HEWLETT PACKARD

CONTENTS

HEWLETT PACKARD

BURNNI
BASIC STANDARD I/O

External Reference Specification

Martin Speer
Bruce Bergmann
Hewlett-Packard Co.
Fort Collins System Division
February 28, 1990

1. General

BURNNI is one of two core I/O ASICs developed to save PC board area and cost by integrating most of the "glue logic" surrounding the Series 300 core I/O plus some of the core I/O itself. In addition to this, some new I/O was added to improve system adaptability and improve the ability for a user to configure the system. The enhancement which was added to improve the adaptability of the system was a programmable BUS Error timer. EEPROM support was added in order to eliminate configuration switches and for the storage of the LAN Lanic Address. This allows users to easily change the configuration of their system through the use of configuration software(BOOT ROM resident) and not by opening up the system and trying to figure out switch settings(although this will still be necessary for some optional I/O cards).

1.1 Features

This is the fill list of functions performed by BURNNI:

- Enhanced Programmable Timer Module(PTM) software compatible with Motorola MC6840. Interrupts tied to Interrupt Level 6.
- LAN support logic. NOVRAM replaced by 64 nibbles of RAM within BURNNI which is initialized by the BOOT ROM. Dual Port Control of LAN Buffer RAM with enhanced performance via 32 bit RAM interface implementation. Interrupt level and select code switches eliminated through "hidden" registers which are configured by BOOTROM using EEPROM data.
- Address Decode, "DTACK" generation, and Chip Select for interfacing a BOOT ROM to the system. The BOOT ROM's address space can be doubled by setting a bit in one of BURNNI's control registers.
- Address Decode, "DTACK" generation, and write strobe for interfacing System Status LEDs to the system.
- Address Decode, "DTACK" generation, and necessary control signals provided to connect an INTEL 8042 Universal Peripheral Interface 8-Bit Microcomputer into the system with a minimum of other logic(external logic is needed to report the status of Non-Maskable Interrupt).
- Address Decode, "DTACK" generation, Chip Select, and Write Enable provided for interfacing an EEPROM to the system.
- Programmable BUS ERROR time out period(6 micro-seconds to 1026 micro-seconds in 4 micro-second increments). The new design of the BUS ERROR time out counter allows block mode transfers to work properly.
- DIO Bus Buffer Control. "NdmaActive" and "Nima" signals used to perform Buffer Control for the DMA chip and for NIKKI(or any other device sharing the local Data BUS).

HEWLETT PACKARD

1.2 Pin List

Pin	Name	Dir.	BURNNI Pinout DRIVE Capability (0=Drive 0) (1=Drive 1) (Z=Tri-State)	Pull Up	Side	Ploc	Pad
1	lad[0]	I/O	0,1,Z	Y	TOP	420	BUSPAD10
2	lad[1]	I/O	0,1,Z	Y	TOP	410	BUSPAD10
3	lad[2]	I/O	0,1,Z	Y	TOP	400	BUSPAD10
4	lad[3]	I/O	0,1,Z	Y	TOP	390	BUSPAD10
5	lad[4]	I/O	0,1,Z	Y	TOP	380	BUSPAD10
6	lad[5]	I/O	0,1,Z	Y	TOP	370	BUSPAD10
7	lad[6]	I/O	0,1,Z	Y	TOP	360	BUSPAD10
8	lad[7]	I/O	0,1,Z	Y	TOP	350	BUSPAD10
9	GND(dirty)				TOP	340	DGNDPAD
10	lad[8]	I/O	0,1,Z	Y	TOP	330	BUSPAD10
11	lad[9]	I/O	0,1,Z	Y	TOP	320	BUSPAD10
12	lad[10]	I/O	0,1,Z	Y	TOP	310	BUSPAD10
13	lad[11]	I/O	0,1,Z	Y	TOP	300	BUSPAD10
14	lad[12]	I/O	0,1,Z	Y	TOP	290	BUSPAD10
15	lad[13]	I/O	0,1,Z	Y	TOP	280	BUSPAD10
16	lad[14]	I/O	0,1,Z	Y	TOP	270	BUSPAD10
17	lad[15]	I/O	0,1,Z	Y	TOP	260	BUSPAD10
18	VDD(dirty)				TOP	250	DVDDPAD
19	Sad[0]	I/O	0,1,Z	Y	TOP	240	BUSPAD10
20	Sad[1]	I/O	0,1,Z	Y	TOP	230	BUSPAD10
21	Sad[2]	I/O	0,1,Z	Y	TOP	220	BUSPAD10
22	Sad[3]	I/O	0,1,Z	Y	TOP	210	BUSPAD10
23	Sad[4]	I/O	0,1,Z	Y	TOP	200	BUSPAD10
24	Sad[5]	I/O	0,1,Z	Y	TOP	190	BUSPAD10
25	Sad[6]	I/O	0,1,Z	Y	TOP	180	BUSPAD10
26	Sad[7]	I/O	0,1,Z	Y	TOP	170	BUSPAD10
27	GND(dirty)				TOP	160	DGNDPAD
28	Sad[8]	I/O	0,1,Z	Y	TOP	150	BUSPAD10
29	Sad[9]	I/O	0,1,Z	Y	TOP	140	BUSPAD10
30	Sad[10]	I/O	0,1,Z	Y	TOP	130	BUSPAD10
31	Sad[11]	I/O	0,1,Z	Y	TOP	120	BUSPAD10
32	Sad[12]	I/O	0,1,Z	Y	TOP	110	BUSPAD10
33	Sad[13]	I/O	0,1,Z	Y	TOP	100	BUSPAD10
34	Sad[14]	I/O	0,1,Z	Y	TOP	90	BUSPAD10
35	Sad[15]	I/O	0,1,Z	Y	TOP	80	BUSPAD10
36	NdmaActive	I		Y	TOP	50	BUSPAD10
37	Ntest	I		Y	TOP	40	BUSPAD10
38	oe	I		Y	TOP	30	BUSPAD10
39	NsrCsu	O(I)	0,1,Z	N	TOP	20	BUSPAD10
40	NsrCsl	O	0,1,Z	N	TOP	10	BUSPAD10

HEWLETT PACKARD

BURNNI Pinout(Cont.)

Pin	Name	Dir.	DRIVE Capability (0=Drive 0) (1=Drive 1) (Z=Tri-State)	Pull Up	Side	Ploc	Pad
41	Nsrwr	O	0,1,Z	N	LEFT	440	BUSPAD10
42	NsrOe	O	0,1,Z	N	LEFT	430	BUSPAD10
43	srAle	O	0,1,Z	N	LEFT	420	BUSPAD10
44	Nkbcs	O(I)	0,1,Z	Y	LEFT	410	BUSPAD10
45	kbRs0	O	0,1,Z	N	LEFT	400	BUSPAD10
46	NkbReset	O	0,1,Z	N	LEFT	390	BUSPAD10
47	VDD(dirty)				LEFT	380	DVDDPAD
48	GND(dirty)				LEFT	370	DGNDPAD
49	NkbIor	O	0,1,Z	N	LEFT	360	BUSPAD10
50	NkbIow	O	0,1,Z	N	LEFT	350	BUSPAD10
51	Nromcs	O(I)	0,1,Z	Y	LEFT	340	BUSPAD10
52	Nledcs	O(I)	0,1,Z	Y	LEFT	330	BUSPAD10
53	Neeoe	O(I)	0,1,Z	Y	LEFT	320	BUSPAD10
54	Neewe	O	0,1,Z	N	LEFT	310	BUSPAD10
55	GND(dirty)				LEFT	280	DGNDPAD
56	Nclk5M	O(I)	0,1,Z	N	LEFT	270	BUSPAD7
57	clk5M	O(I)	0,1,Z	N	LEFT	260	BUSPAD7
58	clk10M	I(O)	0,1,Z	N	LEFT	250	BUSPAD10
59	VDD(dirty)				LEFT	240	DVDDPAD
60	clk40M	I		N	LEFT	210	BUSPAD10
61	GND(core)				LEFT	200	GND
62	ba[1]	I		Y	LEFT	190	BUSPAD10
63	ba[2]	I		Y	LEFT	180	BUSPAD10
64	ba[3]	I		Y	LEFT	170	BUSPAD10
65	ba[4]	I		Y	LEFT	160	BUSPAD10
66	ba[5]	I		Y	LEFT	150	BUSPAD10
67	ba[6]	I		Y	LEFT	140	BUSPAD10
68	ba[7]	I		Y	LEFT	130	BUSPAD10
69	ba[8]	I		Y	LEFT	120	BUSPAD10
70	ba[9]	I		Y	LEFT	110	BUSPAD10
71	ba[10]	I		Y	LEFT	100	BUSPAD10
72	ba[11]	I		Y	LEFT	90	BUSPAD10
73	ba[12]	I		Y	LEFT	80	BUSPAD10
74	ba[13]	I		Y	LEFT	70	BUSPAD10
75	ba[14]	I		Y	LEFT	60	BUSPAD10
76	ba[15]	I		Y	LEFT	50	BUSPAD10
77	ba[16]	I		Y	LEFT	40	BUSPAD10
78	ba[17]	I		Y	LEFT	30	BUSPAD10
79	ba[18]	I		Y	LEFT	20	BUSPAD10
80	ba[19]	I		Y	LEFT	10	BUSPAD10

HEWLETT PACKARD

BURNNI Pinout(Cont.)

