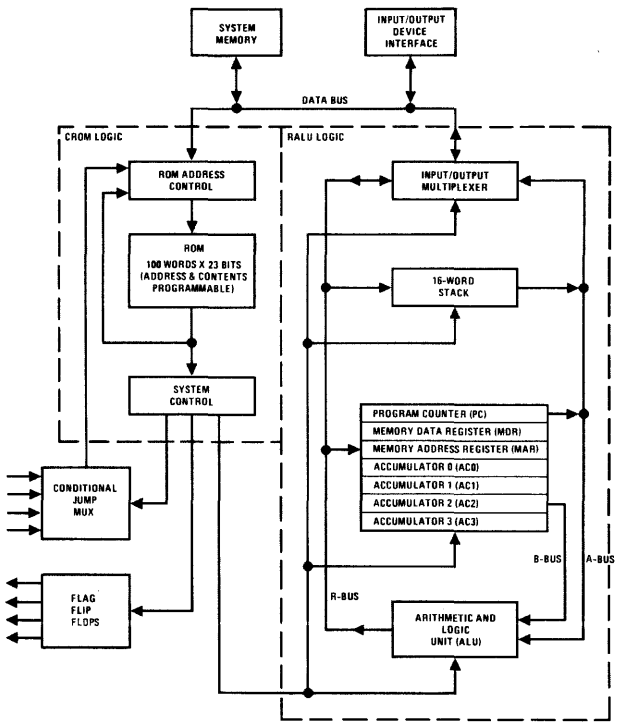

FIGURE 2. CROM/RALU/Processor System Interconnection. The CROM receives instructions from main memory and outputs control commands to the RALUs and the Flag and Jump Condition Multiplexers.

perform basic functions in such a way that the instruction is carried out *(Figure 3)*. The instruction set of the processor can be altered simply by changing the pattern in the CROM.

FACE and the Microprogram Development System

The Field-Alterable Control Element (FACE) is a chip which works with ROM or Read/Write Memory to replace the CROM in a microprocessor system during microprogram development.

The FACE chip, its associated memories, and other development aids together constitute the Microprogram Development System (MDS).

The MDS actually consists of three parts, which are used together during the development of custom microcode. The first part of the system is the Microprogram Control Logic (MCL) board, which contains the FACE chip itself, 512 x 23 bits of bipolar PROM memory, and an interface cable which connects the board to the CROM socket in a microprocessor system. With this board connected, the RALUs and other processor elements communicate with the FACE in exactly the same way that they communicate with a CROM. But now the user can substitute his own microcode (in PROMs) for the mask-programmed code of the CROM. *Figure 4* is a block diagram of the FACE and its microprogram memory.

The second board in the Microprogram Development System is a Writeable Control Store (WCS), which is used with the MCL board above during microprogram debugging. This board contains 512 x 23 bits of Read/Write memory, in which the user can write his microcode. The microcode is written in a symbolic language, assembled with a micro-assembler, and loaded into the WCS board from the processor's main memory, through the use of control flags. Control is then transferred to it so that the new microinstructions can be executed and debugged. The WCS now becomes the Microprogram Control Memory in *Figure 4.*

During the debug process, the user can easily alter microcode within the WCS. Once he has confidence in his code he can transfer it to PROM for more lengthy evaluation in a test bed. He might even use the MCL board (with microcode in PROM) in a limited production run of his equipment, to gain experience with his new code before committing it to masks for a CROM.

The third part of the Microprogram Development System is the Display and Debug Unit, a circuit board which can manipulate and display the microprogram control signals being fed back to the processor system. With this unit, the user can observe and interact with his microcode, stepping through microinstructions and displaying them as required to debug the code. This board is used in conjunction with the other two boards to provide a debug capability similar to that enjoyed by assembly language programmers using standard debug software. Here however, the debug package must be hardware, because the architecture of the processor is changing under control of the new microcode.

