Eight-Bit Bus Interface for the NS32CG16; NS32CG16 Applications Note 6

National Semiconductor
Application Note 635
Tom Norton
April 1991

1.0 INTRODUCTION

The NS32CG16 is a 32-bit CMOS, graphics oriented processor. It has a 16-bit external data bus and a 16 Mbyte linear address space. It is software compatible with other Series 32000® CPUs, with new instructions for high speed graphics. The NS32CG16 is designed specifically for page oriented printing technologies such as laser, LCS, LED, and Ink Jet.

This applications note discusses an 8-bit bus interface for the NS32CG16. The NS32CG16 bus interface is normally 16 bits wide, yet when compressed into 8 bits provides a small, low cost system with 32-bit CPU performance. The low cost is attributed to the size and organization of memory. 16-bit wide memory chips are becoming available, but the cost is prohibitively high. Hence, this application discusses a low cost design with one 8-bit wide SRAM and one 8-bit wide EPROM. The relatively simple logic required for a 16- to 8-bit interface will fit in a small, 600 gate array with room for additional functions.

2.0 DESCRIPTION

This note contains the features and specifications for designing an 8-bit bus interface for the NS32CG16 high performance printer processor. A complete 8-bit system is described, however, specific details are given regarding the bus interface.

The user can download and execute programs on the board (with a system clock of 15 MHz) using the dbg32 debug utility for testing and demonstrating the instructions in the NS32CG16 CPU.

The interface does not restrict the NS32CG16 on the size or type of data transfer. This means that operands, program code, and interrupts are treated the same as if a 16-bit data bus were used. In other words, the bus interface operation is transparent to the CPU, and thus transparent to user programs.

3.0 BUS INTERFACE OPERATION

Data transfers in Series 32000 software consist of byte, word, and doubleword sizes (with even or odd alignment depending on the address). However, every NS32CG16 bus cycle consists of one of the three following formats: even byte, odd byte, or even word. In general terms, the bus interface accomplishes a data transfer as a byte wide data multiplexer/demultiplexer (see *Figure 1*). The interface is a simple fall through buffer for byte transfers. The byte read case is handled as a 1:2 byte demultiplexer where the data path (AD0–AD7, or AD8–AD15) is determined by the alignment of the addressed data (even or odd). For byte write operations, the bus interface is a 2:1 byte multiplexer where the source data path (AD0–AD7, or AD8–AD15) is again determined by the alignment.

Even word transfers occur in 2 8-bit cycles. During even word read cycles, the least significant byte is read during the first cycle and temporarily stored in a buffer/register. During the second cycle the most significant byte is read and made available, through a buffer, to the upper half of the NS32CG16 data bus (AD8–AD15).

For even word write cycles, the least significant byte is passed through a buffer/register during the first cycle, and the most significant byte is passed through a buffer during the second cycle.

Separate wait logic is included to accommodate slower memory and peripheral devices in the 8-bit system. Wait requests are supplied to the CPU via the CWAIT/ pin (see *Figure 2*).

Wait logic is recreated on the bus interface side to provide two binary weighted (B8WAIT1/ and B8WAIT2/) and a continuous wait signal B8CWAIT/ for 8-bit system wait requests. The bus interface tests these signals at the beginning of an 8-bit system bus cycle (see *Figure 3*) and extends CPU bus cycles with the CPU CWAIT/ signal. The CPU signals WAIT1/ and /WAIT2/ are left to the system designer for 16-bit system wait requests.

The CPU starts the bus interface state machine when it accesses the address range containing the 8-bit section of the system. Next, wait requests and the size of transfer (byte or word) are tested. For byte bus cycles, the interface will enable the appropriate data buffer and memory or I/O device and proceed as a "normal" NS32CG16 bus cycle (with wait states if needed). The only differences between even and odd byte transfers are the high byte enable (HBE/) signal and address bit 0 (AD0). A byte transfer has no extra overhead because wait states are added only when accessing slower memories or peripherals (byte cycles are fall through).

Even word bus transfers occur in two 8-bit cycles. With no wait requests a minimum of eight CTTL clocks is required for word cycles. The best case word transfer consists of four CTTL clocks plus four wait states (CWAIT/ pulled low for four clocks). Each wait request added to an even word bus cycle adds two wait requests to the CPU (one for each byte access). See *Figure 4* for a breakout of the number of CTTL clocks for each bus cycle.

The bus interface provides sequential addressing to memory during even word transfers by ORing the cycle count bit with the buffered address bit 0 (BA00) (see *Figure 5*). During the first, or least significant, byte transfer the system address bit is 0, and 1 during the most significant byte transfer.

Series 32000® is a registered trademark of National Semiconductor Corporation.
PAL® is a registered trademark of and used under license from Monolithic Memories, Inc.

TL/EE10494

RRD-B30M75/Printed in U. S. A.

Eight-Bit Bus Interface for the NS32CG16; NS32CG16 Applications Note 6

4.0 BUS INTERFACE SPECIFIC FEATURES

1. Automatic conversion of 8- to 16-bit, or 16- to 8-bit transfers.
2. Fall through for 8-bit CPU cycles (no extra overhead).
3. Wait generation controlled by jumpers, a wait counter, and PAL.
4. Permits small system with one 8-bit wide RAM, and one EPROM.
5. Allows existing debug tools full utilization.

5.0 SYSTEM FEATURES

1. NS32CG16-15 32-bit CPU
2. NS32081-15 FPU (Floating Point Unit) for floating operations. The FPU is directly connected to the NS32CG16 data bus without buffering required.
3. NS32202-10 ICU (Interrupt Control Unit) interfaced to CPU. The ICU has 16 interrupt inputs, 2 16-bit timers, and an 8-bit I/O port.
4. 32k bytes of ROM/EPROM (32k x 8). 2–3 wait states at 15 MHz.
5. 32k bytes of Static RAM (32k x 8). No wait states at 15 MHz. Socket array is provided for expansion to 128k bytes.
6. Serial I/O—2 RS232C ports, configured for MONCG/DBG32 debug. One port is used for terminal I/O and the other for Host I/O. 9600 baud is supported (initialized in MONCG).
7. Memory and I/O map controlled with PAL devices to permit easy changes. The memory is fully decoded to allow for complete benchmark analysis.
8. LED indicators to show board status. The LEDs are DUART output port status indicators. At power-on or reset, the MONCG program writes to the output port bits of the DUART turning on the LEDs. It demonstrates that the board runs a short section of diagnostic code.
9. 2 push buttons, NMI and RESET to NS32CG16.
10. Toggle switch connects CPU NMI interrupt pin to ICU timer output or external push button switch.
11. MONCG installed in EPROMs (Monitor program to control board).
12. "Splice" (Hardware/Software debug tool) Control signal interface.
13. Single +5V power supply with ±5% tolerance.

Note: The EXTBLT instruction is not supported in this system example. Extra hardware can be added to manage 16-bit transfers to/from the DP8510 BPU (Bitblt Processing Unit).

6.0 MEMORY AND I/O MAP

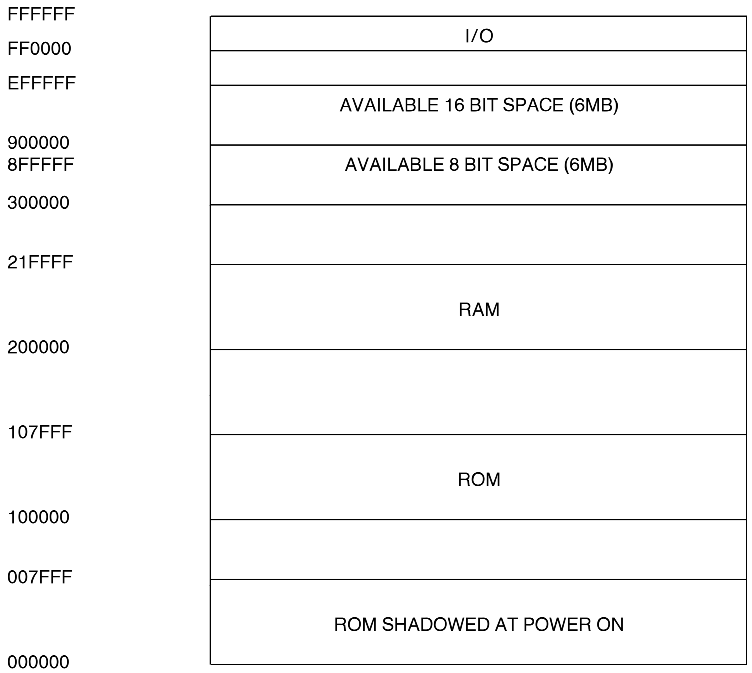
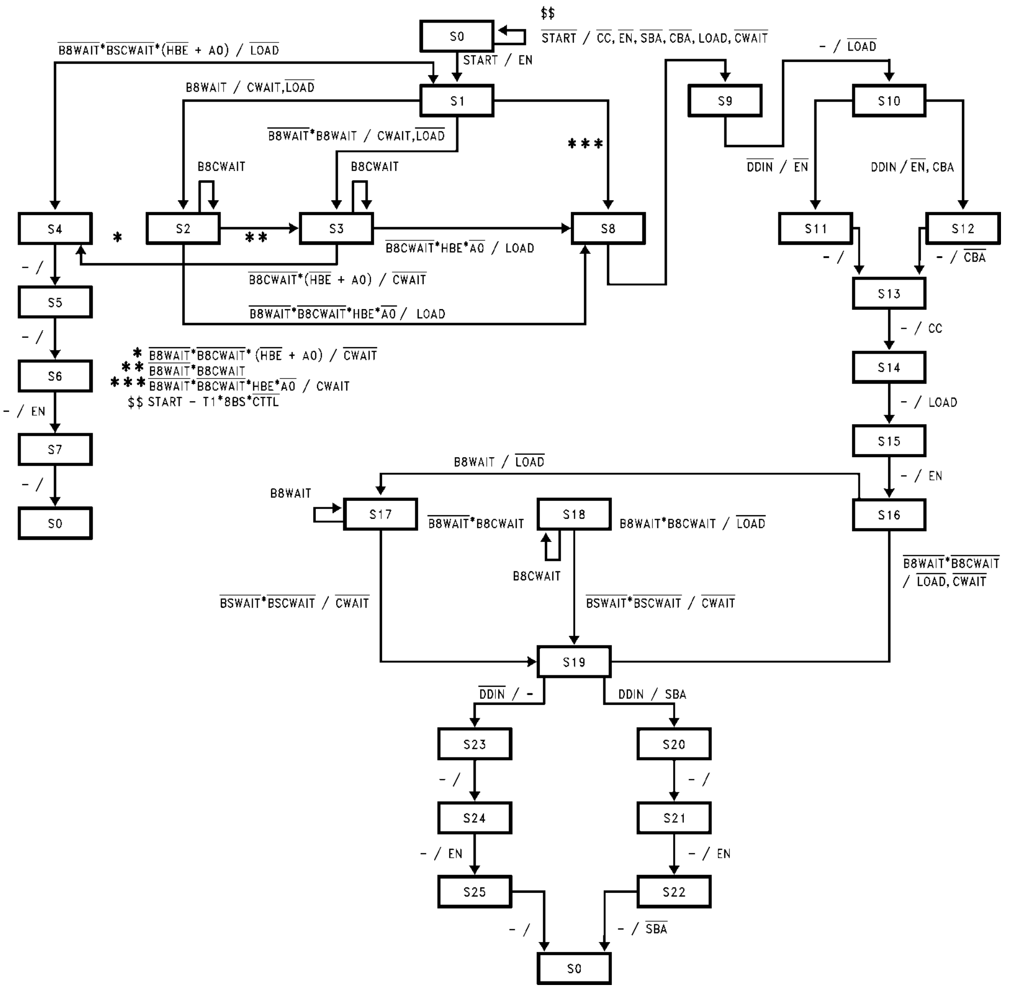


FIGURE 1

Note: At power on, the ROM shadow flip-flop is reset and the 32k bytes of ROM can be accessed at location range 000000–007FFF or 100000–107FFF (ROM "shadow"). MONCG executes a dummy write cycle to location 100005 and sets the flip-flop allowing the lower 32k of RAM to occupy 000000–007FFF and 200000–207FFF. For example, when the ROM shadow flip-flop is set (RAM), accessing location 000005 would be the same as accessing location 200005.