Pin	Name	Dir.	DRIVE Capability (0=Drive 0) (1=Drive 1) (Z=Tri-State)	Pull Up	Side	Ploc	Pad
81	ba[20]	I		Y	BOTTOM	10	BUSPAD10
82	ba[21]	I		Y	BOTTOM	20	BUSPAD10
83	ba[22]	I		Y	BOTTOM	30	BUSPAD10
84	ba[23]	I		Y	BOTTOM	40	BUSPAD10
85	VDD(clean)				BOTTOM	50	CVDDPAD
86	bd[0]	I/O	0,1,Z	N	BOTTOM	60	BUSPAD10
87	bd[1]	I/O	0,1,Z	N	BOTTOM	70	BUSPAD10
88	bd[2]	I/O	0,1,Z	N	BOTTOM	80	BUSPAD10
89	bd[3]	I/O	0,1,Z	N	BOTTOM	90	BUSPAD10
90	bd[4]	I/O	0,1,Z	N	BOTTOM	100	BUSPAD10
91	bd[5]	I/O	0,1,Z	N	BOTTOM	110	BUSPAD10
92	bd[6]	I/O	0,1,Z	N	BOTTOM	120	BUSPAD10
93	bd[7]	I/O	0,1,Z	N	BOTTOM	130	BUSPAD10
94	GND(dirty)				BOTTOM	140	DGNDPAD
95	bd[8]	I/O	0,1,Z	N	BOTTOM	150	BUSPAD10
96	bd[9]	I/O	0,1,Z	N	BOTTOM	160	BUSPAD10
97	bd[10]	I/O	0,1,Z	N	BOTTOM	170	BUSPAD10
98	bd[11]	I/O	0,1,Z	N	BOTTOM	180	BUSPAD10
99	bd[12]	I/O	0,1,Z	N	BOTTOM	190	BUSPAD10
100	bd[13]	I/O	0,1,Z	N	BOTTOM	200	BUSPAD10
101	bd[14]	I/O	0,1,Z	N	BOTTOM	210	BUSPAD10
102	bd[15]	I/O	0,1,Z	N	BOTTOM	220	BUSPAD10
103	VDD(dirty)				BOTTOM	230	DVDDPAD
104	xbd[0]	I/O	0,1,Z	N	BOTTOM	240	BUSPAD10
105	xbd[1]	I/O	0,1,Z	N	BOTTOM	250	BUSPAD10
106	xbd[2]	I/O	0,1,Z	N	BOTTOM	260	BUSPAD10
107	xbd[3]	I/O	0,1,Z	N	BOTTOM	270	BUSPAD10
108	xbd[4]	I/O	0,1,Z	N	BOTTOM	280	BUSPAD10
109	xbd[5]	I/O	0,1,Z	N	BOTTOM	290	BUSPAD10
110	xbd[6]	I/O	0,1,Z	N	BOTTOM	300	BUSPAD10
111	xbd[7]	I/O	0,1,Z	N	BOTTOM	310	BUSPAD10
112	GND(dirty)				BOTTOM	320	DGNDPAD
113	xbd[8]	I/O	0,1,Z	N	BOTTOM	330	BUSPAD10
114	xbd[9]	I/O	0,1,Z	N	BOTTOM	340	BUSPAD10
115	xbd[10]	I/O	0,1,Z	N	BOTTOM	350	BUSPAD10
116	GND(clean)				BOTTOM	360	CGNDPAD
117	xbd[11]	I/O	0,1,Z	N	BOTTOM	370	BUSPAD10
118	xbd[12]	I/O	0,1,Z	N	BOTTOM	380	BUSPAD10
119	xbd[13]	I/O	0,1,Z	N	BOTTOM	390	BUSPAD10
120	xbd[14]	I/O	0,1,Z	N	BOTTOM	400	BUSPAD10

HEWLETT PACKARD

BURNNI Pinout(Cont.)

Pin	Name	Dir.	DRIVE Capability (0=Drive 0) (1=Drive 1) (Z=Tri-State)	Pull Up	Side	Ploc	Pad
121	xbd[15]	I/O	0,1,Z	N	RIGHT	10	BUSPAD10
122	Niack	I		N	RIGHT	40	BUSPAD10
123	Nbdrv	I		N	RIGHT	50	BUSPAD10
124	Nlwd	I		N	RIGHT	60	BUSPAD10
125	Nlds	I		N	RIGHT	70	BUSPAD10
126	Nuds	I		N	RIGHT	80	BUSPAD10
127	Nbas32	I		N	RIGHT	90	BUSPAD10
128	Nbas24	I		N	RIGHT	100	BUSPAD10
129	Nwrite	I		N	RIGHT	110	BUSPAD10
130	Ndtkall	I		N	RIGHT	120	BUSPAD10
131	Nreset	I		N	RIGHT	130	BUSPAD10
132	NPuReset	I		N	RIGHT	140	BUSPAD10
133	Nima	I/O	0,1,Z	Y	RIGHT	150	BUSPAD10
134	Ndtack16	O	0,Z	Y	RIGHT	160	BUSPAD7
135	Ndtack32	O	0,Z	Y	RIGHT	170	BUSPAD7
136	Nberr	O(I)	0,Z	Y	RIGHT	180	BUSPAD7
137	GND(dirty)				RIGHT	190	DGNDPAD
138	Nir[3]	O	0,Z	Y	RIGHT	200	BUSPAD10
139	Nir[4]	O	0,Z	Y	RIGHT	210	BUSPAD10
140	Nir[5]	O	0,Z	Y	RIGHT	220	BUSPAD10
141	VDD(core)				RIGHT	230	VDD
142	Nir[6]	O	0,Z	Y	RIGHT	240	BUSPAD10
143	bfDir	O	0,1,Z	N	RIGHT	250	BUSPAD10
144	GND(dirty)				RIGHT	260	DGNDPAD
145	VDD(dirty)				RIGHT	270	DVDDPAD
146	bfHdEn	O	0,1,Z	N	RIGHT	280	BUSPAD10
147	NbfLdEn	O	0,1,Z	N	RIGHT	290	BUSPAD10
148	NbfXdEn	O	0,1,Z	N	RIGHT	300	BUSPAD10
149	NlnBusEn	I/O	0,1,Z	Y	RIGHT	310	BUSPAD10
150	lnread	I/O	0,1,Z	Y	RIGHT	320	BUSPAD10
151	GND(dirty)				RIGHT	330	DGNDPAD
152	Nlance	I/O	0,1,Z	Y	RIGHT	340	BUSPAD10
153	Nlnrdy	I/O	0,1,Z	Y	RIGHT	350	BUSPAD10
154	NlnHlda	O	0,1,Z	N	RIGHT	360	BUSPAD10
155	Nlnreset	O	0,1,Z	Y	RIGHT	370	BUSPAD10
156	NlnBm0	I		Y	RIGHT	380	BUSPAD10
157	NlnBm1	I		Y	RIGHT	390	BUSPAD10
158	lnAle	I		Y	RIGHT	400	BUSPAD10
159	NlnHold	I		Y	RIGHT	410	BUSPAD10
160	NlnIntr	I		Y	RIGHT	420	BUSPAD10

