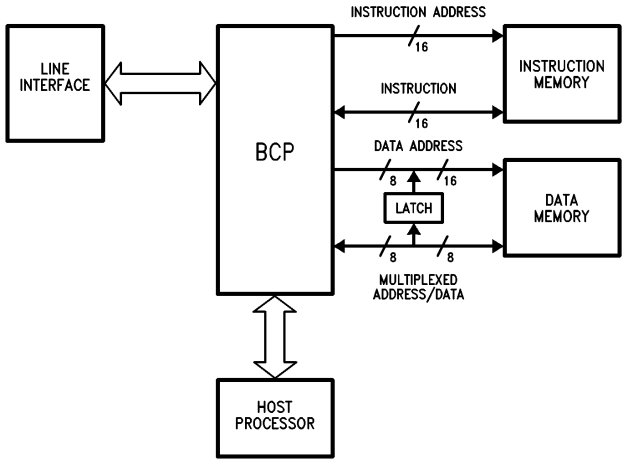
DP8344 BCP Stand-Alone Soft-Load System

National Semiconductor
Application Note 504
Jim Margeson
July 1989

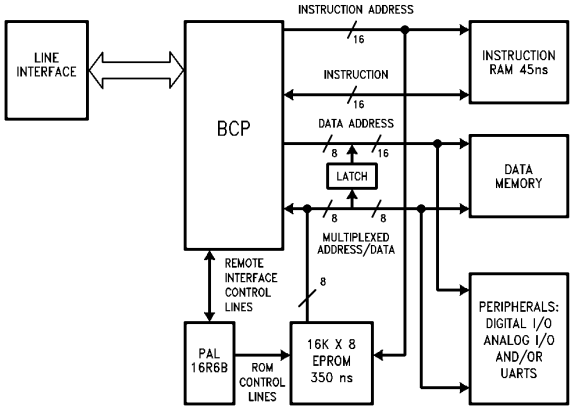
INTRODUCTION

The DP8344 Biphase Communications Processor (BCP) is a 20 MHz Harvard architecture microprocessor with an on-chip transmitter and receiver. The BCP can be used to implement several biphase communication protocols: IBM 3270, IBM 3299, IBM 5250, and National's general purpose 8-bit protocol. This application note shows how DP8344 software can be loaded from EPROM into instruction RAM. It is particularly valuable in stand-alone systems where the BCP is not interfaced to a host processor. Possible applications include: protocol converters, multiplexers, high-speed remote data acquisition systems and remote process control systems.



TL/F/9403–1

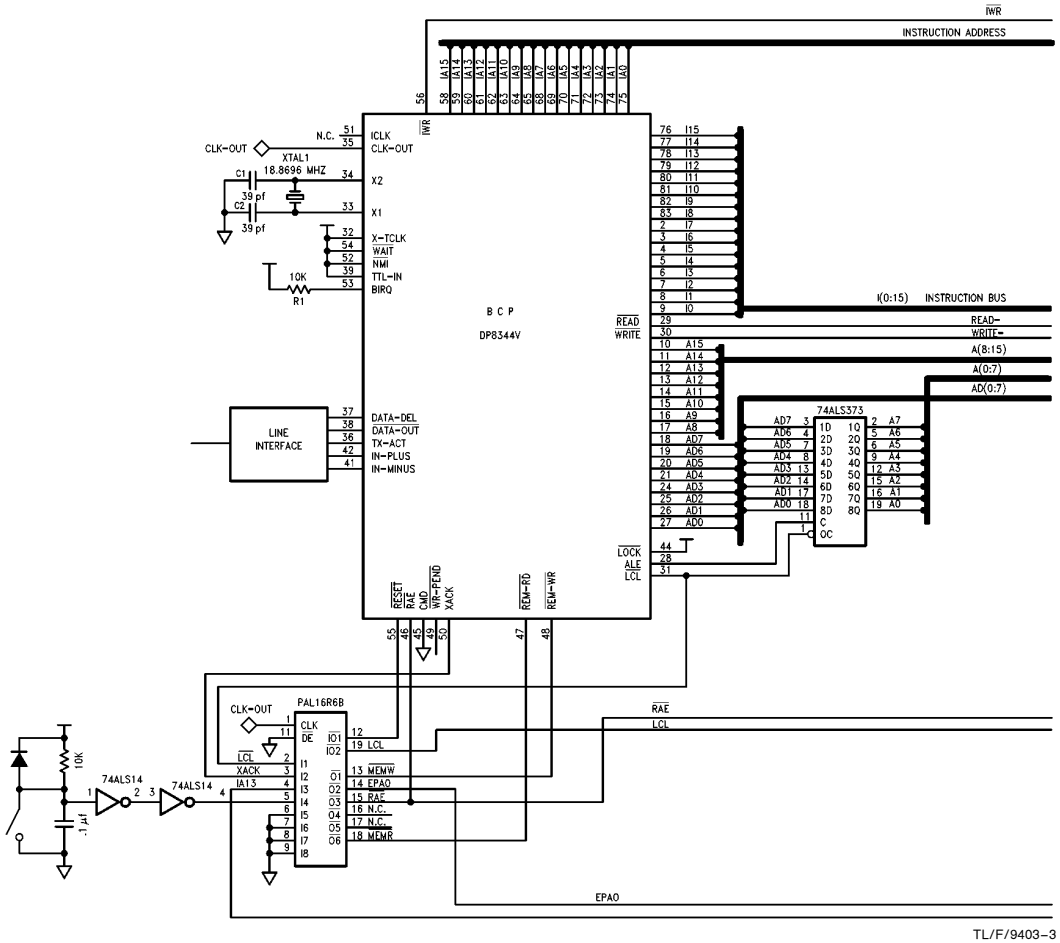
FIGURE 1. BCP System with Host Processor