7.0 STATE DIAGRAM/TABLE



TL/EE/10494–1

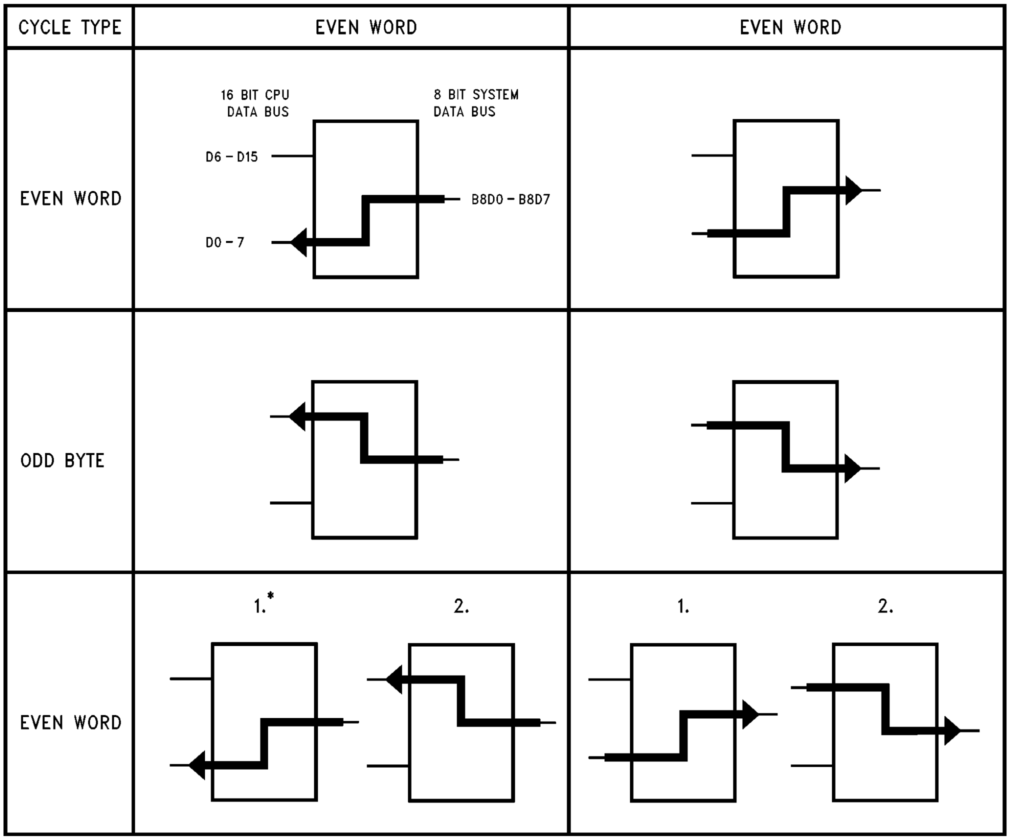
FIGURE 2. BUS8 Interface State Diagram

														OUTPUT
PRESENT	INPUT/NEXT STATE													C
STATE	1	2	3	4	5	6	7	8	9	10	11	12	13	LW
														OA
														SCAI
														CNBBDTQQ
														C/AA//10
S0	S0	S1	X	X	X	X	X	X	X	X	X	X	X	01000100
S1	X	X	X	X	S2	S3	X	X	S8	S4	X	X	X	00000111
S2	X	X	X	X	S2	S3	X	X	S8	S4	X	X	X	00001011
S3	X	X	X	X	X	X	S3	X	X	X	S8	S4	X	00001010
S4	S5	S5	S5	S5	S5	S5	S5	S5	S5	S5	S5	S5	S5	00001100
S5	S6	S6	S6	S6	S6	S6	S6	S6	S6	S6	S6	S6	S6	00001101
S6	S7	S7	S7	S7	S7	S7	S7	S7	S7	S7	S7	S7	S7	00001110
S7	S8	S8	S8	S8	S8	S8	S8	S8	S8	S8	S8	S8	S8	01001100
S8	S9	S9	S9	S9	S9	S9	S9	S9	S9	S9	S9	S9	S9	00000000
S9	S10	S10	S10	S10	S10	S10	S10	S10	S10	S10	S10	S10	S10	00000001
S10	X	X	S12	S11	X	X	X	X	X	X	X	X	X	00001001
S11	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	01001000
S12	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	S13	01011000
S13	S14	S14	S14	S14	S14	S14	S14	S14	S14	S14	S14	S14	S14	01001001
S14	S15	S15	S15	S15	S15	S15	S15	S15	S15	S15	S15	S15	S15	11001000
S15	S16	S16	S16	S16	S16	S16	S16	S16	S16	S16	S16	S16	S16	11000000
S16	X	X	X	X	S17	S18	X	X	X	X	X	X	S19	10000000
S17	X	X	X	X	S17	S18	X	X	X	X	X	X	S19	10001001
S18	X	X	X	X	X	X	S18	S19	X	X	X	X	X	10001010
S19	X	X	X	S20	S23	X	X	X	X	X	X	X	X	10001100
S20	S21	S21	S21	S21	S21	S21	S21	S21	S21	S21	S21	S21	S21	10101100
S21	S22	S22	S22	S22	S22	S22	S22	S22	S22	S22	S22	S22	S22	10101101
S22	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	11101100
S23	S24	S24	S24	S24	S24	S24	S24	S24	S24	S24	S24	S24	S24	10001101
S24	S25	S25	S25	S25	S25	S25	S25	S25	S25	S25	S25	S25	S25	10001110
S25	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	S0	11001100

Input Terms 1–13

1. start/
2. start
3. ddin
4. ddin/
5. b8wait
6. b8wait/*b8cwait
7. b8cwait
8. b8wait/
9. b8wait/*b8cwait/*hbe*A0/
10. b8wait/*b8cwait/*(hbe/ + a0)
11. b8cwait/*hbe*a0/
12. b8cwait*(hbe/ + a0)
13. b8wait/*b8cwait/

FIGURE 3. State Table



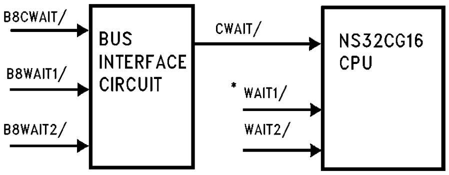
TL/EE/10494–2

*Even word bus cycles are handled as two 8-bit transfers.

Note 1: Refers to the first byte transfer.

Note 2: Refers to the second byte transfer.

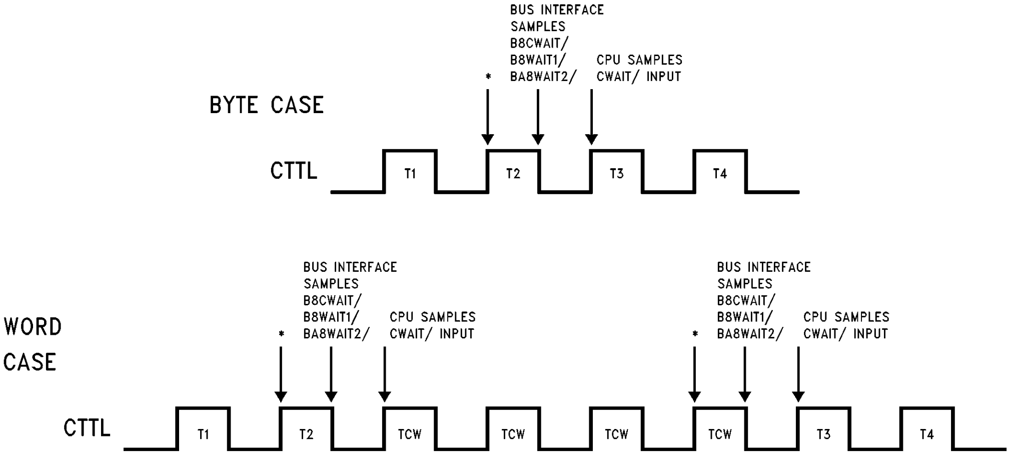
FIGURE 4. Bus Cycle Combinations



TL/EE/10494–3

*WAIT/n inputs available for system designer.

FIGURE 5. Wait Request Structure


BYTE CASE
CTTL
T1
T2
T3
T4
*
BUS INTERFACE SAMPLES B8CWAIT/ B8WAIT1/ BA8WAIT2/
CPU SAMPLES CWAIT/ INPUT
WORD CASE
CTTL
T1
T2
TCW
TCW
TCW
TCW
T3
T4
*
BUS INTERFACE SAMPLES B8CWAIT/ B8WAIT1/ BA8WAIT2/
CPU SAMPLES CWAIT/ INPUT
*
BUS INTERFACE SAMPLES B8CWAIT/ B8WAIT1/ BA8WAIT2/
CPU SAMPLES CWAIT/ INPUT

TL/EE/10494–4

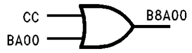
*Recommend wait requests are synchronized with rising edge CTTL in T2 to be stable for bus interface sample.

FIGURE 6. Bus Interface WAIT Sampling

Instruction Type	Cycle Type	CPU Clocks (CTTL)	Bus Interface Clocks (CTTL)
Even Byte	Even Byte	4	4
Odd Byte	Odd Byte	4	4
Even Word	Even Word	4	8
Odd Word	Odd Byte Even Byte	8	8*
Even Double Word	Even Word, Even Word	8	16
Odd Double Word	Odd Byte, Even Word, Even Byte	12	16*