1.3 Pin Description

1.3.1 BURNNI DIO/DIO-II signal pins

Pin Name	Type	Description
ba[23:1]	I	DIO address bus lines BA23 through BA1.
bd[15:0]	I/O	DIO data bus lines BD15 through BD0. These lines are active during byte and word transfers.
xbd[15:0]	I/O	DIO-II data bus lines XD15 through XD0. These lines are used only during long word transfers.
Nbas24	I	DIO $\overline{\text{BAS24}}$ (Low True Buffered Address Strobe) signal. This signal asserts when there is a valid 24 bit address on the address bus.
Nbas32	I	DIO-II $\overline{\text{BAS32}}$ (Low True Buffered Address Strobe for 32 bit address) signal. This signal asserts when there is a valid 32 bit address on the address bus. If the upper 8 address lines(BA24 through BA31) are all low, then both Nbas32 and Nbas24 will be asserted.
Nlds	I	DIO $\overline{\text{BLDS}}$ (Low True Buffered Lower Data Strobe) signal. This signal asserts when there is valid data on the lower data bus(BD7 through BD0) for writes and asserts when it is acceptable to drive the lower data bus for reads.
Nuds	I	DIO $\overline{\text{BUDS}}$ (Low True Buffered Upper Data Strobe) signal. This signal asserts when there is valid data on the upper data bus(BD15 through BD8) for writes and when it is acceptable to drive the upper data bus for reads.
Nwrite	I	DIO BR/$\overline{\text{W}}$ (High True Read/Low True Write) signal.
Ndtkall	I	Low True $\overline{\text{DTACK}}$ all signal. This pin should be asserted when any of the following DIO/DIO-II signals assert: $\overline{\text{DTACK16}}$, $\overline{\text{DSACK32}}$, or $\overline{\text{DSACK16}}$. This pin can also be asserted when it is disarable to disable the Bus ERROR timeout function. In Test Mode(Ntest low), Ndtkall becomes the "scanen" signal. Scanen(Ndtkall) high puts the part in serial mode, scanen low puts the part in parallel mode. These modes are used when running ATG test vectors.
Niack	I	DIO-II $\overline{\text{IACK32}}$ (Low True) signal. The Interrupt Acknowledge signal only goes to the BUS Error Timer circuit and is treated like Nbas32 and Nbas24 in starting and stopping the DIO Bus Error timer. In fact, Niack is logically OR'ed with Nbas32 and Nbas24 within the BUS ERROR Timer circuit. The DIO BUS Specification does not allow Nbas32 or Nbas24 to assert when Niack asserts.

Nbdrv	I	DIO $\overline{\text{BDRV}}$ (Bus Drive Disable) signal. This DIO diagnostic/development signal, when asserted, will only disable BOOT ROM(Both normal and extended) accesses and Test LED accesses. No other BURNNI supported I/O will be affected. However, during the serial phase of test mode(Ntest low and Ndtkall high), this pin becomes the "Scan Path Input" pin.
Nlwd	I	DIO-II $\overline{\text{LWORD}}$ (Low True Long WORD) signal. Long WORD indicates that a long word transfer is being initiated on the bus.
Nreset	I	DIO $\overline{\text{RESET}}$ signal. This signal resets the Programmable Timer Module(PTM) within BURNNI, will cause the NkbReset to assert for 26 uS, clears the EEPROM Write Enable bit, and resets the LAN logic.
NPuReset	I	Power Up Reset Signal which initializes all but the clock generation logic within BURNNI. This signal should be made true(low) when both $\overline{\text{HALT}}$ and $\overline{\text{RESET}}$ are asserted.
Nima	I/O	DIO $\overline{\text{IMA}}$ (Low True I'M ADDRESSED) Signal. This pin is driven low when BURNNI sees a valid address for an I/O device it supports and has enabled. This pin is also driven high for 50 nS(50 to 75 nS after Nbas24 or Nbas32 goes high). At all other times, Nima is an input pin which contains information to control the external Data Bus buffers for NIKKI(or any other external I/O device sharing the local data bus).
Ndtack16	O(OC)	DIO $\overline{\text{DTACK16}}$ (Low True Data Transfer ACKnowledge 16) signal. This signal is used to handshake byte and word length data transfers. This pin only drives low, the bus master is expected to pre-charge this line high at the end of the data transfer cycle.
Ndtack32	O(OC)	DIO-II $\overline{\text{DSACK32}}$ (Low True Datatransfer and Size ACKnowledge 32) signal. This signal is used to handshake longword data transfers. This pin only drives low, the bus master is expected to pre-charge this line high at the end of the data transfer cycle.
Nberr	O(OC)	DIO $\overline{\text{BERR}}$ (Low True Bus ERRor) signal. A bus error is generated if the current bus cycle(Nbas24, Nbas32, or Niack low and Ndtkall not low) has not completed within the time allocated(See Programmable Bus Error Section). This pin only drives low, the bus master is expected to pre-charge this line high at the end of the data transfer cycle. If Nberr is held low through a power-up reset(NPuReset) cycle(See Hold time requirements in timing diagrams), Nberr will remain in tri-state until the next power-up reset in which Nberr is pulled high. If Nberr was held low through a power-up reset, the Bus Error timeout support circuits within BURNNI will be disabled and bit #2 in the Status Register at address $400003 will be cleared.

Nir[6:3]	O(OC)	DIO $\overline{IR6}$ through $\overline{IR3}$ (Low True Interrupt Request) signals. This pin only drives low, a pullup resistor on either the backplane or on the CPU board is expected to return this pin high after the interrupt is serviced. BURNNI's Programmable Timer Module(PTM) can generate an $\overline{IR6}$ signal. The LAN circuits can generate an interrupt on $\overline{IR6}$ through $\overline{IR3}$, depending on which interrupt level is enabled in the LAN's Control Register.

1.3.2 BURNNI Control/Support signals

Pin Name	Type	Description
clk40M	I	40 MHz clock for all logic with the exception of LAN. During test mode(Ntest low), this is the clock source for all flip flops in the scan path.
clk10M	I(O)	10 MHz clock for LAN logic. During test mode(Ntest low), this pin is an output driving the clk40M signal out and then back into BURNNI.
clk5M	O(I)	5 MHz clock out used to support the keyboard and low speed HP-IB circuits. During test mode(Ntest low), this pin becomes the "masken" input. When "masken" is high, a one is forced out on "NkbIow". A low on "masken" has no effect on the logic value on "NkbIow".
Nclk5M	O	5 MHz clock(inverted clk5M) clock out used to support the keyboard circuits. During test mode(Ntest low), this pin is tri-stated.
NbfLdEn	O	Low True Lower Data Bus Buffer Enable.
bfHdEn	O	High True Upper Data Bus Buffer Enable. This signal is high true so that it can be easily gated with " $\overline{FOLD}$ " in order to complete the logic necessary to control the Upper Data Bus Buffer.
NbfXdEn	O	Low True Extended Data Bus Buffer Enable. This signal enables the xbd[15:0] data bus buffers.
bfDir	O	Buffer Direction Control. A high is meant to enable data from the local bus onto the DIO bus.
Ntest	I	Low True test pin used to prepair the IC for ATG vectors. Asserting this pin also disables all internal pull-ups with the exception of the internal pull up resistor on the Ntest pin itself. The internal pull-up resistor on the Ntest pin is controlled by the first flip-flop in the scan chain.
oe	I	High True output enable pin. When this pin is driven low, all outputs on BURNNI are tri-stated.
NdmaActive	I	Low True Dma Active signal. This signal is used in the external buffer control equations for active DMA.

1.3.3 Keyboard Support(8042 Universal Peripheral Interface)

Pin Name	Type	Description
kbRs0	O	Latched ba[1] signal. This is necessary to meet hold time on address for the 8042's "A0" pin.
NkbIor	O	Low True Keyboard Read Strobe.
NkbIow	O	Low True Keyboard Write Strobe.
NkbReset	O	This signal goes low for a minimum of 25 micro-seconds as a result of NPuReset or Nreset asserting. This signal is used to reset the 8042(Keyboard) and reset the 9914(Low Speed Internal HP-IB).
Nkbcs	O(I)	Low True Keyboard(8042) Chip Select. This signal is asserted for Read or Write cycles to DIO Address $42XXXX. If Nkbcs is held low through a power-up reset(NPuReset) cycle(See Hold time requirements in timing diagrams), Nkbcs will remain in tri-state until the next power-up reset in which Nkbcs is pulled high. If Nkbcs was held low through a power-up reset, the keyboard support circuits within BURNNI will be disabled.