TL/F/9403–2

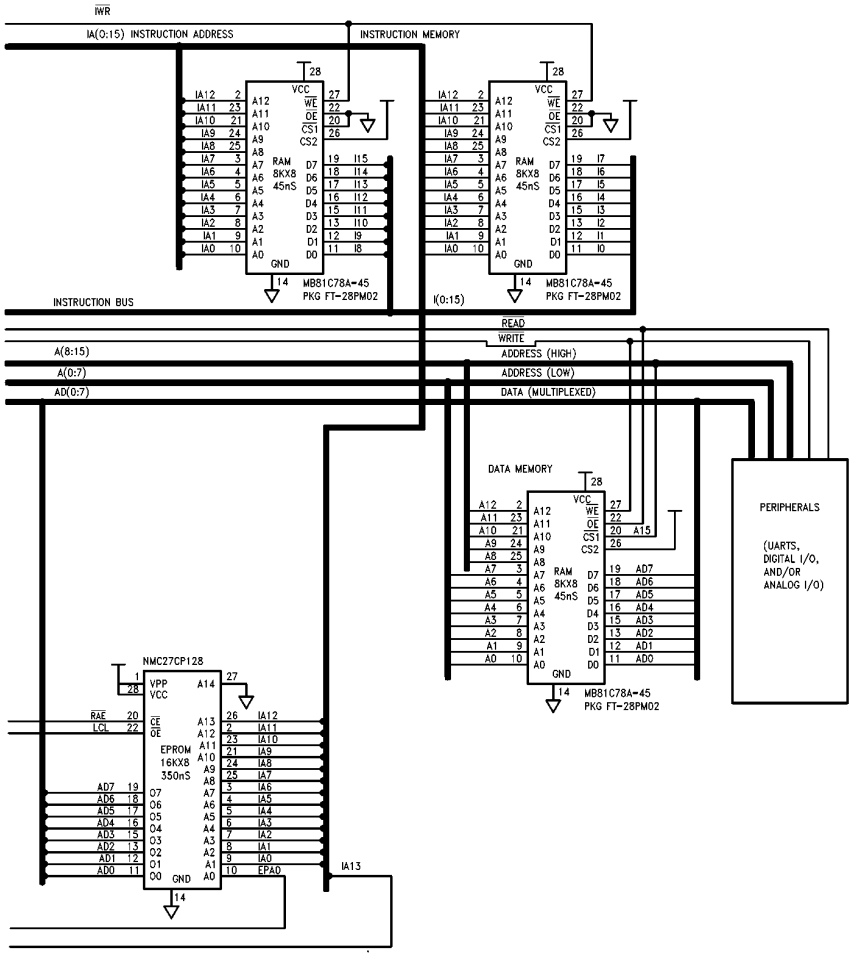
FIGURE 2. BCP Stand-Alone System with EPROM Soft Load Circuit

RRD-B30M105/Printed in U. S. A.