***Note:** Odd alignment does not decrease bus interface efficiency

FIGURE 7. Clocks/Bus Cycle Performance

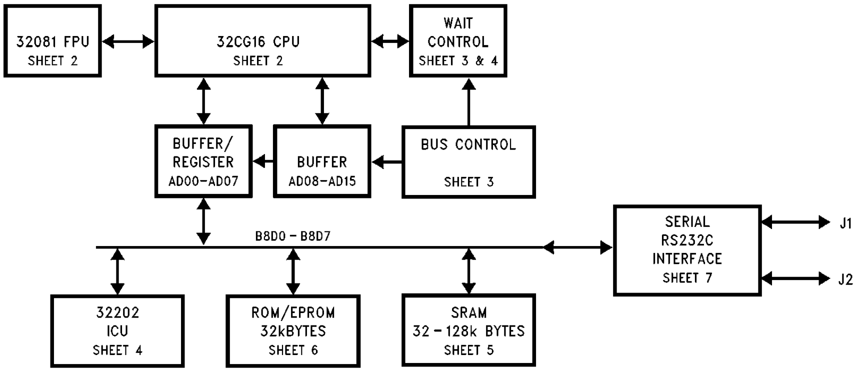


TL/EE/10494–5

FIGURE 8. Address Bit 0 for 8-Bit System

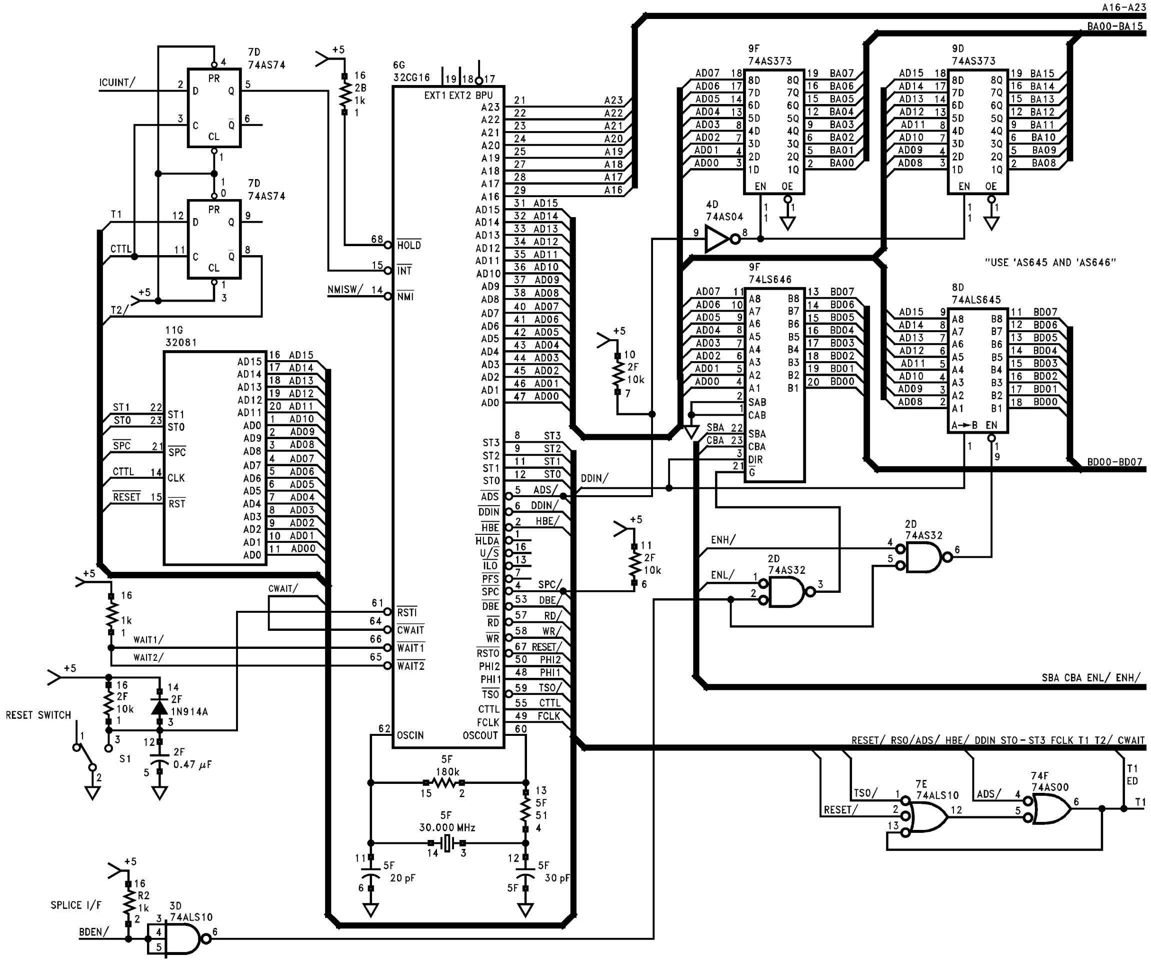
8.0 SCHEMATIC

32CG16 8-Bit Bus Functional Block Diagram



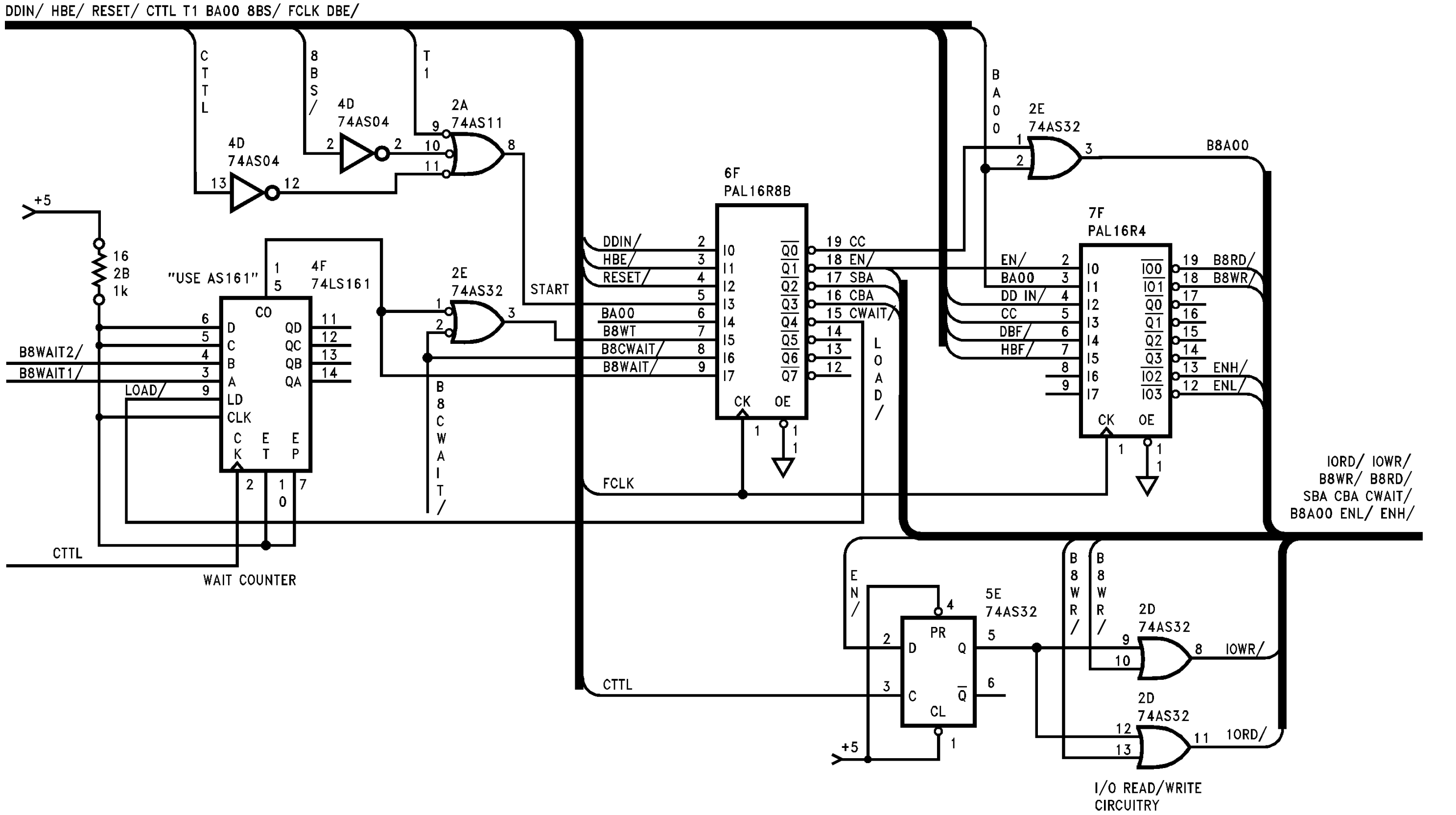
TL/EE/10494–6

CPU and Buffering

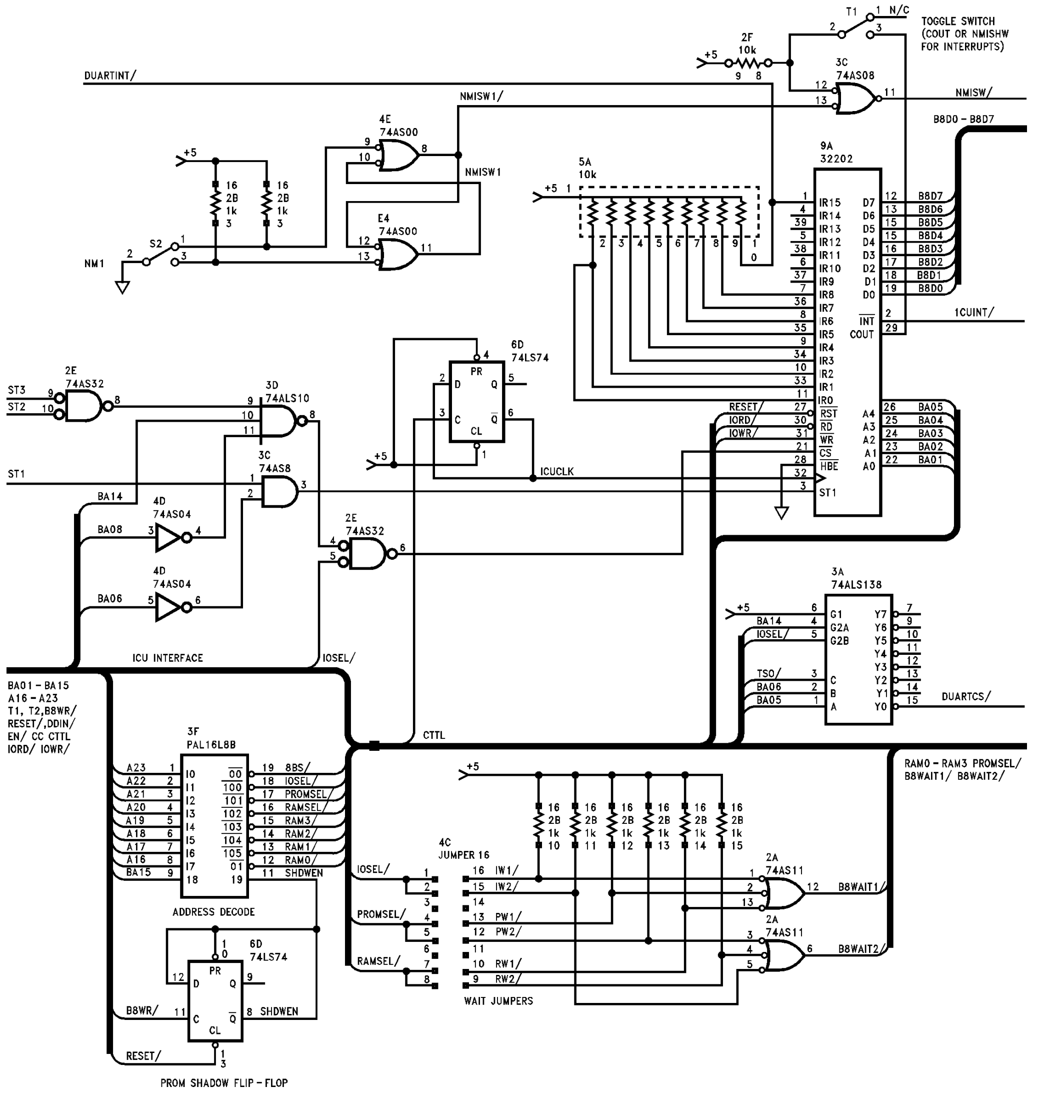


TL/EE/10494-7

Bus Interface Circuit

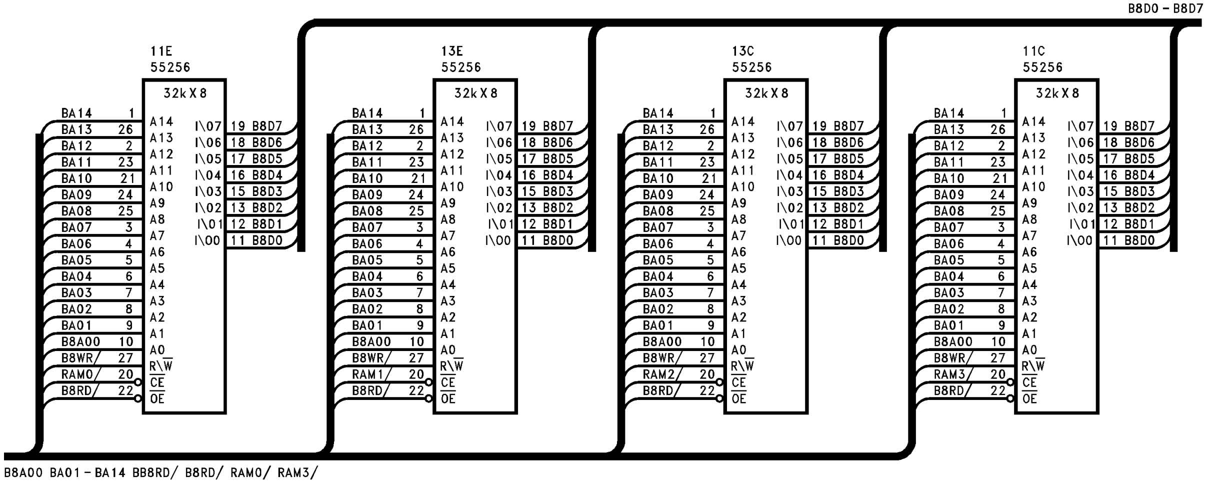


TL/EE/10494–8


T1
1 N/C
2
3
TOGGLE SWITCH
(COUT OR NMISHW
FOR INTERRUPTS)
+5
2F
10k
9 8
3C
74AS08
12
13
11
NMISW/
DUARTINT/
NMISW1/
B8D0 - B8D7
4E
74AS00
9
10
8
NMISW1
+5
16
2B
1k
3
E4
74AS00
12
13
11
S2
NM1
5A
10k
+5 1
2 3 4 5 6 7 8 9 1 0
9A
32202
IR15
IR14
IR13
IR12
IR11
IR10
IR9
IR8
IR7
IR6
IR5
IR4
IR3
IR2
IR1
IR0
RST
RD
WR
CS
HBE
ST1
D7
D6
D5
D4
D3
D2
D1
D0
INT
COUT
A4
A3
A2
A1
A0
B8D7
B8D6
B8D5
B8D4
B8D3
B8D2
B8D1
B8D0
ICUINT/
BA05
BA04
BA03
BA02
BA01
RESET/
IORD/
IOWR/
6D
74LS74
PR
D
Q
C
Q
CL
ICUCLK
2E
74AS32
ST3
ST2
3D
74ALS10
ST1
3C
74AS8
BA14
4D
74AS04
BA08
BA06
2E
74AS32
ICU INTERFACE
IOSEL/
BA01 - BA15
A16 - A23
T1, T2,B8WR/
RESET/,DDIN/
EN/ CC CTTL
IORD/ IOWR/
CTTL
3A
74ALS138
+5
BA14
IOSEL/
TSO/
BA06
BA05
G1
G2A
G2B
C
B
A
Y7
Y6
Y5
Y4
Y3
Y2
Y1
Y0
DUARTCS/
RAM0 - RAM3 PROMSEL/
B8WAIT1/ B8WAIT2/
3F
PAL16L8B
A23
A22
A21
A20
A19
A18
A17
A16
BA15
I0
I1
I2
I3
I4
I5
I6
I7
I8
O0
IO0
IO1
IO2
IO3
IO4
IO5
O1
I9
8BS/
IOSEL/
PROMSEL/
RAMSEL/
RAM3/
RAM2/
RAM1/
RAM0/
SHDWEN
ADDRESS DECODE
6D
74LS74
PR
D
Q
C
Q
CL
B8WR/
RESET/
SHDWEN
PROM SHADOW FLIP-FLOP
+5
16
2B
1k
10
11
12
13
14
15
4C
JUMPER 16
IOSEL/
PROMSEL/
RAMSEL/
IW1/
IW2/
PW1/
PW2/
RW1/
RW2/
WAIT JUMPERS
2A
74AS11
B8WAIT1
2A
74AS11
B8WAIT2