1.3.4 BOOT ROM/Test Status LEDs

Pin Name	Type	Description
Nromcs	O(I)	Low True ROM Chip Select. This signal is asserted for Read cycles to DIO Addresses $000000 through $01FFFF. If the "exr" bit is set in the control register at address $40000A, then the "extended ROM decode" feature is activated and Nromcs will assert for Read cycles to DIO Addresses $000000 through $03FFFF. If Nromcs is held low through a power-up reset(NPuReset) cycle(See Hold time requirements in timing diagrams), Nromcs will remain in tri-state until the next power-up reset in which Nromcs is pulled high. If Nromcs was held low through a power-up reset, the BOOT ROM support circuits within BURNNI will be disabled.

Nledcs	O(I)	Low True test LED Chip Select. This signal is asserted for Write cycles to the DIO Addresses $004000 through $01FFFF in which ba[14] is high. Nledcs is used to write BOOT ROM Test status information to an eight LEDs. Due to past hardware designs, the address $01FFFF should be used to write information to the test LEDs. Also, writing a '0' to $01FFFF will turn on all the test LEDs, while writing a '$FF' will turn off all of them. If Nledcs is held low through a power-up reset(NPuReset) cycle(See Hold time requirements in timing diagrams), Nledcs will remain in tri-state until the next power-up reset in which Nledcs is pulled high. If Nledcs was held low through a power-up reset, the test LED support circuits within BURNNI will be disabled, however, if the BOOT ROM is still enabled, writes to the BOOT ROM address space will still cause a "Ndtack16" to be generated.

1.3.5 EEPROM Support

Pin Name	Type	Description
Neeoe	O(I)	Low True EEPROM Output Enable. This signal is asserted for Read cycles to the DIO Addresses $408000 through $40FFFF. The EEPROM can be read at all times unless the EEPROM circuitry has been disabled as described below. If Neeoe is held low through a power-up reset(NPuReset) cycle(See Hold time requirements in timing diagrams), Neeoe will remain in tri-state until the next power-up reset in which Neeoe is pulled high. If Neeoe was held low through a power-up reset, the EEPROM support circuits within BURNNI will be disabled.
Neewe	O	Low True EEPROM Write Enable Strobe. This signal is asserted for Write cycles involving ODD bytes in the DIO Addresses $408000 through $40FFFF if EEPROM writes have been enabled. If EEPROM writes have not been enabled, then Neewe will not assert, however, Ndtack16 will still be asserted(Unless EEPROM support has been disabled using the "Neeoe" pin).

1.3.6 LAN Support

Pin Name	Type	Description
lad[15:0]	I/O	This is a multiplexed address and data bus used for communication to/from the AMD 7990 LANCE local area network controller chip. When $\overline{\text{Holda}}$ is asserted low, the LANCE chip is the Master of this bus. At power-up Burnni is the bus master.
Sad[15:0]	I/O	The Static ram multiplexed Address and Data bus. This 16 bit bus is used for accessing 8K words of static RAM data. This is where incoming and outgoing lan packet data is buffered.

NsrCsu, and NsrCsl	O(I)	Low True static ram Chip select upper and lower. These are asynchronous chip enables for the 2 Static Ram chips on the Sad bus. They are the output of a multiplexer and reflect The DIO data strobes for CPU accesses, and the Byte Masks from the Lance chips for 7990 accesses. At power-up, the NsrCsu pin is temporarily an input sensing if it is pulled-up or pulled-down. This state is saved and is reflected in a hidden status register; its intent is to let the system software sense if the lan subsystem has been configured for Thin lan (pulled-up) or Thick lan (pull-down).
Nsrwr	O	Low True static ram write. This signal is asserted low to perform a write operation of SAD data into the static Ram. Data is valid before the positive edge.
NsrOe	O	Low True static ram Output Enable. This signal is asserted during reads of the static ram. Data is latched on the rising edge.
srALE	O	High True staic ram Address Latch Enable. When this signal is high, the external latch on the SAD bus is enabled to sample an Address to the static ram. When it is low, address is latched for the static ram and a read or write cycle can proceed.
Nlnreset	O	Low True reset signal to the Lance chip also reflects an internal Burnni lan reset. This can be generated by: 1) a power-up reset, 2) a DIO backplane reset, or 3) a write to the lan reset register (reg 1).
NlnHold	I	Low True 'Bus Request' signal from the Lance indicating it wants bus mastership of the lad bus.
NlnHlda	O	Low True 'BusGrant' signal to the Lance indicating it is bus master of the lad bus. This signal once asserted will remain asserted until the Lance chip de-asserts NlnHold.
lnAle	I	High True lan Address Latch Enable. When this signal is high a latch sensing the lad bus is enabled. When NlnHlda is asserted low and lnAle consequently is asserted low, the latch is closed with a valid address from the Lance for the Static Ram. This signal is then synchronized, and used in the CPU/ Lance arbitration for the SAD bus. If a CPU cycle is not in progress, then an internal Lance grant is issued, and a Lance cycle to/from static memory is initiated. This signal also has a special meaning at power-up; if it is high, then the lan section is enabled; if it is low, then all lan logic is disabled.
NlnBm1, and NlnBm0	I	Low True Byte Masks from the Lance chip. When a Lance grant has occured, these signals become NsrCsu and NsrCsl respectively.
NlnIntr	I	Low True Lance Interrupt. When the Lance asserts an interrupt, it is reflected in the lan status register. If it is asserted and interrupts are enabled then Burnni will assert one of NIr[6:3] depending on which interrupt level has been programmed.

lnread	I/O	High True Read, low true write. This signal is used for Lance communication via the lad bus. If the Lance is a slave, High means the 7990 will drive data on the lad bus, and if Low it will accept data. If the Lance is the lad bus master, High means the 7990 wants to read data from the static ram, Low means it wants to drive the lad bus and write to the static ram.
Nlance	I/O	Low true $\overline{DAS}$ to the Lance chip. When the Lance is a Bus Slave, Burnni drives this signal to indicate the data portion of a lad bus transaction. The Lance accepts data on the rising edge. When the Lance is the lad bus master, it drives this signal indicating the data portion of its transaction, the lan interface, however, assumes the data portion of the transaction occurs after lnAle is driven low by the Lance.
NlnBusEn	I/O	Low True $\overline{DALI}$ to the Lance chip. When the Lance is a Bus Slave, Burnni drives this signal low to enable data out to the Lance chip on write cycles. When the Lance chip is Bus Master, it drive NlnBusEn low to enable Burnni to drive data on the lad bus on read cycles.
NlnRdy	I/O	Low True Ready. When the Lance chip is a Bus slave, the Lance chip drives NlnRdy. It asserts low to indicate it is ready to accept data on writes, or is providing vaild data on reads. When the Lance chip is a Bus Master, Burnni drives this pin low when it is ready to accept data on Lance writes, or is providing valid data on Lance reads.

1.4 Electrical Specifications

ABSOLUTE MAXIMUM RATINGS			
Parameter	Symbol	Value	Unit
Supply Voltage	V_{dd}	-.5 to +7.0	V
Input Voltage	V_{in}	-.5 to +7.0	V
Input DC Current	I_{in}	± 100	mA
Power Dissipation	P_D	1.0	W
Storage Temperature Range	T_{stg}	-40 to + 125	°C

Stress beyond listed max. ratings may cause permanent damage to the device. Exposure to max. rated conditions for extended periods will adversely affect device reliability.