TL/F/9403–3

FIGURE 3. Schematic



TL/F/9403–4

FIGURE 3. Schematic (Continued)

WHY EPROM SOFT-LOAD?

In a stand-alone application, the BCP instruction code must be kept in non-volatile memory. Instruction memory with 45 ns access time is required to run the BCP at full speed.

EPROM at this speed can be quite expensive, much more than 45 ns RAM or 350 ns EPROM. RAM with 45 ns access time can be used for instruction memory if a scheme is employed to load the BCP code into the RAM from slow (350 ns), inexpensive EPROM, upon power-up.

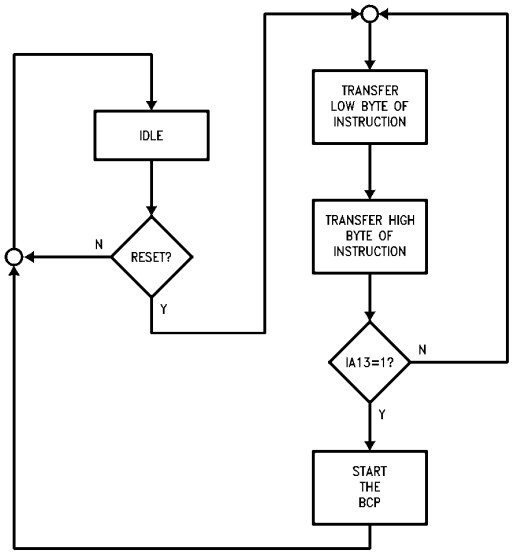
In non-stand-alone applications, a host processor would communicate with the BCP through the BCP's built-in remote interface *(Figure 1)*. In such a system, BCP code would be loaded from the host into the BCP's instruction RAM using the remote interface. In a stand-alone system, however, the BCP is not interfaced to a host; the program is loaded from EPROM through the remote interface. As shown in *Figure 2* a PAL® sequencer controls the loading of the program, generating handshaking signals similar to those of a typical host processor. When the load is complete, the sequencer tells the BCP to begin execution of the program.

HOW THE SOFT-LOAD CIRCUIT WORKS

The BCP, as configured in this system, comes up halted after reset *(Figure 3)*. The program counter is set to zero, and the remote interface is configured to receive 16-bit instructions in 8-bit pieces and write them into instruction memory. The BCP has the feature that it can be configured to come up stopped or to begin program execution after a reset has occurred. If the following conditions are true when reset is de-asserted then the processor will begin running: RAE∼ (Remote Access Enable, active low) = High, REMWR∼ (Remote Write, active low) = low, REMRD∼ (Remote Read, active low) = low. Otherwise, it will come up halted.

The PAL sequencer begins the software load by writing the low byte of the first instruction to the remote interface. A simplified flowchart of the sequence operation is shown in *Figure 4.*

This byte comes from address 0000H of the EPROM. The corresponding locations of EPROM and RAM are shown in *Figure 5*. The least significant address line of the EPROM is controlled by the sequencer; the other address lines are driven by the instruction address bus of the BCP. The instruction address bus reflects the contents of the BCP's program counter (PC), which contains the destination of the instruction currently being loaded. After the low byte of the first instruction is written to the remote interface, the sequencer brings the least significant address line of the EPROM high. Now location 0001H of the EPROM is addressed, and the high byte of the first instruction is written to the remote interface. At this point the BCP writes both bytes into address 0000H of instruction RAM, and increments its program counter.



TL/F/9403–5

FIGURE 4. Sequencer Operation

EPROM Address	Instruction Memory Address	
0	0	(Low Byte)
1	0	(High Byte)
2	1	(Low Byte)
3	1	(High Byte)
4	2	(Low Byte)
5	2	(High Byte)
•	•	•
•	•	•
•	•	•
•	•	•
16382	8190	(Low Byte)
16383	8191	(High Byte)

FIGURE 5. EPROM to RAM Address Mapping

The first 16-bit instruction has been transferred; the second is done in a similar manner. The sequencer brings the least significant address line of the EPROM low again. The PC now contains 0001H, which is output on the instruction address bus. Location 0002H of the EPROM is addressed, and the low byte of the second instruction is written to the remote interface. The sequencer then brings the least significant address line of the EPROM high (to address location 0003H) and the high byte of the second instruction is transferred. The BCP writes the second 16-bit instruction to location 0001H of instruction RAM. This process is repeated until the last instruction is transferred.