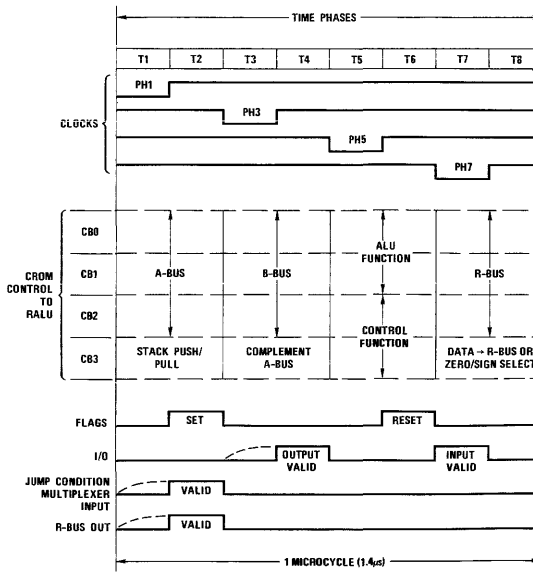

FIGURE 3. RALU Timing Control

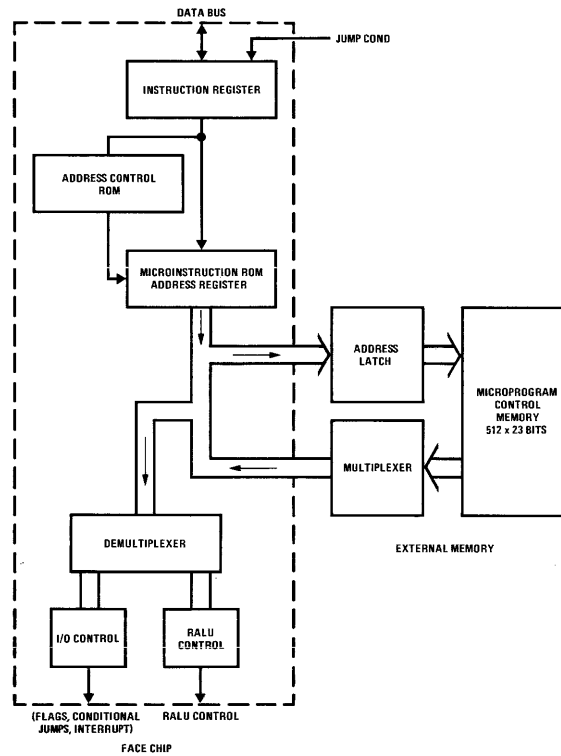

FIGURE 4. FACE in the Microprogram Development System. The internal circuitry of FACE, along with external control memory, substitutes for a CROM in IMP Microprocessor systems. The FACE and its memory mount on the MCL board, which connects to a standard CROM socket through an umbilical cord.

A flow diagram of the steps in the development of new microcode is presented in *Figure 5.*

Use of the Microprogram Development System

The Microprogram Development System is intended for use in the IMP-16L, IMP-16P, and IMP-8P Prototyping Systems during the hardware and software development phases of a project. The MDS plugs directly into these systems, and draws its power from them. Interface to the processor itself is through an umbilical cord from the MCL board to the "2nd CROM" socket on any of National's standard microprocessor boards, or on the user's custom board *(Figure 6).* Because the MDS can operate in parallel with a standard CROM, the programmer has full use of the vendor's verified instruction set on his processor while he is developing new instructions. National's diagnostics and support software will run with the instruction set contained in the standard CROM.

Once the user has debugged his microcode, he can send it to National Semiconductor and a custom-masked CROM can be prepared. This new CROM can then be used in parallel with standard NSC CROMs, or it can be used alone to create a completely new instruction set.

Special instructions such as mathematical functions, data handling and formating, text editing, and error code detection/correction can be implemented in microcode and made transparent to user software. This provides both speed improvement and main memory savings because, in general, microinstructions are more efficient than macroinstructions. For instance, in the case of National Semiconductor's 16-bit multiply and divide instructions, the effective execution speed was increased by a factor of 6:1 while storage requirements were reduced by 13:1.