HEWLETT PACKARD

Recommended Operating Conditions		Limits			
Parameter	Symbol	Min.	Nom.	Max.	Unit
Supply Voltage	V_{dd}	4.75	5.00	5.25	V
Supply Current	I_{dd}		60	100	mA
High-level Input Voltage	V_{IH}	2			V
Low-level Input Voltage	V_{IL}			0.8	V
Operating Free-air Temperature	T_A	0		70	°C

DC Electrical Characteristics			Limits			
Parameter	Pins	Symbol	Min.	Nom.	Max.	Unit
High-level Output Voltage at I_{OH}	All	V_{OH}	2.4			V
Low-level Output Voltage at I_{OL}	All	V_{OL}			0.4	V
High-level Output Current	All except: Nclk5M, clk5M, Nberr, Ndtack16, and Ndtack32	I_{OH}			3.0	mA
Low-level Output Current	All except: Nclk5M, clk5M, Nberr, Ndtack16, and Ndtack32	I_{OL}			6.2	mA
High-level Output Current	Nclk5M, clk5M, Nberr, Ndtack16, and Ndtack32	I_{OH}			4.4	mA
Low-level Output Current	Nclk5M, clk5M, Nberr, Ndtack16, and Ndtack32	I_{OL}			9.9	mA

Input Protection	Limits			
Parameter	Min.	Nom.	Max.	Unit
Electrostatic Discharge(between any 2 pins) without damage ‡	2			KV
Latchup Current(DC)	±100			mA

‡ ± 2 KV through 1500 Ohms in series with 100 pF

Output Load Capacitance(used for signals in timing diagrams)	
Pin	Capacitance(pF)
NsrCsu, NsrCsl, Nsrwr, NsrOe, srAle, Nkbcs, kbRs0, NkbIor, NkbIow, Nromcs, Nledcs, Neeoe, Neewe, bfHdEn, NbfLdEn, NbfXdEn, NlnBusEn, lnread, Nlance, Nlnrdy, NlnHlda, Nlnreset	60
lad[0..15], Sad[0..15], NkbReset, Nclk5M, clk5M, clk10M, Nima, Nir[3..6], bfDir, Ndtack32	90
bd[0..15], xbd[0..15]	150
Nberr, Ndtack16 ‡	500

‡ LAN timing used 90 pF for Ndtack16

2. LAN Support

3. NEW I/O Capabilities

3.1 Overview

BURNNI decodes a previously unused 64K byte internal I/O address space whose starting address is $00400000. This new internal I/O address space is used to add four new features to the Series 300 family:

- A Programmable Bus Error timer.
- Up to 8K-bytes of EEPROM and EEPROM support.
- Extended BOOT ROM address space decoding.
- Programmable Timer Module(PTM) enhancements.

Also within this internal I/O address space are locations for controlling "built-in I/O" such as the LAN support offered by BURNNI and all the I/O supported by another ASIC which is code named NIKKI.

Addresses $00400000 through $0040000B represent control registers for the new I/O capabilities listed above. Addresses $0040000C through $0040000F are reserved for future use and currently "Bus Error" when accessed. Addresses $00400010 through $0040003F are where the special "built-in I/O" control registers are located. If a register is present, it will "DTACK" otherwise the address will "Bus Error". Addresses $00400040 through $00407FFF should not be used. These addresses contain images of address space $00400000 through $0040003F, an artifact of not decoding all address lines. This situation could change in the future.

Addresses $00408000 through $0040FFFF will "DTACK" if the EEPROM is enabled(a logic "1" is present on the "Neeoe" pin after a Power Up Reset). The lower(odd) bytes within this address range is where the EEPROM is located. The upper(even) bytes are currently reserved and if read, the data returned will be implementation dependent(Designers using BURNNI are strongly urged to put pull-up resistors on the data bus in order to avoid software reading a floating data bus should this upper byte not be used).

The EEPROM is intended to be used to store information which would normally be set using switches and store LAN information which was kept in "NOVRAM" in older systems. Currently, a 2K byte EEPROM will be used which has space left over for other information storage. The BOOT ROM will dictate what information will be stored and what form this information will take. See BOOT ROM documentation for more information.

Two words of caution. First, the 2K byte EEPROM is not an exact fit for the address space being decoded, hence, any one location in the 2K byte EEPROM can be accessed with eight different DIO Bus addresses. A 4K byte EEPROM will be mapped to 4 different DIO Bus addresses and a 8K byte EEPROM to 2 different DIO Bus addresses. Only a 16K byte EEPROM will not be multiply mapped. Second, a XICOR X2816 EEPROM is currently being designed into the first system using BURNNI. This part takes up to 10 milli-seconds to write one byte. Software must wait until the EEPROM has had time to complete the current byte write before writing the next byte! Data polling can be used to tell when the write cycle has completed. See the EEPROM data sheet for more information.

"Registers" below refer to bytes of information with the Register number being the byte offset from address $00400000. In the rest of this section, Only a 24 bit address will be shown. The reader should assume that if the upper 8 address bits are not shown, they are implied to be "0".

3.2 Primary ID/Reset Register

Register 1 should be accessed using byte operators operating on address $400001. Writing any information to this register will generate a signal which will clear the "wen" bit in Register 3. The significance of this is pointed out in the next section dealing with Register 3. The data which was written is ignored and hence, lost. Reading this register will return a "0", which indicates the "true" ID is in Register 5.

Register 0 is not really present, however, due to the way decoding is performed, operations to this register will generate a "DTACK" and writes to this register will have an effect on other registers. Writing Register 0 will have the same effect as writing Register 1, however, reading Register 0 will return whatever value the local Upper Data Bus has floated to.

Registers 0 and 1 can be accessed as a word, however, only the lower byte returned from a read operation will contain valid data. Writing both registers using word operators will have the same effect as writing Register 1 or Register 0 using byte operators.

PRIMARY ID/RESET REGISTER - $400000

	REGISTER 0								REGISTER 1							
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ	X	X	X	X	X	X	X	X	0	0	0	0	0	0	0	0
WRITE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

3.3 Status/First ARM Control Register

Register 3 should be accessed using byte operators operating on address $400003. Writing $53 to this register is the first step in enabling writes to the EEPROM. Writing any data will clear the EEPROM write enable(the same as writing to Register 1), however, writing $53 will begin the arming process(with only a write of $CC to Register 5 needed in order to re-arm writes to the EEPROM). Writing Register 2 will also clear the EEPROM write enable. Should Register 2 and Register 3 be written at the same time using word operators, writing $XX53("X" representing any 4-bit pattern) will have the same effect as writing $53 to Register 3 using byte operators. Data written to Register 2 is always lost.

STATUS/FIRST ARM CONTROL REGISTER - $400002

	REGISTER 2								REGISTER 3							
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ	X	X	X	X	X	X	X	X	0	0	0	wen	een	ben	0	0
WRITE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

There are three bits within Register 3 which have meaning, the "wen" or "write enable" bit, the "een" or "eeprom enable" bit, and the "ben" or "bus error enable" bit. All three bits are READ only. Data written to these bits is lost.

When "wen" is set, writes to the EEPROM, Registers 10 or Register 11 are allowed to happen. If "wen" is

cleared, writes to the EEPROM complete as usual with "DTACK" asserting, however, the data being written is not transferred and is hence lost. When "wen" is set, data will be transferred when either the EEPROM, Register 10 or Register 11 is written. "wen" is cleared by any of the following operations:

- Register 0 or Register 1 is written(data written is ignored).
- Register 2 or Register 3 is written(Do not write $53 to Register 3 as this starts the arming process).
- Register 4 is written or Register 5 is written with data other than $CC.
- A DIO backplane Reset is generated(*NRESET).

After modifying the EEPROM, Register 10 or Register 11, software should clear the "wen" bit in order to prevent accidental changes from being made. Notice that BURNNI wakes up(when NPuReset goes false) with this bit cleared.

Should a future product decide to not use BURNNI's EEPROM support, the EEPROM circuits can be disabled by tying the "Neeoe" pin low. By doing this, BURNNI will not "DTACK" reads or writes to addresses $408000 through $40FFFF. Another effect of tying the "Neeoe" pin low is that the "een" bit of Register 3 will be cleared. When the "Neeoe" pin is not tied low(allowed to float high), the "een" bit of Register 3 will be set and BURNNI will "DTACK" all reads or writes to addresses $408000 through $40FFFF. The status of the "Neeoe" pin is sampled shortly after "NPuReset" goes false(HIGH).

The third bit reflects another configuration option of BURNNI. When "ben" is set, BURNNI thinks it is performing the "BUS ERROR timeout" function for the system(or local DIO bus). When "ben" is cleared, then BURNNI is not performing the "BUS ERROR timeout" function. "ben" is cleared by tying "Nberr" low. When "Nberr" is not tied low(allowed to float high), "ben" will be set. The status of the "Nberr" pin is sampled shortly after "NPuReset" goes false(HIGH).

3.4 Extended ID/Second Arm Control Register

BURNNI had Secondary ID $08 assigned to it. In the past internal I/O devices have not had IDs assigned to them. This has led to internal select codes not being able to be "recycled" and software having to special case all internal I/O devices. It was decided to follow external I/O practices when adding this internal I/O device to BURNNI. Register 5 thus returns the Secondary ID of $08 when read. Software should check both the Primary ID and Secondary ID before trying to set up any of the Registers being described in this section.