The sequencer senses that the load is complete when instruction address line 13 comes high. This occurs when the program counter is incremented to a value of 4000H, indicating that 8K instruction words have been transferred. At this point the BCP must be started. To achieve this, the sequencer resets the BCP again, while holding RAE~ high, REMRD~ low, and REMWR~ low. A reset during these conditions brings the processor up running, and also clears the program counter. The BCP begins execution at instruction address 0000H and the sequencer and EPROM go into an inactive state, transparent to the software being executed. A detailed version of the sequencer flowchart is shown in *Figure 6.* A hardware compiler/minimizer was used to obtain the equations shown in *Figure 7.* These equations were used to program a National PAL16R6B. Typical timing waveforms of the soft-load are shown in *Figure 8.*

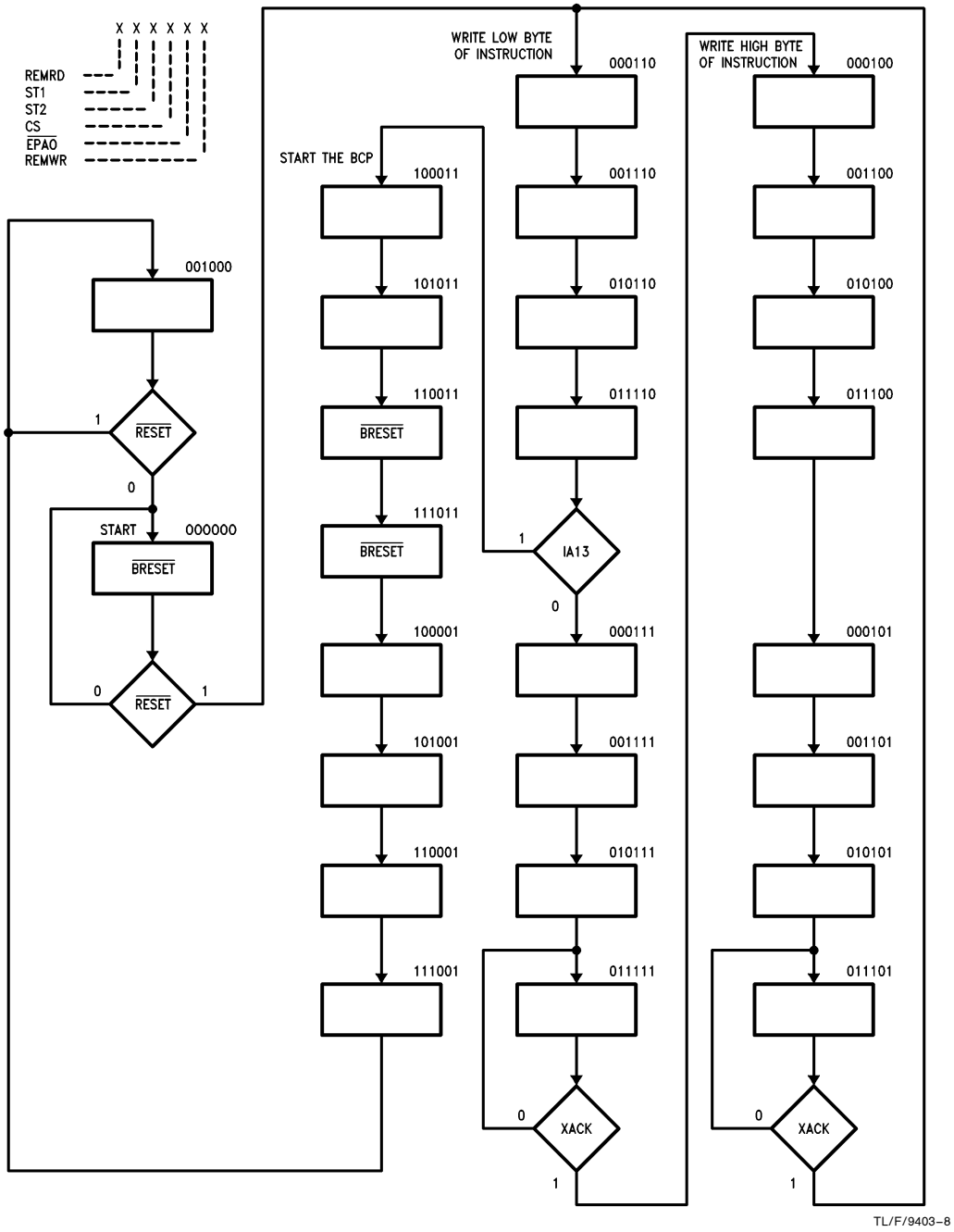


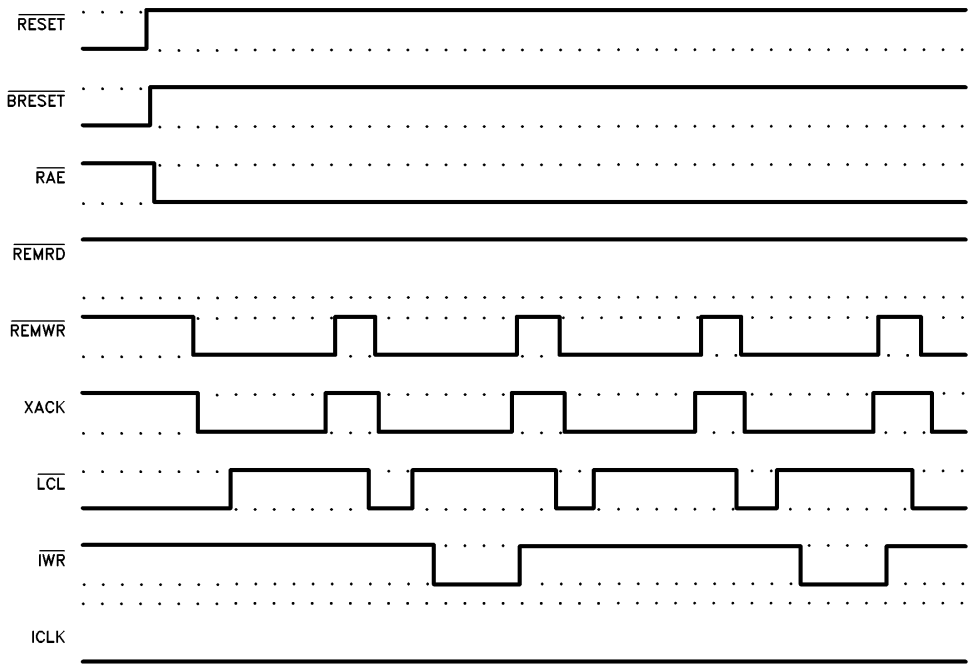
FIGURE 6. Sequencer Flowchart

There are several advantages to using the remote interface to load the BCP software. If a scheme like the one in *Figure 9* was used to load the program directly from EPROM to instruction RAM, much more hardware would be required and the access time of the RAM would need to be shorter. Two EPROMs would have to be used instead of one because the transfer would be 16 bits wide instead of 8 bits. In this case the BCP's program counter could not be used to increment through the memory locations, thus an external 13-bit counter would be needed. TRI-STATE® buffers would isolate the RAM and EPROM from the instruction data and instruction address busses during soft-load. These buffers would add propagation delays to memory accesses demanding that faster RAM be used. Soft-loading through the remote interface requires fewer I.C.'s and does not degrade the performance of the processor.

```
DMPAL16R6B;   SOFTLOAD
CK LCL XACK IA13 RESET NC6 NC7 NC8 IWR GND