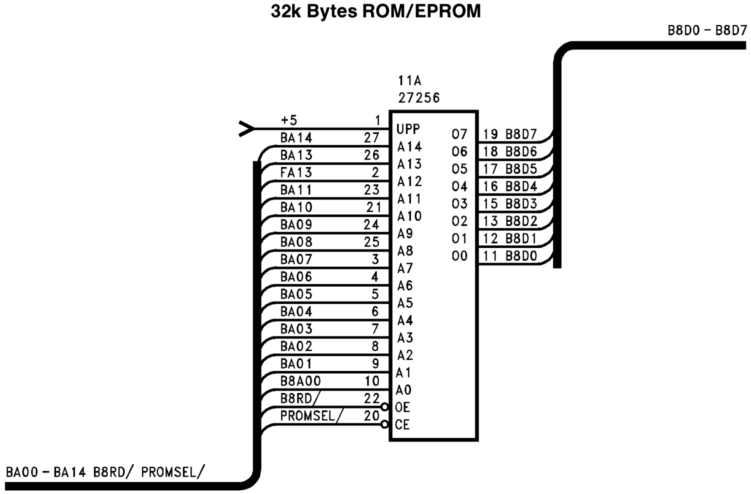
TL/EE/10494-9

32k–128k Byte Static RAM



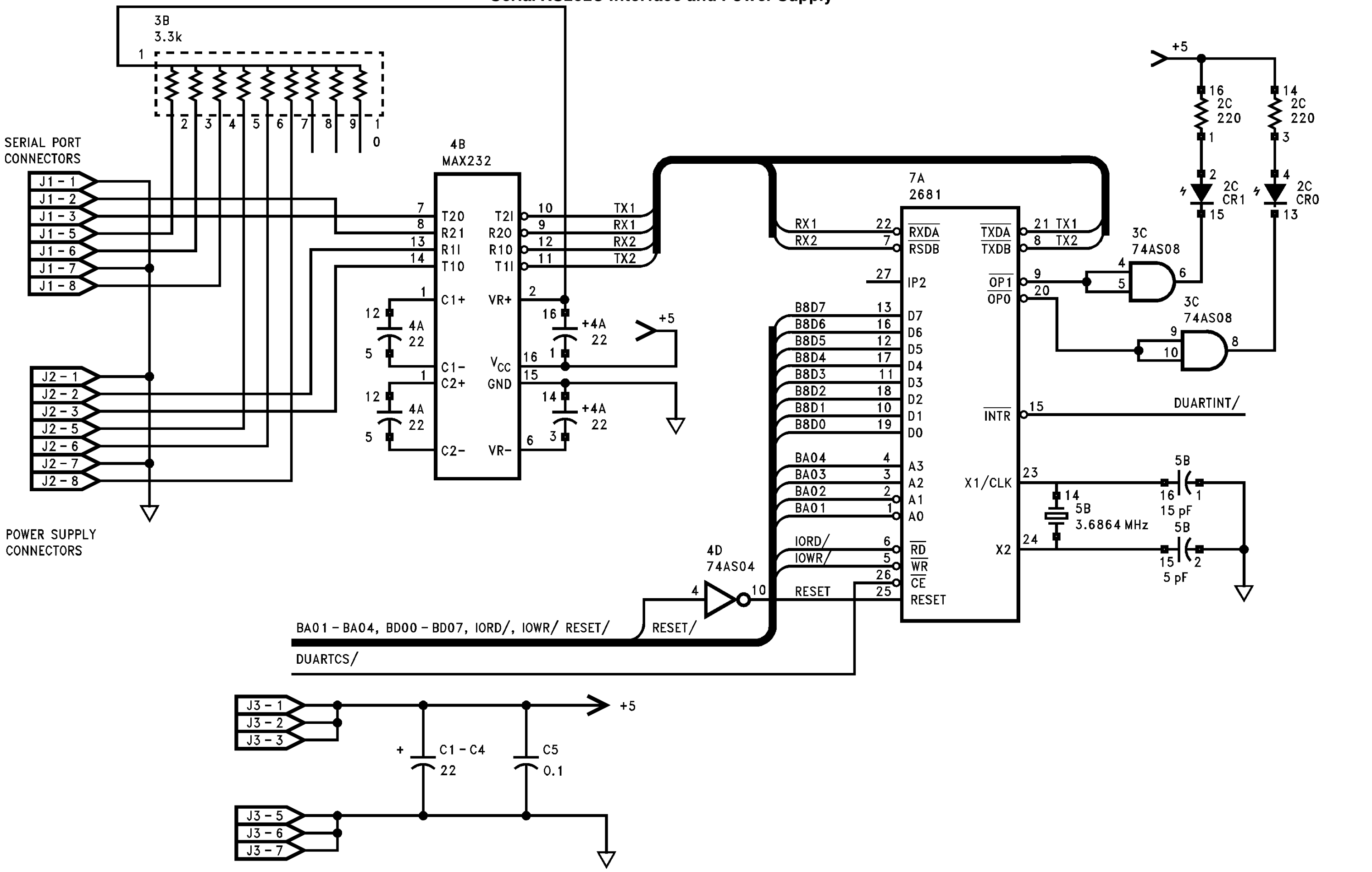
TL/EE/10494–10

32k Bytes ROM/EPROM



TL/EE/10494–11

Serial RS232C Interface and Power Supply



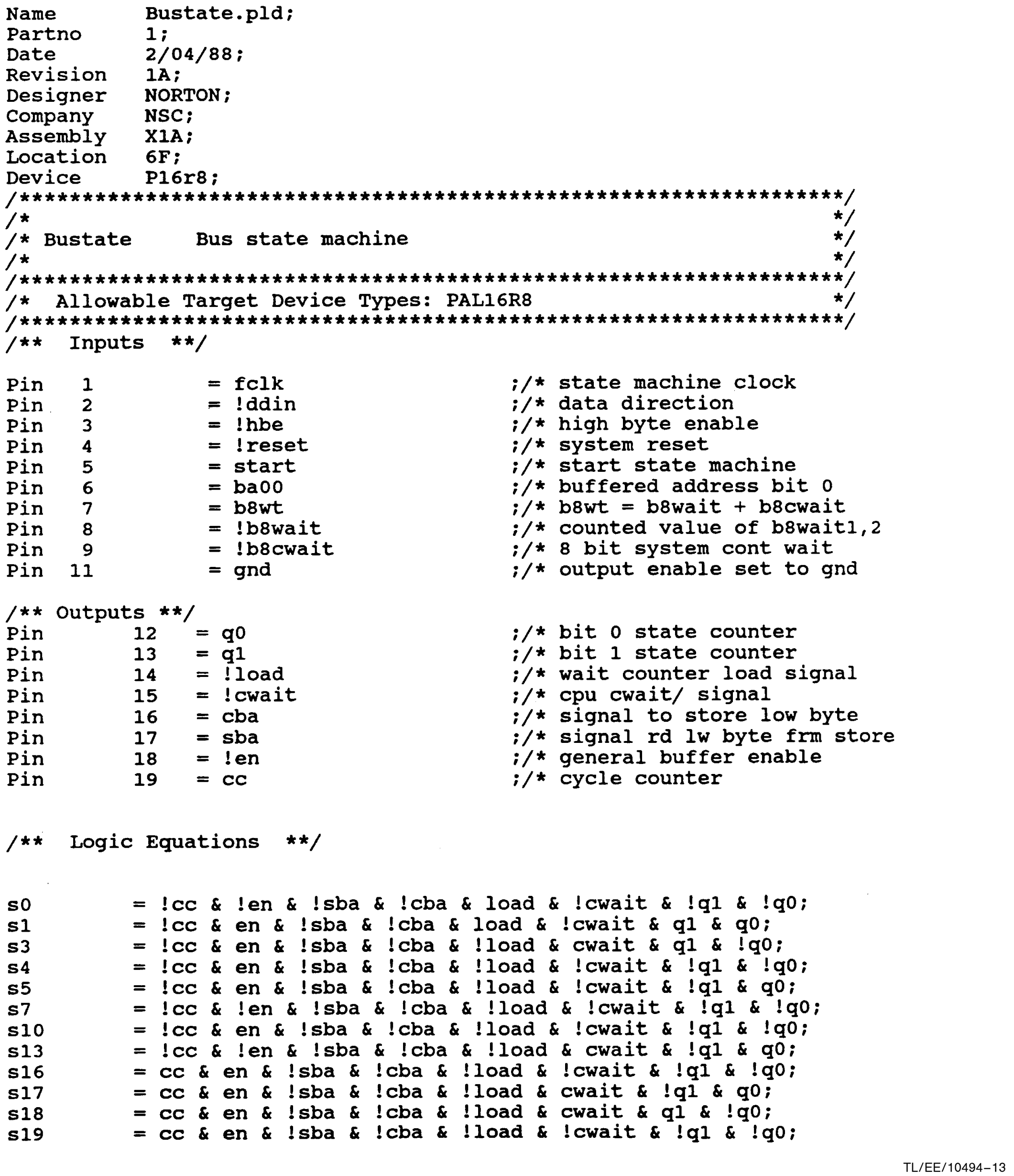
TL/EE/10494-12

Parts List BUS8-32CG16

11A	27256	A/06
11C	55256	A/05
11E	55256	A/05
11G	32081	A/02
13C	55256	A/05
13E	55256	A/05
2A	74AS11	A/04, B/04, C/03
2B	PRES	A/02, B/02, C/04, D/04, I/03, J/04, K/04, L/04, M/04, N/04
2C	HRES	A/07, B/07, C/07, D/07
2D	74AS32	A/02, B/02, C/03, D/03
2E	74AS32	A/03, B/04, C/04
2F	HRES	A/02, C/02, E/02, F/02, G/02, H/04
3A	74ALS138	A/04
3B	DRPAC10	A/07
3C	74AS08	A/04, B/07, C/07, D/04
3D	74ALS10	A/02, C/04
3F	PAL16L8B	A/04
4A	HCAP	A/07, C/07, E/07, G/07
4B	MAX232	A/07
4C	JUMPER16	A/04
4D	74AS04	A/03, B/04, C/04, D/02, E/07, F/03
4E	74AS00	A/03, B/02, C/04, D/04
4F	74AS161	A/03
5A	DRPAC10	A/04
5B	HCAP	A/07, A/07, B/07
5E	74AS74	A/03
5F	HCAP	A/02, B/02, D/02, E/02, F/02
6D	74LS74	A/04, B/04
6F	PAL16R8B	A/03
6G	32CG16	A/02, A/02
7A	2681	A/07
7D	74AS74	A/02, B/02
7F	PAL16R4	A/03
8D	74ALS645	A/02
8F	74LS646	A/02
9A	32202	A/04
9D	74AS373	A/02
9F	74AS373	A/02
C1–C4	DCAPE	A/07
C5 . . .	DCAP	A/07
J1	^^DB32F	C/07, C/07, C/07, C/07, C/07, C/07, C/07
J2	^^DB32F	C/07, C/07, C/07, C/07, C/07, C/07, C/07
J3	^^DCONN	C/07, C/07, C/07, C/07, C/07, C/07
S1	DSPDT	A/02
S2	DSPDT	A/04
T1	DSPDT	A/04

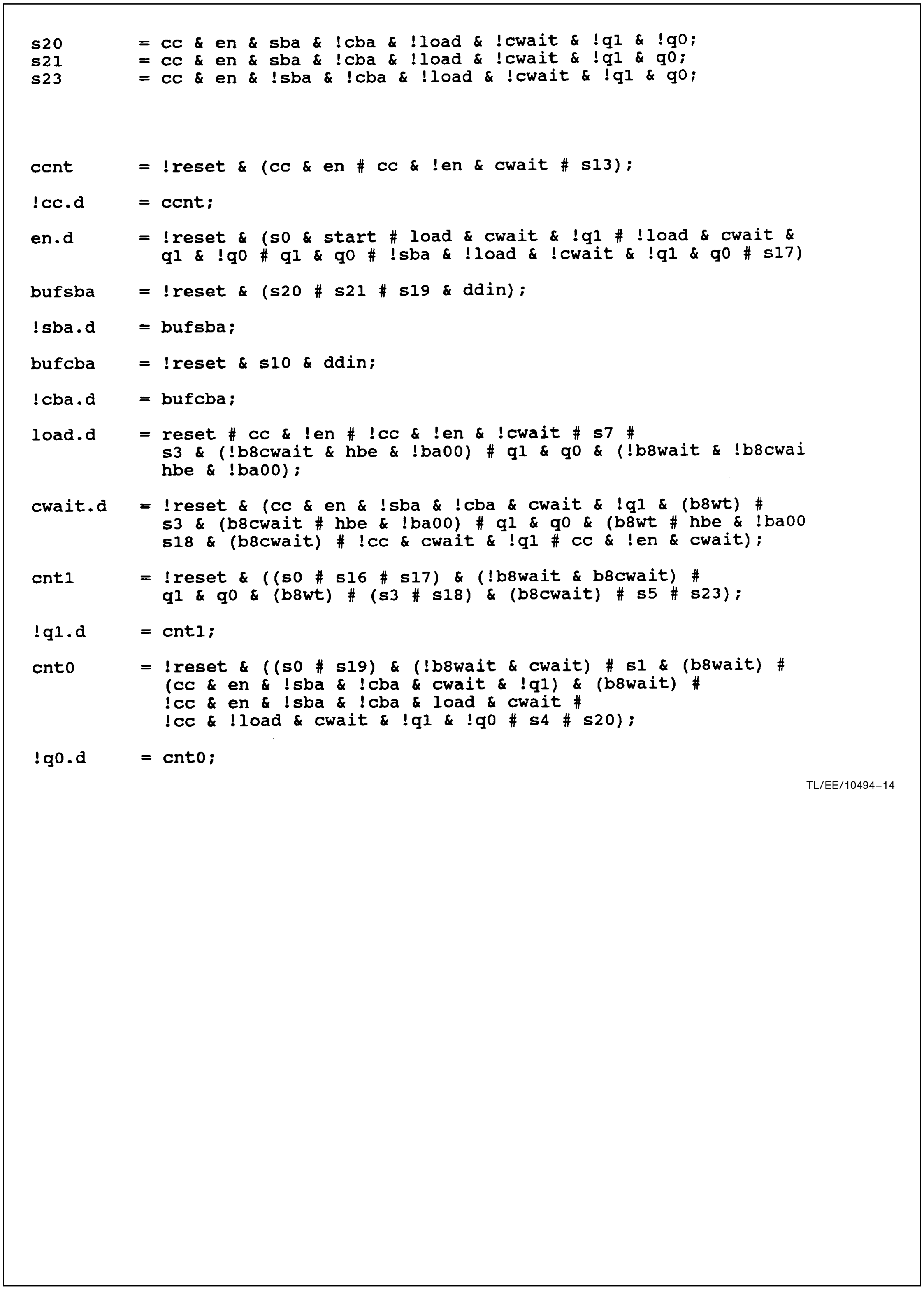
9.0 PAL® EQUATIONS

The 3 PALs implemented in the bus interface design are listed on the following pages. CUPL rev. 2.11c from Personal CAD Systems Inc. was used to create the files.



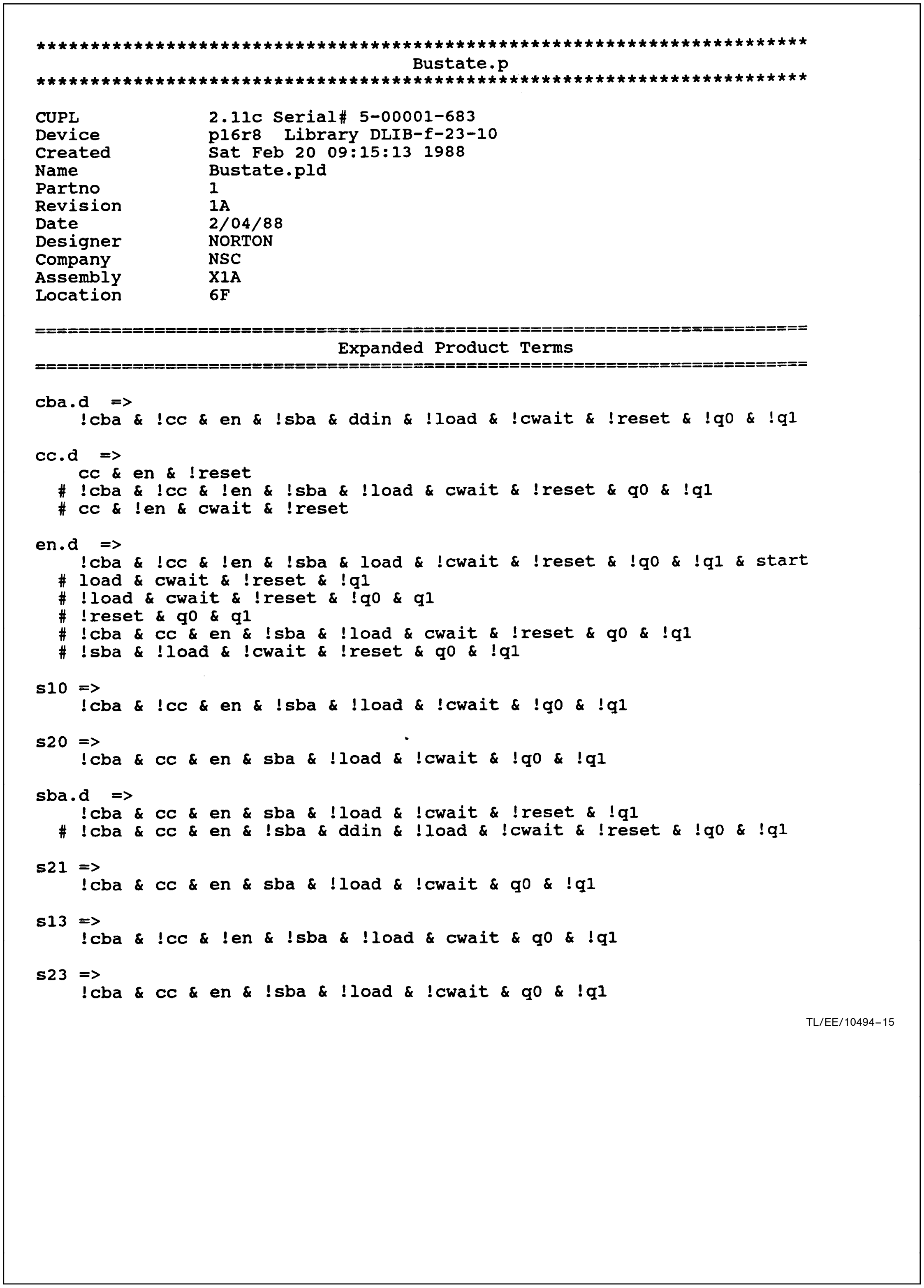
```
Name       Bustate.pld;
Partno     1;
Date       2/04/88;
Revision   1A;
Designer   NORTON;
Company    NSC;
Assembly   X1A;
Location   6F;
Device     P16r8;
/******************************************************************/
/*                                                                */
/* Bustate     Bus state machine                                  */
/*                                                                */
/******************************************************************/