Register 5 should be accessed using byte operators at address $400005. If Register 4 and Register 5 are read together as a word, only the lower byte($08) will be valid. The upper byte will reflect the status of the upper data bus, which may be floating.

EXTENDED ID/SECOND ARM CONTROL Register - $400004

	REGISTER 4								REGISTER 5							
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ	X	X	X	X	X	X	X	X	0	0	0	0	1	0	0	0
WRITE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Writing $CC to Register 5 or writing $XXCC("X" represents any arbitrary 4 bit pattern) to Registers 4 and 5 as a word operation after having first written $53 to Register 3 will set the "wen" bin in Register 3 assuming

none of the following occur between the write to Register 3 and the write to Register 5:

- Register 0 or Register 1 is written(data written is ignored).
- Register 2 or Register 3 is written(data other than $53 written to Register 3).
- Register 4 is written or Register 5 is written with data other than $CC.
- A DIO backplane Reset is generated(*NRESET).

Any of the above will also clear the "wen" bit once it is set. Writing a $53 to Register 3 will also clear "wen", however, this will leave the write enable process half-way complete.

Writing $CC to Register 5 will have no effect if the first part of the "wen" arming process has not been completed first. Register 5 will, however, "DTACK" this or any other write.

3.5 Bus Error Timeout Control Register

Reads and writes to Register 6 terminate normally with a "DTACK", however, no other operation is performed. Data is lost for the writes while reads return whatever state the Upper Data Bus has floated to.

Register 7 sets the Bus Error Timeout period. It is recommended that this register be accessed using byte operators to address $400007, however, word accessed to address $400006 will also work assuming only the lower byte of the word is treated as useful data for reads and the lower byte of the word contains the "Bus Error Timeout period" for writes.

BUS ERROR TIMEOUT CONTROL REGISTER - $400006

	REGISTER 6								REGISTER 7							
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ	X	X	X	X	X	X	X	X	MSB	-	-	-	-	-	-	LSB
WRITE	X	X	X	X	X	X	X	X	MSB	-	-	-	-	-	-	LSB

Register 7 is cleared by the Power Up Reset signal. The value of this register is used to determine the length of time it takes for the BUS ERROR timer to time out. $00 represents a 6 uS timeout period. $01 represents a 10 uS timeout period and $02 represents a 14 uS timeout period. The following equation can be used to determine the timeout period:

$$P = 6 + (R * 4)$$

where P is the timeout period in micro seconds and R is the decimal value of the Register 7.

The BUS ERROR timer has some additional improvements over earlier designs. Block Mode transfers will now work properly. In the past, the BUS ERROR timer continued to count for as long as Address Strobe($\overline{BAS24}$, and/or $\overline{BAS32}$) was held low. BURNNI's implementation will not only reset the timer when "Nbas24", "Nbas32", and "Niack" are all false(HIGH), but it will also reset the timer whenever the "Ndtkall" signal is asserted. "Ndtkall" is the logical "OR" of all system "DTACKs" and is LOW true. "Ndtkall" can also have a "BUS ERROR TIMING DISABLE" signal externally logically "ORed" into its equation for systems which need to temporarily disable the BUS Error timing function. The BUS ERROR

timer will be started whenever "Nbas24", "Nbas32", or "Niack" assert and "Ndtkall" is not asserted. Whenever a "TIMEOUT" occurs, the "Nberr" pin(normally in tri-state) asserts thus generating a "BUS ERROR".

3.6 Scratch Register

The two Scratch registers(Register 8 and Register 9) can be accessed using byte operators or they both can be accessed together using word operators. These two registers have no control function and can be used by software for any purpose. Both of these registers are cleared by the Power Up Reset signal(NPuReset). Nreset has no effect on the state of these registers. The original purpose of these registers was to assist BOOT ROM code in bring up a multiple processor system.

SCRATCH REGISTER - $400008

	REGISTER 8								REGISTER 9							
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
WRITE	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

3.7 PTM/Expanded BOOT ROM Control Register

Register 10 and Register 11 are used for Programmable Timer Module enhancements and for extended BOOT ROM address space decoding. These two registers can be accessed either as two bytes or as one word. They both can be read at any time, however, writes are not allowed("DTACK" occurs, however data is lost) unless the "wen" bit of Register 3(see section above on Register 3) is set. Both registers are cleared by the Power Up Reset(NPuReset) signal.

PTM/EXPANDED BOOT ROM CONTROL REGISTER - $40000A

	REGISTER 10								REGISTER 11							
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
READ	es7	es6	es5	es4	es3	es2	es1	es0	-	-	-	-	-	t2c	Net	exr
WRITE	es7	es6	es5	es4	es3	es2	es1	es0	-	-	-	-	-	t2c	Net	exr

Register 11's least three significant bits control either PTM enhancements or BOOT ROM decoding. The 5 most significant bits have no control functions and the last data written to these bits(assuming "wen" set in Register 3) can be read back later(the state of "wen" will have no effect on reads). The LSB("exr" or "extended rom") controls the address space occupied by the BOOT ROM(Address Space decoded for the "Nromcs" pin). With "exr" cleared, BOOT ROM decoding is from $00000000 through $0001FFFF. With "exr" set, BOOT ROM decoding is extended and now covers the range of addresses starting at $00000000 and going through $0003FFFF.

Bit 1, or the "Net"(Not enable timer) bit, enables address decoding for the PTM when it is cleared(the power up default condition). This bit was originally put into BURNNI to aid in system turn-on should there have been a fatal error in BURNNI's PTM design. Should a need be found for it, the "Net" bit can be used

to disable address decoding(and "DTACK" generation) for BURNNI's PTM. Caution should be exercised if the "Net" bit is ever set. The "Net" bit just disables decoding(and "DTACK" generation) for the PTM. The PTM itself is still operational. The consequence of this is that should BURNNI's PTM be set up to interrupt and counting is enabled before the "Net" bit is set, the PTM will eventually generate an interrupt. This interrupt can not be cleared until the "Net" bit is cleared. Once the "Net" bit is cleared, the interrupt can then be cleared in the usual way(i.e. through PTM register accesses).

Bit 2, or the "t2c"(timer 2 clock)bit, connects Timer #2's clock input($\overline{C2}$ pin on a Motorola MC6840) to timer 3's output(O3 pin on a Motorola MC6840) when it is cleared(power up default). When "t2c" is set, Timer #2's input is controlled by the "Timer #3 Counter Output Enable" bit in "Control Register #3"(see the "Programmable Timer" Chapter for register location and bit location within the register). With the "Timer #3 Counter Output Enable" bit cleared, the 250 KHz clock will be selected as the clock source. With the "Timer #3 Counter Output Enable" bit set, timer 3's output(O3 pin on a Motorola MC6840) will be selected as the clock source. If "t2c" is cleared and the "Timer #3 Counter Output Enable" bit is also cleared, then there is no signal present to clock Timer #2. For predictable operation of Timer #2, it is recommended that the "t2c" bit not be changed while the PTM is enabled to count. Also, if the "t2c" bit is set, it is recommended that the "Timer #3 Counter Output Enable" bit not be changed while the PTM is enabled to count.

Register 10 is used to "program" the "E" clock rate. All bits in this register(es0 through es7) are used to determine the "E" clock rate. A $00(power up default) in this register will produce an "E" clock period of 100 nS(10 MHz). The "E" clock period is determined by the following expression:

Eperiod = (N * .1) + .1

Eperiod is in micro-seconds and N is the decimal value stored in Register 10. For predictable operation, it is recommended that Register 10 not be changed while the PTM is enabled to count and there are counters using "E" clock as their clock source.