/OE /BRESET /REMWR /EPA0 /CS /ST2 /ST1 /REMRD /LCLINV VCC
/REMRD := RESET*     /REMRD*           CS*/EPA0*/REMWR
        + RESET*     /REMRD*      ST2* CS*      /REMWR
        + RESET*     /REMRD* ST1*      CS*      /REMWR
        + RESET*IA13* REMRD*/ST1*/ST2*/CS*/EPA0* REMWR
/ST1   := RESET*      REMRD*/ST1* ST2*/CS
        + RESET*      REMRD* ST1*/ST2*/CS
        + RESET*     /REMRD* ST1*/ST2* CS*      /REMWR
        + RESET*     /REMRD*/ST1* ST2* CS*      /REMWR
        + RESET*/XACK*REMRD*     /ST2*/CS*      /REMWR
/ST2   := RESET*      REMRD*      ST2*/CS
        + RESET*/XACK*REMRD*/ST1*     /CS*      /REMWR
        + RESET*     /REMRD*      ST2* CS*      /REMWR
        + RESET*     /REMRD*/ST1*      CS* EPA0*/REMWR
        + RESET*      REMRD* ST1*/ST2* CS* EPA0* REMWR
/CS    := RESET*      REMRD*          /CS*      /REMWR
        + RESET*      REMRD* ST1*     /CS
        + RESET*      REMRD*          /CS* EPA0
        + RESET*      REMRD*      ST2*/CS
        + RESET*      REMRD* ST1* ST2*     EPA0* REMWR
        + RESET*/IA13*REMRD*          /CS
*/EPA0  := RESET*      REMRD*      ST2*/CS*/EPA0
        + RESET*/XACK*REMRD*          /CS*/EPA0
        + RESET*      REMRD*          /CS*/EPA0* REMWR
        + RESET*      REMRD* ST1*     /CS*/EPA0
        + RESET*     /REMRD* ST1*      CS*/EPA0*/REMWR
        + RESET*     /REMRD*      ST2* CS*/EPA0*/REMWR
        + RESET*XACK* REMRD*/ST1*/ST2*/CS*EPA0*/REMWR
        + RESET*      REMRD* ST1* ST2* CS*EPA0* REMWR
/REMWR := RESET*     /REMRD*       ST2*/CS*     /REMWR
        + RESET*      REMRD* ST1*      /CS*     /REMWR
        + RESET*     /REMRD*           CS*/EPA0*/REMWR
        + RESET*     /REMRD*      ST2* CS*      /REMWR
        + RESET*      REMRD*/ST1*/ST2*/CS*       REMWR
        + RESET*     /REMRD* ST1*      CS*      /REMWR
        + RESET*/XACK*REMRD*          /CS*      /REMWR
/BRESET = /RESET +   /REMRD*/ST1*      CS*/EPA0*/REMWR
/LCLINV = LCL
```