/*  Allowable Target Device Types: PAL16R8                        */
/******************************************************************/
/**  Inputs  **/

Pin   1         = fclk                ;/* state machine clock
Pin   2         = !ddin               ;/* data direction
Pin   3         = !hbe                ;/* high byte enable
Pin   4         = !reset              ;/* system reset
Pin   5         = start               ;/* start state machine
Pin   6         = ba00                ;/* buffered address bit 0
Pin   7         = b8wt                ;/* b8wt = b8wait + b8cwait
Pin   8         = !b8wait             ;/* counted value of b8wait1,2
Pin   9         = !b8cwait            ;/* 8 bit system cont wait
Pin  11         = gnd                 ;/* output enable set to gnd

/** Outputs **/
Pin       12   = q0                   ;/* bit 0 state counter
Pin       13   = q1                   ;/* bit 1 state counter
Pin       14   = !load                ;/* wait counter load signal
Pin       15   = !cwait               ;/* cpu cwait/ signal
Pin       16   = cba                  ;/* signal to store low byte
Pin       17   = sba                  ;/* signal rd lw byte frm store
Pin       18   = !en                  ;/* general buffer enable
Pin       19   = cc                   ;/* cycle counter


/**  Logic Equations  **/


s0        = !cc & !en & !sba & !cba & load & !cwait & !q1 & !q0;
s1        = !cc & en & !sba & !cba & load & !cwait & q1 & q0;
s3        = !cc & en & !sba & !cba & !load & cwait & q1 & !q0;
s4        = !cc & en & !sba & !cba & !load & !cwait & !q1 & !q0;
s5        = !cc & en & !sba & !cba & !load & !cwait & !q1 & q0;
s7        = !cc & !en & !sba & !cba & !load & !cwait & !q1 & !q0;
s10       = !cc & en & !sba & !cba & !load & !cwait & !q1 & !q0;
s13       = !cc & !en & !sba & !cba & !load & cwait & !q1 & q0;
s16       = cc & en & !sba & !cba & !load & !cwait & !q1 & !q0;
s17       = cc & en & !sba & !cba & !load & cwait & !q1 & q0;
s18       = cc & en & !sba & !cba & !load & cwait & q1 & !q0;
s19       = cc & en & !sba & !cba & !load & !cwait & !q1 & !q0;
```