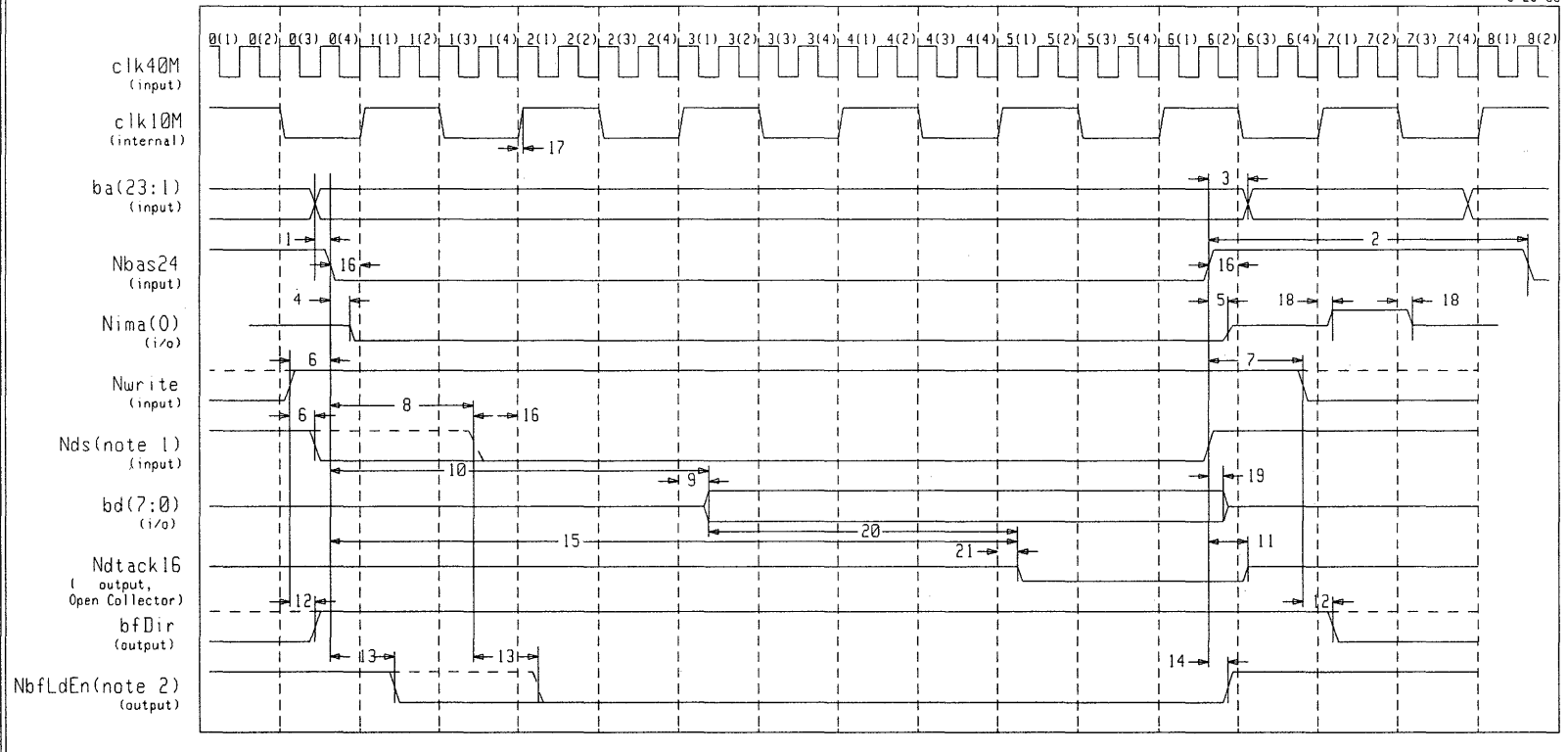
3.8 Timing

The following two figures specify READ and WRITE cycle timing for the BURNNI registers just discussed. Synchronous timing shown is intended for IC test purposes only. "INTERNAL" signals are also shown to assist in IC test vector generation generation. PC board designers should use the asynchronous timing. The same is true for the next two figures which show timing for READ and WRITE cycles to the EEPROM.

HEWLETT PACKARD

DIO read of BURNNI EEPROM/Bus Error Control Registers

6-28-89

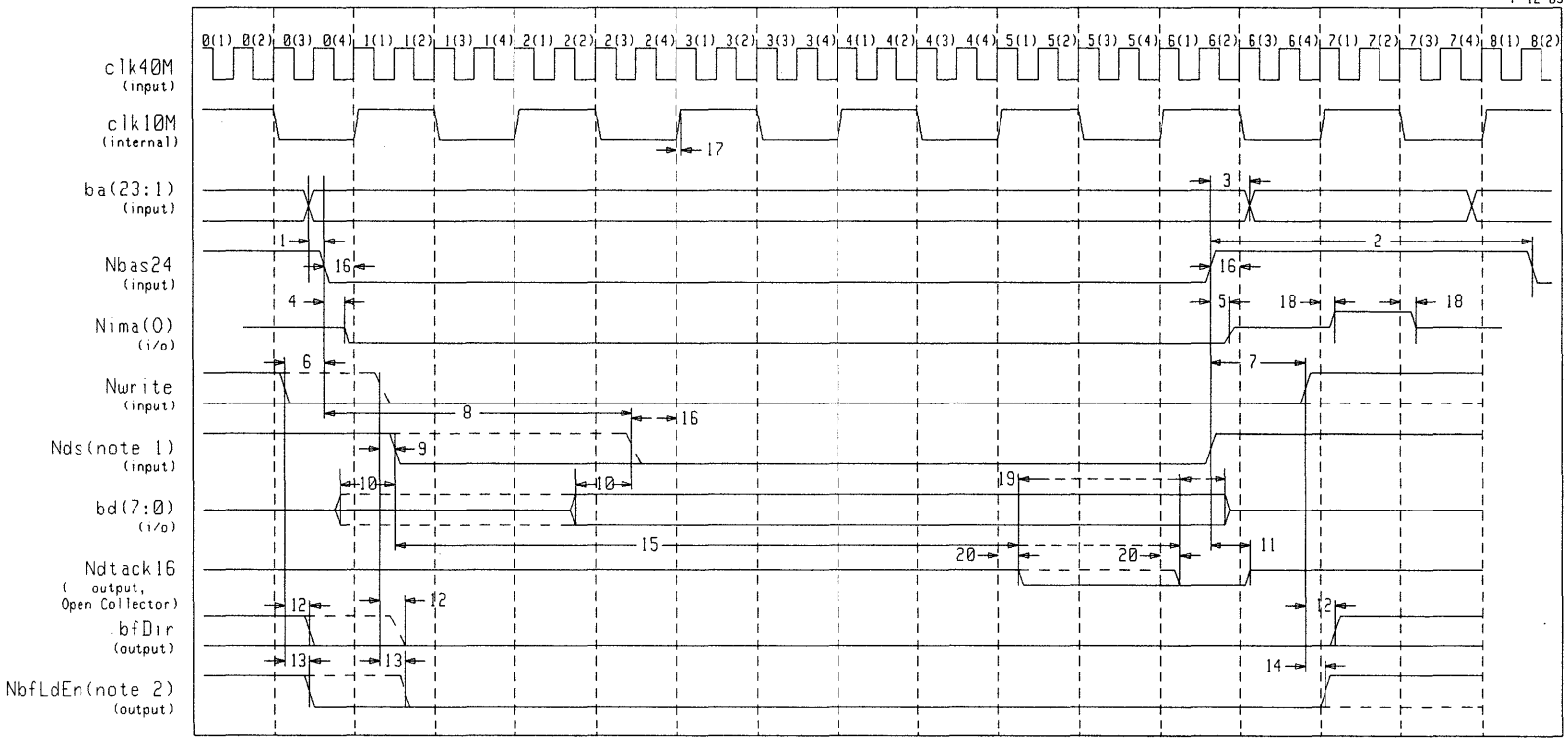


	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite high to Nbas24 or Nds low		7	
7	Nwrite hold after Nbas24 or Nds high		43	8
8	Nbas24 low to Nds low	90	-75	9
9	clk40M high to data bus driven	60		
10	Nbas24 low to valid data	375		7
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nds/Nbas24 low to buffer enable true	40		4
14	Nds/Nbas24 high to buffer enable false	20		5
15	Nbas24 low to Ndtack16 low	575	415	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Nbas24 high to data bus tri-state	25		
20	Data set up before Ndtack low		140	
21	clk40M to Ndtack16 low	60		

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) The DIO Bus Specification requires a 65 nS hold time.
9) This is a DIO bus specification whose limits are not directly tested.