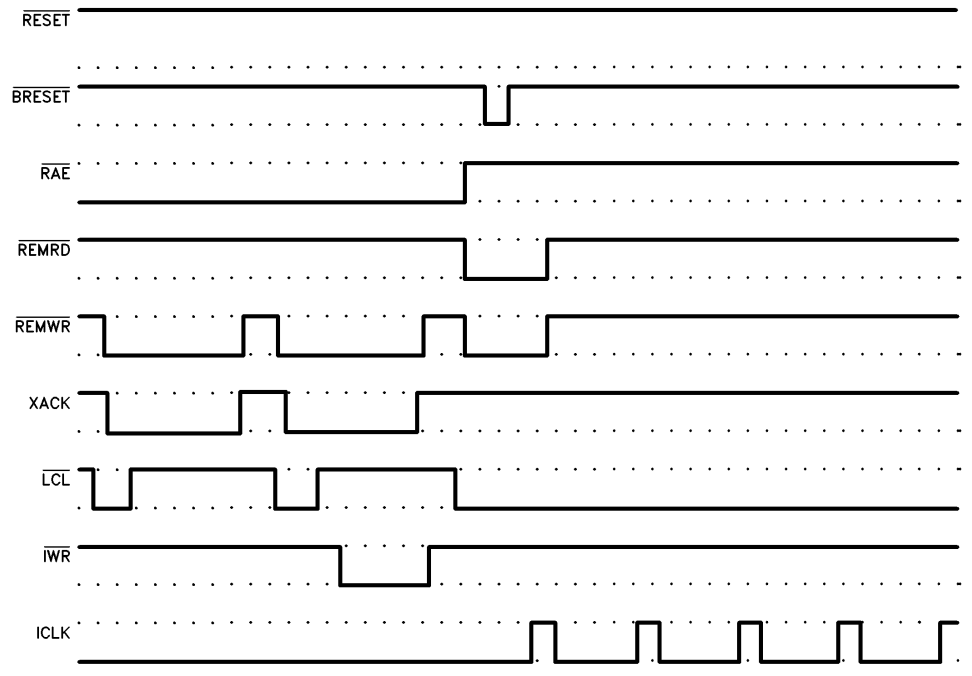
FIGURE 7

Timing at Beginning of Instruction Load



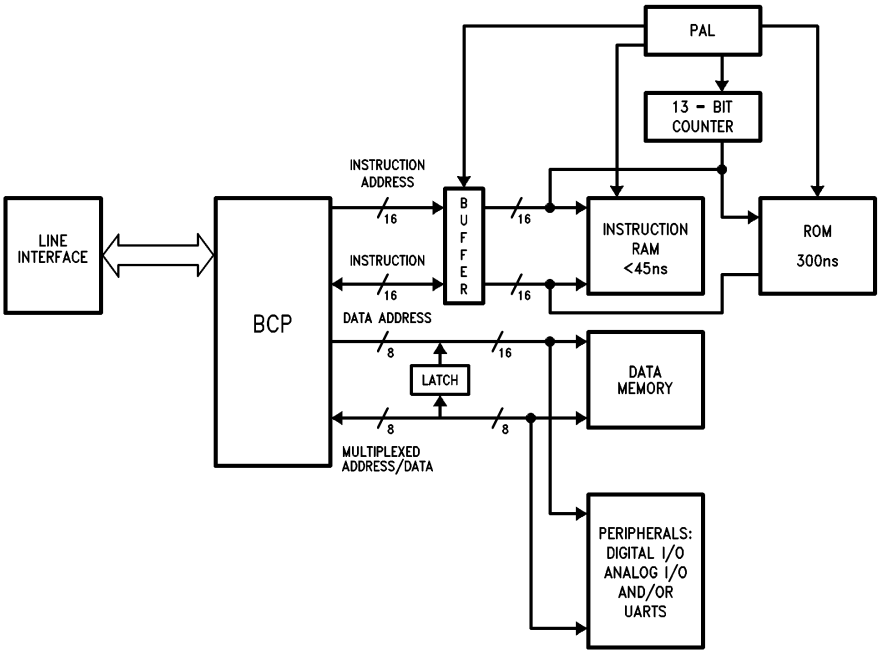
TL/F/9403–6

Timing at End of Instruction Load



TL/F/9403–9

FIGURE 8. Example of Timing Waveforms



TL/F/9403–7

FIGURE 9. Another Method of Soft-Loading (A Non-Ideal Solution)

MODIFYING THE SOFT-LOAD SYSTEM FOR LARGER MEMORY

The soft-load system as documented loads 8K x 16 bits of instruction memory. Large programs may require more memory; smaller, lower cost systems may use less. The soft-load system can easily be altered to load larger or smaller instruction memory by changing one connection.

Connecting a different instruction address line to pin 4 of the PAL changes how much instruction memory is loaded: These connections are shown in *Figure 10.*

Instruction Memory Size:	Connect Pin 4 of PAL to:
32k x 16	IA15
16k x 16	IA14
8k x 16	IA13
4k x 16	IA12
2k x 16	IA11

FIGURE 10. Connections for Altering Instruction Memory Size

USING THE CAPSTONE CT-104 DEVELOPMENT BOARD TO EVALUATE THE SOFT-LOAD APPLICATION

A DP8344 biphase Communications Process development board is available from Capstone Technology Inc., of Fremont, California. The board is designed to reside in an IBM® PC. A breadboard area is provided on the board so that custom circuitry can be added. It can be converted into a stand-alone soft-load system by wire-wrapping three additional I.C.'s into the breadboard area. A diagram of the CT-104 board with the additional components is shown in *Figure 11.* Note that most of the prototyping area remains available, enabling the addition of other circuitry specific to the application being developed. A parts list is shown in *Figure 12.* The PAL16R6 is programmed with the equations shown in *Figure 7.* U22 and U23 must be removed from the CT-104 board and be replaced with specially wired 20-pin headers. The wiring on these headers, shown in *Figure 13,* provides access to the RESET~ signal and disables the unused interface circuitry on the board. Pin 11 of the header that replaces U23 must be wired to pin 13 of the 74LS14. A wiring list is shown in *Figure 14.* Power supply connections must be added because the board can no longer reside in the PC. Development of a stand-alone soft-load application can be done easily and quickly by using the CT-104 board because minimal circuit construction is required.

SUMMARY

The soft-load circuit uses the BCP's remote interface to load BCP code from slow EPROM to fast RAM, with a minimum of extra hardware. This method is useful in systems where there is no host processor directly interfaced to the BCP and the full processing speed of the BCP is needed.