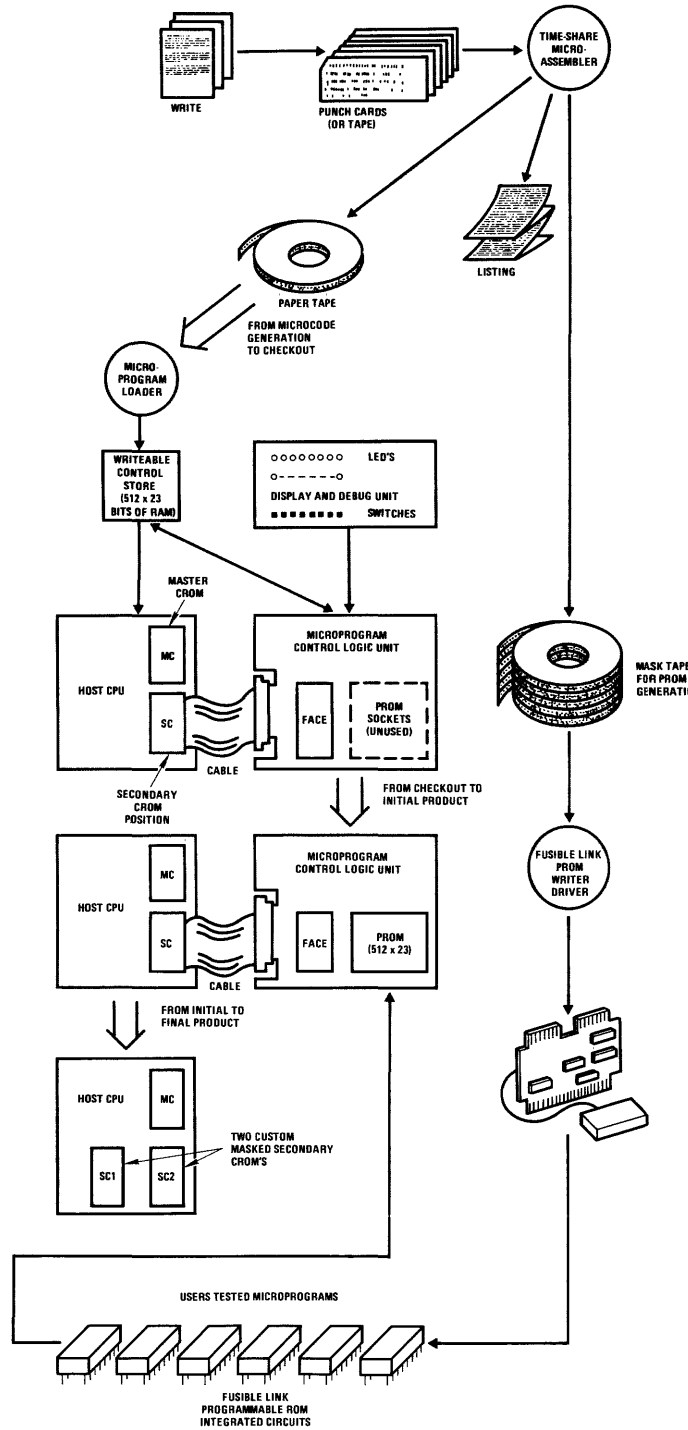

FIGURE 5. Development Cycles for User Designed Microprocessor Instruction Set Through Custom CROM's.

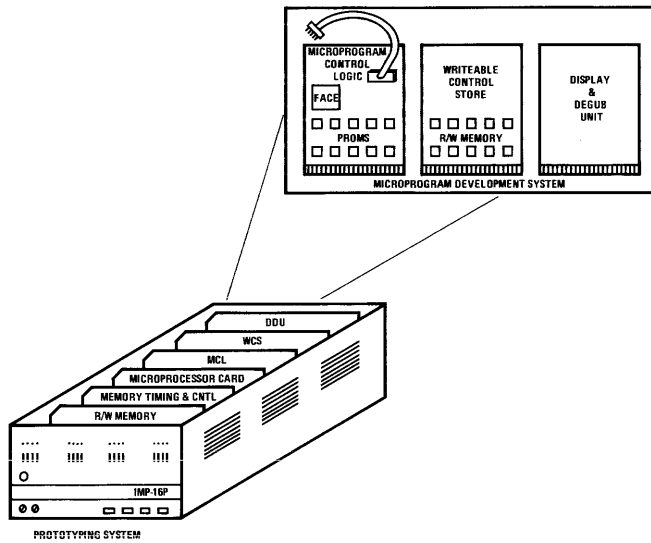

FIGURE 6. Microprogram Development System, as it would be used with a Prototyping System during Microprogram Development.

CONCLUSION

A new Microprogram Development System (MDS) using National Semiconductor's FACE chip allows the advantages of microprogramming to be applied to LSI processors for the first time.

This greatly increased the flexibility of the processor, and allows standard microprocessor hardware to be used in applications that would otherwise require a special-purpose processor. The resultant user benefits are higher processor speed, greater system flexibility, less memory required, and easier application programming.

BIBLIOGRAPHY

Alan Weissberger, *User Microprogram Development for an LSI Processor.* National Semiconductor Corp., 1974.

Manufactured under one or more of the following U.S. patents: 3083262, 3189758, 3231797, 3303356, 3317671, 3323071, 3381071, 3408542, 3421025, 3426423, 3440498, 3518750, 3519897, 3557431, 3560765, 3566218, 3571630, 3575609, 3579059, 3593069, 3597640, 3607469, 3617859, 3631312, 3633052, 3638131, 3648071, 3651565, 3693248.

National Semiconductor Corporation
2900 Semiconductor Drive, Santa Clara, California 95051, (408) 732-5000/TWX (910) 339-9240
National Semiconductor GmbH
808 Fuerstenfeldbruck, Industriestrasse 10, West Germany, Tele. (08141) 1371/Telex 05-27649
National Semiconductor (UK) Ltd.
Larkfield Industrial Estate, Greenock, Scotland, Tele. (0475) 33251/Telex 778-632