TL/EE/10494–13



```
s20         = cc & en & sba & !cba & !load & !cwait & !q1 & !q0;
s21         = cc & en & sba & !cba & !load & !cwait & !q1 & q0;
s23         = cc & en & !sba & !cba & !load & !cwait & !q1 & q0;



ccnt        = !reset & (cc & en # cc & !en & cwait # s13);

!cc.d       = ccnt;

en.d        = !reset & (s0 & start # load & cwait & !q1 # !load & cwait &
              q1 & !q0 # q1 & q0 # !sba & !load & !cwait & !q1 & q0 # s17)

bufsba      = !reset & (s20 # s21 # s19 & ddin);

!sba.d      = bufsba;

bufcba      = !reset & s10 & ddin;

!cba.d      = bufcba;

load.d      = reset # cc & !en # !cc & !en & !cwait # s7 #
              s3 & (!b8cwait & hbe & !ba00) # q1 & q0 & (!b8wait & !b8cwai
              hbe & !ba00);

cwait.d     = !reset & (cc & en & !sba & !cba & cwait & !q1 & (b8wt) #
              s3 & (b8cwait # hbe & !ba00) # q1 & q0 & (b8wt # hbe & !ba00
              s18 & (b8cwait) # !cc & cwait & !q1 # cc & !en & cwait);

cnt1        = !reset & ((s0 # s16 # s17) & (!b8wait & b8cwait) #
              q1 & q0 & (b8wt) # (s3 # s18) & (b8cwait) # s5 # s23);

!q1.d       = cnt1;

cnt0        = !reset & ((s0 # s19) & (!b8wait & cwait) # s1 & (b8wait) #
              (cc & en & !sba & !cba & cwait & !q1) & (b8wait) #
              !cc & en & !sba & !cba & load & cwait #
              !cc & !load & cwait & !q1 & !q0 # s4 # s20);

!q0.d       = cnt0;
```

TL/EE/10494–14

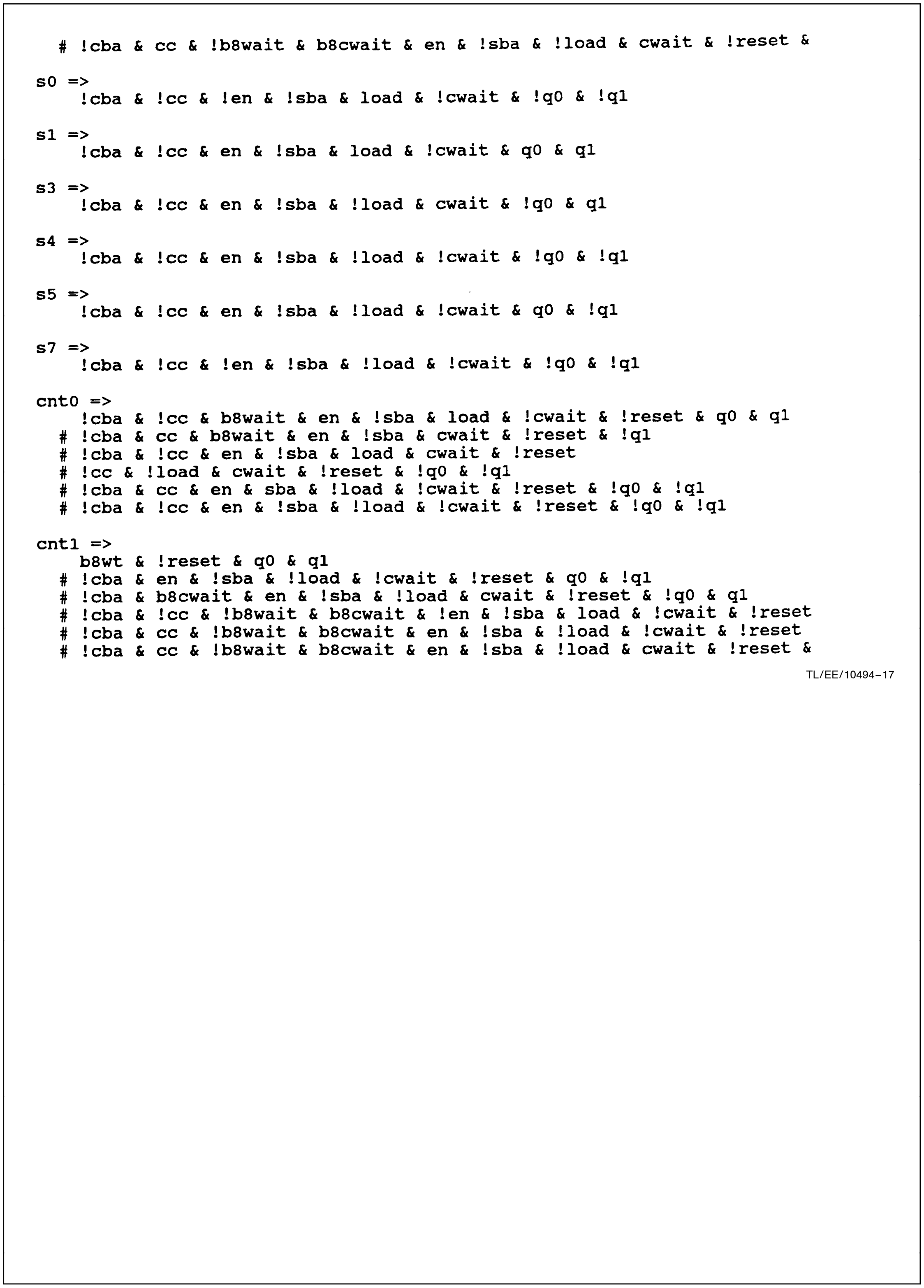


```
***********************************************************************
                            Bustate.p
***********************************************************************

CUPL           2.11c Serial# 5-00001-683
Device         p16r8  Library DLIB-f-23-10
Created        Sat Feb 20 09:15:13 1988
Name           Bustate.pld
Partno         1
Revision       1A
Date           2/04/88
Designer       NORTON
Company        NSC
Assembly       X1A
Location       6F

=======================================================================
                       Expanded Product Terms
=======================================================================

cba.d  =>
    !cba & !cc & en & !sba & ddin & !load & !cwait & !reset & !q0 & !q1

cc.d  =>
    cc & en & !reset
  # !cba & !cc & !en & !sba & !load & cwait & !reset & q0 & !q1
  # cc & !en & cwait & !reset

en.d  =>
    !cba & !cc & !en & !sba & load & !cwait & !reset & !q0 & !q1 & start
  # load & cwait & !reset & !q1
  # !load & cwait & !reset & !q0 & q1
  # !reset & q0 & q1
  # !cba & cc & en & !sba & !load & cwait & !reset & q0 & !q1
  # !sba & !load & !cwait & !reset & q0 & !q1

s10 =>
    !cba & !cc & en & !sba & !load & !cwait & !q0 & !q1

s20 =>
    !cba & cc & en & sba & !load & !cwait & !q0 & !q1

sba.d  =>
    !cba & cc & en & sba & !load & !cwait & !reset & !q1
  # !cba & cc & en & !sba & ddin & !load & !cwait & !reset & !q0 & !q1

s21 =>
    !cba & cc & en & sba & !load & !cwait & q0 & !q1

s13 =>
    !cba & !cc & !en & !sba & !load & cwait & q0 & !q1

s23 =>
    !cba & cc & en & !sba & !load & !cwait & q0 & !q1
```

TL/EE/10494–15

```
s16 =>
    !cba & cc & en & !sba & !load & !cwait & !q0 & !q1

s17 =>
    !cba & cc & en & !sba & !load & cwait & q0 & !q1

s18 =>
    !cba & cc & en & !sba & !load & cwait & !q0 & q1

s19 =>
    !cba & cc & en & !sba & !load & !cwait & !q0 & !q1

load.d  =>
    reset
  # cc & !en
  # !cc & !en & !cwait
  # !ba00 & !cba & !cc & !b8cwait & hbe & en & !sba & !load & cwait & !q
  # !ba00 & !b8wait & !b8cwait & hbe & q0 & q1

bufcba =>
    !cba & !cc & en & !sba & ddin & !load & !cwait & !reset & !q0 & !q1

ccnt =>
    cc & en & !reset
  # !cba & !cc & !en & !sba & !load & cwait & !reset & q0 & !q1
  # cc & !en & cwait & !reset

bufsba =>
    !cba & cc & en & sba & !load & !cwait & !reset & !q1
  # !cba & cc & en & !sba & ddin & !load & !cwait & !reset & !q0 & !q1

cwait.d  =>
    b8wt & !cba & cc & en & !sba & cwait & !reset & !q1
  # !cba & b8cwait & en & !sba & !load & cwait & !reset & !q0 & q1
  # cc & !en & cwait & !reset
  # !cc & cwait & !reset & !q1
  # !ba00 & hbe & !reset & q0 & q1
  # b8wt & !reset & q0 & q1
  # !ba00 & !cba & !cc & hbe & en & !sba & !load & cwait & !reset & !q0

q0.d  =>
    !cba & !cc & b8wait & en & !sba & load & !cwait & !reset & q0 & q1
  # !cba & cc & b8wait & en & !sba & cwait & !reset & !q1
  # !cba & !cc & en & !sba & load & cwait & !reset
  # !cc & !load & cwait & !reset & !q0 & !q1
  # !cba & cc & en & sba & !load & !cwait & !reset & !q0 & !q1
  # !cba & !cc & en & !sba & !load & !cwait & !reset & !q0 & !q1

q1.d  =>
    b8wt & !reset & q0 & q1
  # !cba & en & !sba & !load & !cwait & !reset & q0 & !q1
  # !cba & b8cwait & en & !sba & !load & cwait & !reset & !q0 & q1
  # !cba & !cc & !b8wait & b8cwait & !en & !sba & load & !cwait & !reset
  # !cba & cc & !b8wait & b8cwait & en & !sba & !load & !cwait & !reset
```