The circuit can easily be modified to load different sizes of memory. The Capstone Technology, Inc. CT-104 development board can easily be converted to a stand-alone soft-load system for evaluation of the application.

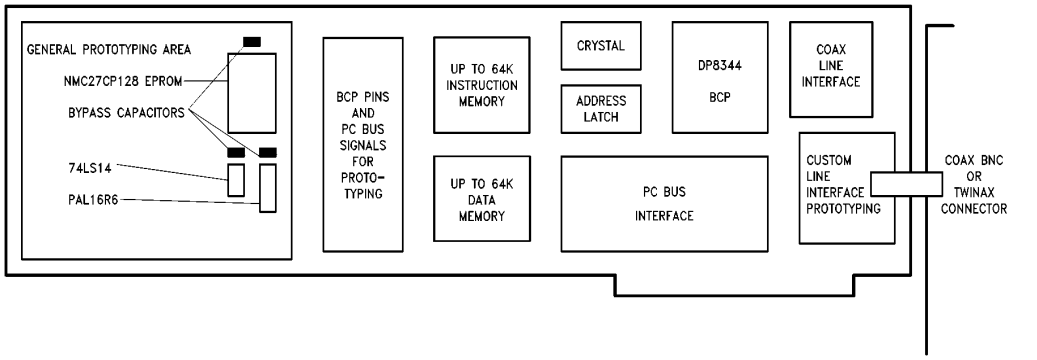


FIGURE 11. CT-104 Development Board with Soft-Load Circuitry

NMC27CP128 350 ns access time or faster
PAL16R6B
DM74LS14N
28-pin wire-wrap socket
20-pin wire-wrap socket
14-pin wire-wrap socket
3 Bypass capacitors, 0.1 μF
2 50-pin wire-wrap strips, 2 pins wide
2 20-pin headers

FIGURE 12. Parts List for Conversion of CT-104 Board

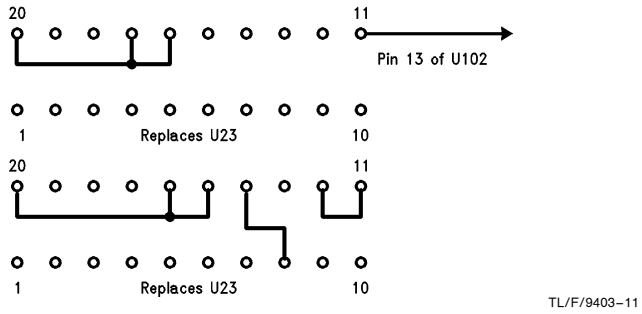


FIGURE 13. Header Wiring for Conversion of CT-104 Board

Pin	Unit	to	Pin	Unit
1	U100		—	VCC
2	U100		12	W1
3	U100		7	W1
4	U100		6	W1
5	U100		5	W1
6	U100		4	W1
7	U100		3	W1
8	U100		2	W1
9	U100		1	W1
10	U100		14	U101
11	U100		33	W1
12	U100		34	W1
13	U100		35	W1
14	U100		—	GND
15	U100		36	W1
16	U100		37	W1
17	U100		38	W1
18	U100		39	W1
19	U100		40	W1
20	U100		46	W2
21	U100		10	W1
22	U100		19	U101
23	U100		11	W1
24	U100		9	W1
25	U100		8	W1
26	U100		13	W1
27	U100		—	VCC

Pin	Unit	to	Pin	Unit
28	U100		—	VCC
1	U101		17	W2
2	U101		11	W2
3	U101		7	W2
4	U101		14	W1
5	U101		10	U102
6	U101		—	GND
7	U101		—	GND
8	U101		—	GND
9	U101		50	W1
10	U101		—	GND
11	U101		49	W2
12	U101		8	W2
13	U101		48	W2
15	U101		46	W2
18	U101		47	W2
20	U101		—	VCC
1	U102		—	GND
3	U102		—	GND
5	U102		—	GND
7	U102		—	GND
9	U102		—	GND
11	U102		12	U102
13	U102		11	U23 HEADER
14	U102		—	VCC
45	W2		—	GND

FIGURE 14. Wiring List for Conversion of CT-104 Board

Lit. # 100504

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408