TL/EE/10494–16

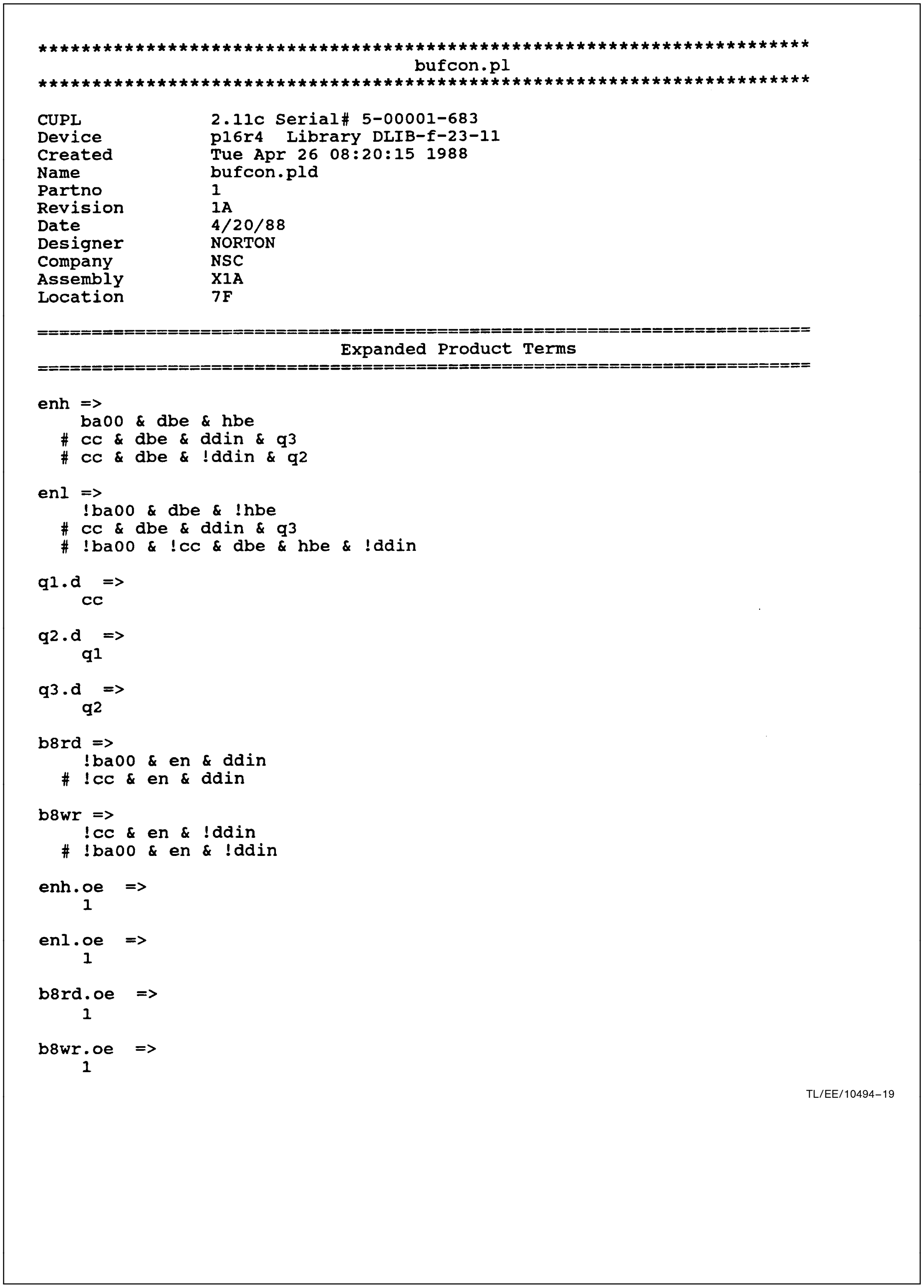


```
  # !cba & cc & !b8wait & b8cwait & en & !sba & !load & cwait & !reset &

s0 =>
    !cba & !cc & !en & !sba & load & !cwait & !q0 & !q1

s1 =>
    !cba & !cc & en & !sba & load & !cwait & q0 & q1

s3 =>
    !cba & !cc & en & !sba & !load & cwait & !q0 & q1

s4 =>
    !cba & !cc & en & !sba & !load & !cwait & !q0 & !q1

s5 =>
    !cba & !cc & en & !sba & !load & !cwait & q0 & !q1

s7 =>
    !cba & !cc & !en & !sba & !load & !cwait & !q0 & !q1

cnt0 =>
    !cba & !cc & b8wait & en & !sba & load & !cwait & !reset & q0 & q1
  # !cba & cc & b8wait & en & !sba & cwait & !reset & !q1
  # !cba & !cc & en & !sba & load & cwait & !reset
  # !cc & !load & cwait & !reset & !q0 & !q1
  # !cba & cc & en & sba & !load & !cwait & !reset & !q0 & !q1
  # !cba & !cc & en & !sba & !load & !cwait & !reset & !q0 & !q1

cnt1 =>
    b8wt & !reset & q0 & q1
  # !cba & en & !sba & !load & !cwait & !reset & q0 & !q1
  # !cba & b8cwait & en & !sba & !load & cwait & !reset & !q0 & q1
  # !cba & !cc & !b8wait & b8cwait & !en & !sba & load & !cwait & !reset
  # !cba & cc & !b8wait & b8cwait & en & !sba & !load & !cwait & !reset
  # !cba & cc & !b8wait & b8cwait & en & !sba & !load & cwait & !reset &
```

TL/EE/10494-17

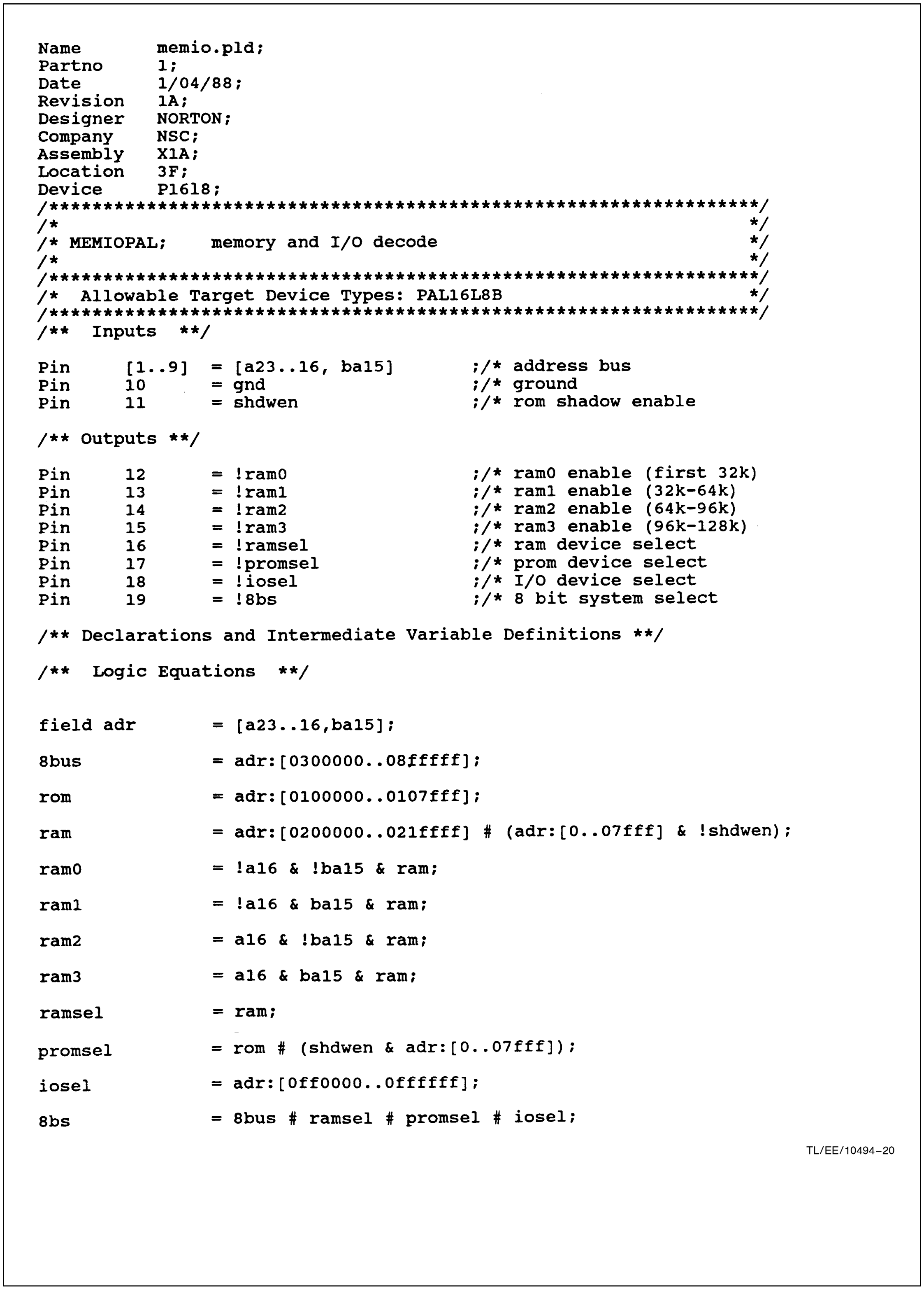
```
Name       bufcon.pld;
Partno     1;
Date       4/20/88;
Revision   1A;
Designer   NORTON;
Company    NSC;
Assembly   X1A;
Location   7F;
Device     P16r4;
/*****************************************************************/
/*                                                               */
/* BUFCON;      bus control pal                                  */
/*                                                               */
/*****************************************************************/
/*  Allowable Target Device Types: PAL16r4B                      */
/*****************************************************************/
/**  Inputs  **/
Pin     1       = fclk          ;/* counter clock
Pin     2       = !en           ;/* general buffer enable
Pin     3       = ba00          ;/* address bit 0
Pin     4       = !ddin         ;/* data direction
Pin     5       = cc            ;/* cycle count
Pin     6       = !dbe          ;/* cpu data buffer enable
Pin     7       = !hbe          ;/* cpu high byte enable

/**  Outputs  **/
Pin    12       = !enl          ;/* low byte buffer enable
Pin    13       = !enh          ;/* high byte buffer enable
Pin    14       = !q3           ;/* counter bit 3
Pin    15       = !q2           ;/* counter bit 2
Pin    16       = !q1           ;/* counter bit1
Pin    18       = !b8rd         ;/*system read strobe
Pin    19       = !b8wr         ;/*system write strobe


/** Declarations and Intermediate Variable Definitions **/

/**  Logic Equations  **/

q1.d = cc;

q2.d = q1;

q3.d = q2;

enl  = dbe & !hbe & !ba00 # cc & dbe & q3 & ddin # !cc & dbe & !ddin &
       hbe & !ba00;

enh  = dbe & hbe & ba00 # cc & dbe & (q3 & ddin # q2 & !ddin);

b8wr = !cc & en & !ddin # !ba00 & !ddin & en;

b8rd = !ba00 & ddin & en # !cc & en & ddin;
```

TL/EE/10494–18



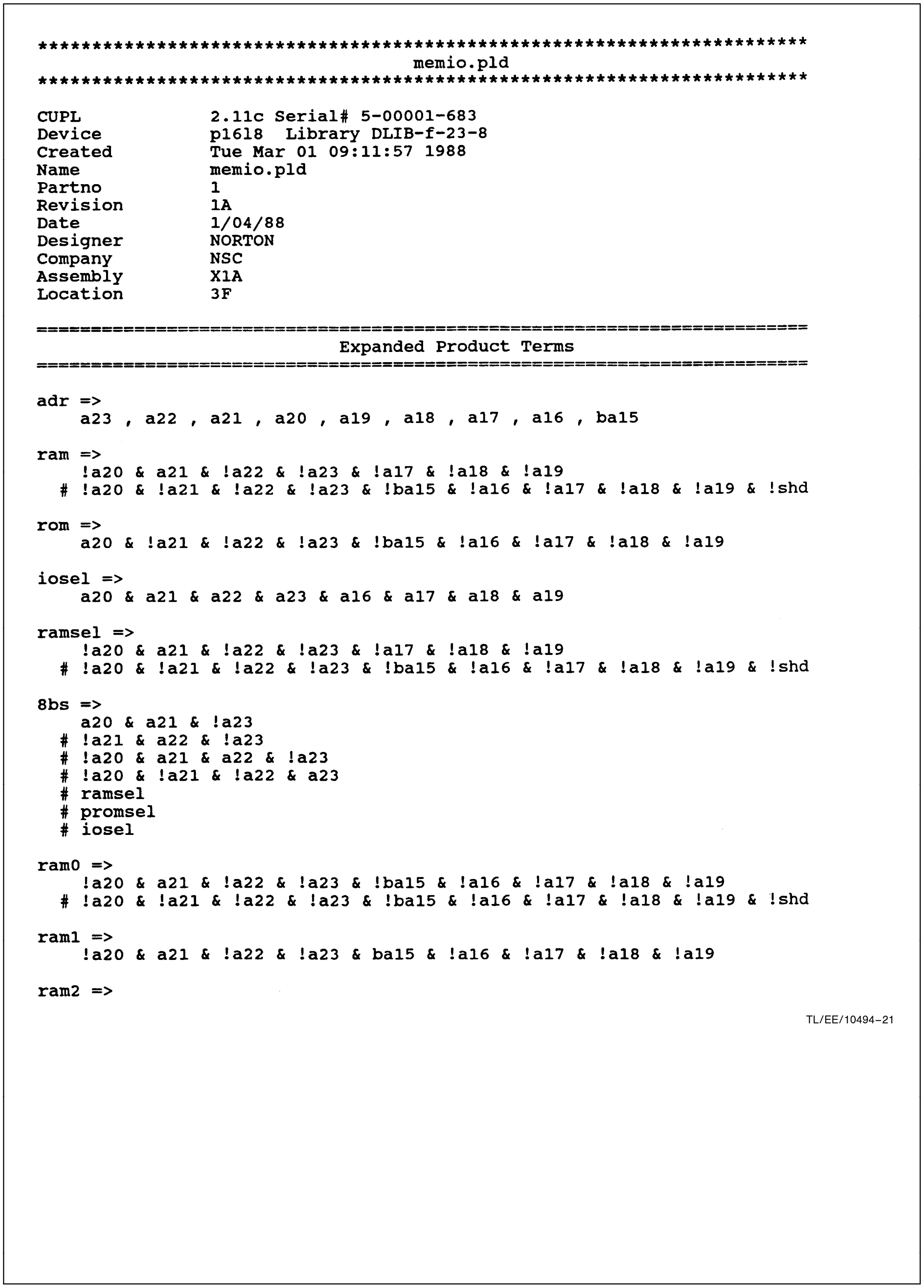
```
**************************************************************************
                                   bufcon.pl
**************************************************************************

CUPL            2.11c Serial# 5-00001-683
Device          p16r4  Library DLIB-f-23-11
Created         Tue Apr 26 08:20:15 1988
Name            bufcon.pld
Partno          1
Revision        1A
Date            4/20/88
Designer        NORTON
Company         NSC
Assembly        X1A
Location        7F

==========================================================================
                            Expanded Product Terms
==========================================================================

enh =>
    ba00 & dbe & hbe
  # cc & dbe & ddin & q3
  # cc & dbe & !ddin & q2

enl =>
    !ba00 & dbe & !hbe
  # cc & dbe & ddin & q3
  # !ba00 & !cc & dbe & hbe & !ddin

q1.d  =>
    cc

q2.d  =>
    q1

q3.d  =>
    q2

b8rd =>
    !ba00 & en & ddin
  # !cc & en & ddin

b8wr =>
    !cc & en & !ddin
  # !ba00 & en & !ddin

enh.oe  =>
    1

enl.oe  =>
    1

b8rd.oe  =>
    1

b8wr.oe  =>
    1
```

TL/EE/10494-19



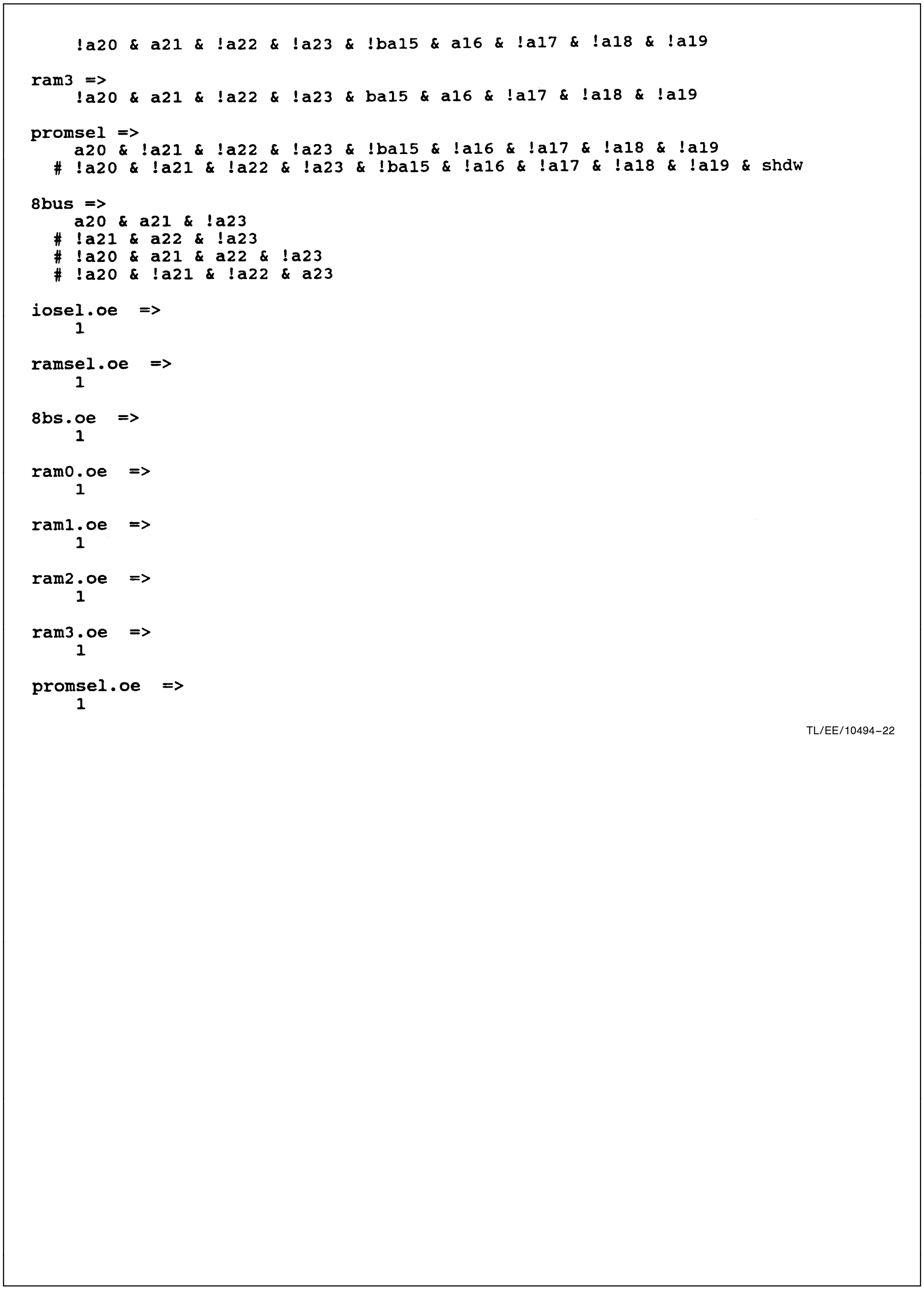
```
Name       memio.pld;
Partno     1;
Date       1/04/88;
Revision   1A;
Designer   NORTON;
Company    NSC;
Assembly   X1A;
Location   3F;
Device     P16l8;
/*****************************************************************/
/*                                                               */
/* MEMIOPAL;    memory and I/O decode                            */
/*                                                               */
/*****************************************************************/
/*  Allowable Target Device Types: PAL16L8B                      */
/*****************************************************************/
/**  Inputs  **/

Pin     [1..9]  = [a23..16, ba15]       ;/* address bus
Pin     10      = gnd                   ;/* ground
Pin     11      = shdwen                ;/* rom shadow enable

/** Outputs **/

Pin     12      = !ram0                 ;/* ram0 enable (first 32k)
Pin     13      = !ram1                 ;/* ram1 enable (32k-64k)
Pin     14      = !ram2                 ;/* ram2 enable (64k-96k)
Pin     15      = !ram3                 ;/* ram3 enable (96k-128k)
Pin     16      = !ramsel               ;/* ram device select
Pin     17      = !promsel              ;/* prom device select
Pin     18      = !iosel                ;/* I/O device select
Pin     19      = !8bs                  ;/* 8 bit system select

/** Declarations and Intermediate Variable Definitions **/

/**  Logic Equations  **/


field adr       = [a23..16,ba15];

8bus            = adr:[0300000..08fffff];

rom             = adr:[0100000..0107fff];

ram             = adr:[0200000..021ffff] # (adr:[0..07fff] & !shdwen);

ram0            = !a16 & !ba15 & ram;

ram1            = !a16 & ba15 & ram;

ram2            = a16 & !ba15 & ram;

ram3            = a16 & ba15 & ram;

ramsel          = ram;

promsel         = rom # (shdwen & adr:[0..07fff]);

iosel           = adr:[0ff0000..0ffffff];

8bs             = 8bus # ramsel # promsel # iosel;
```

TL/EE/10494–20



```
************************************************************************
                                   memio.pld
************************************************************************

CUPL            2.11c Serial# 5-00001-683
Device          p1618  Library DLIB-f-23-8
Created         Tue Mar 01 09:11:57 1988
Name            memio.pld
Partno          1
Revision        1A
Date            1/04/88
Designer        NORTON
Company         NSC
Assembly        X1A
Location        3F

========================================================================
                            Expanded Product Terms
========================================================================

adr =>
    a23 , a22 , a21 , a20 , a19 , a18 , a17 , a16 , ba15

ram =>
    !a20 & a21 & !a22 & !a23 & !a17 & !a18 & !a19
  # !a20 & !a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19 & !shd

rom =>
    a20 & !a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19

iosel =>
    a20 & a21 & a22 & a23 & a16 & a17 & a18 & a19

ramsel =>
    !a20 & a21 & !a22 & !a23 & !a17 & !a18 & !a19
  # !a20 & !a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19 & !shd

8bs =>
    a20 & a21 & !a23
  # !a21 & a22 & !a23
  # !a20 & a21 & a22 & !a23
  # !a20 & !a21 & !a22 & a23
  # ramsel
  # promsel
  # iosel

ram0 =>
    !a20 & a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19
  # !a20 & !a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19 & !shd

ram1 =>
    !a20 & a21 & !a22 & !a23 & ba15 & !a16 & !a17 & !a18 & !a19

ram2 =>
```

TL/EE/10494–21



```
    !a20 & a21 & !a22 & !a23 & !ba15 & a16 & !a17 & !a18 & !a19

ram3 =>
    !a20 & a21 & !a22 & !a23 & ba15 & a16 & !a17 & !a18 & !a19

promsel =>
    a20 & !a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19
  # !a20 & !a21 & !a22 & !a23 & !ba15 & !a16 & !a17 & !a18 & !a19 & shdw

8bus =>
    a20 & a21 & !a23
  # !a21 & a22 & !a23
  # !a20 & a21 & a22 & !a23
  # !a20 & !a21 & !a22 & a23

iosel.oe  =>
    1

ramsel.oe  =>
    1

8bs.oe  =>
    1

ram0.oe  =>
    1

ram1.oe  =>
    1

ram2.oe  =>
    1

ram3.oe  =>
    1

promsel.oe  =>
    1
```

TL/EE/10494–22

AN-635 Eight-Bit Bus Interface for the NS32CG16; NS32CG16 Applications Note 6

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
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

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 Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.