HEWLETT PACKARD

DIO write of BURNNI EEPROM/Bus Error Control Registers



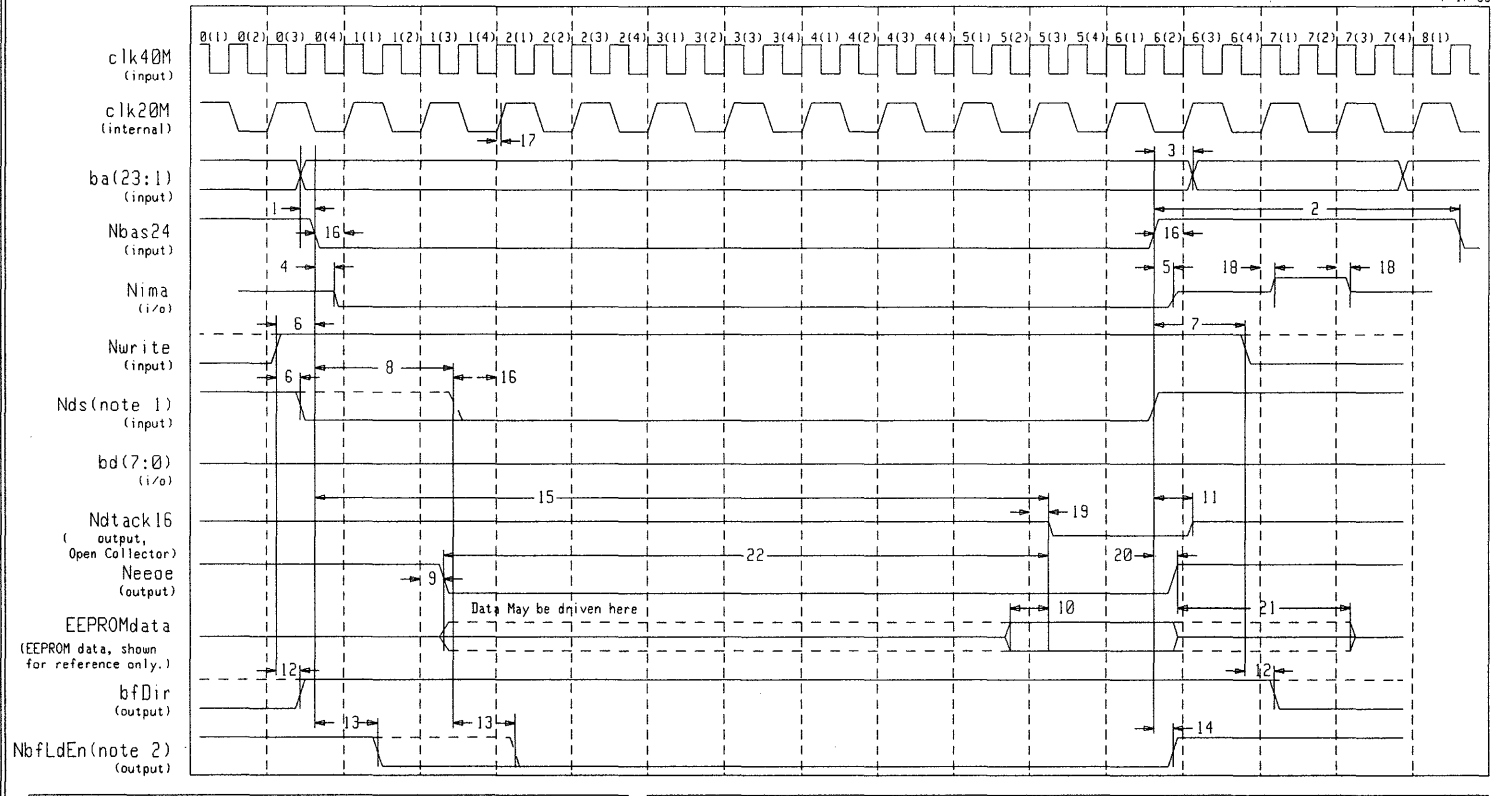
	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite low to Nbas24 low	53	-75	10
7	Nwrite hold after Nbas24 or Nds high		7	8
8	Nbas24 low to Nds low		50	9
9	Nwrite low before Nds low		65	
10	Data setup before Nds low		7	10
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nwrite low to buffer enable true	20		4
14	Nwrite high to buffer enable false	20		5
15	Nds low to Ndtack16 low	460	300	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Data release after Ndtack16 asserted		85	10
20	clk40M to Ndtack16 low	60	0	

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) DIO Bus Specification. For test purposes, Nwrite can go high 7 nS before Nbas24 goes high.
9) DIO Bus specifies 2500 ns Max, however with a programmable Bus ERROR timer, this maximum can increase.
10) DIO Bus Specification.

HEWLETT PACKARD

DIO read of EEPROM

7-17-89



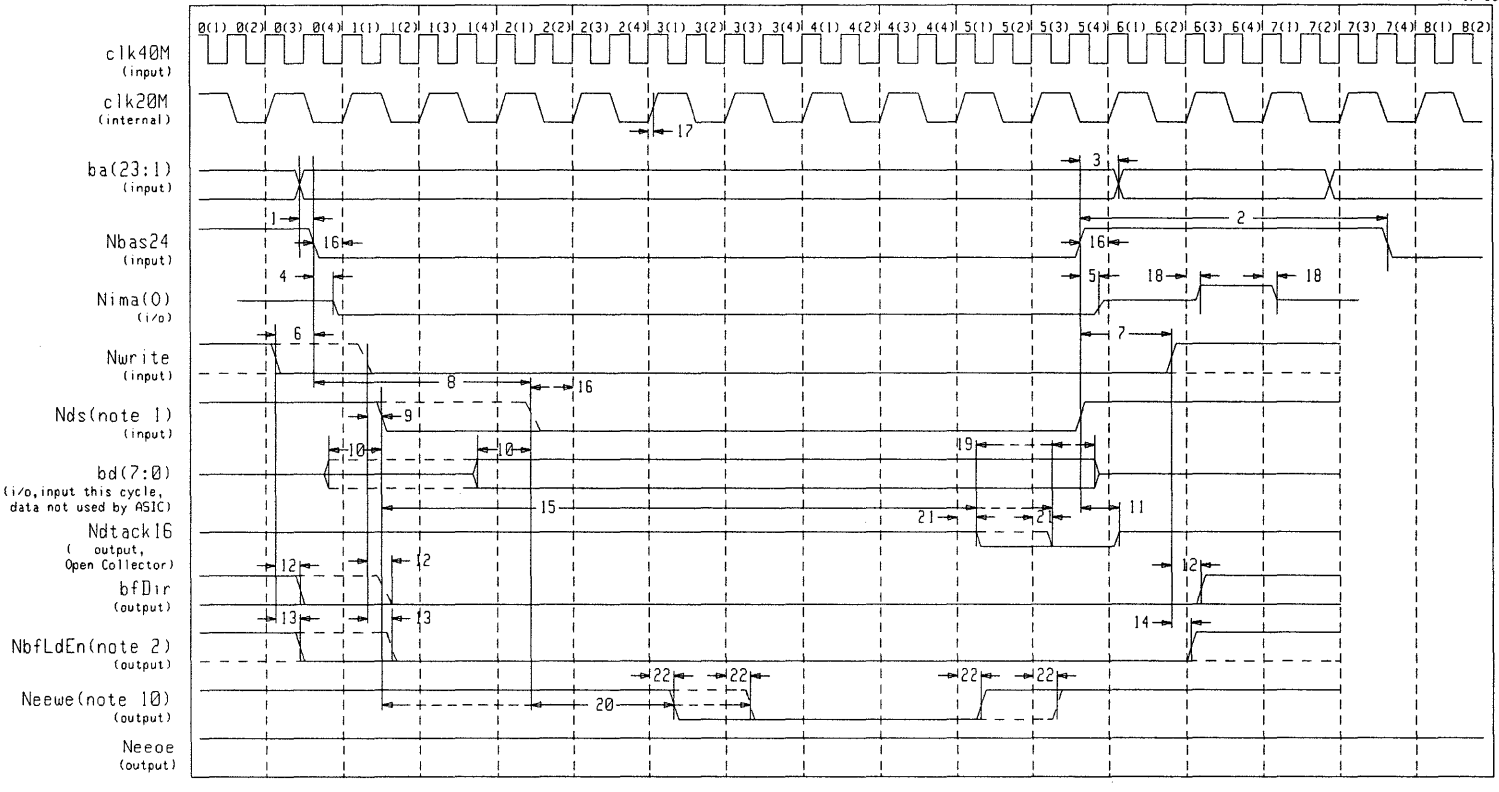
	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite high to Nbas24 or Nds low		7	
7	Nwrite hold after Nbas24 or Nds high		43	8
8	Nbas24 low to Nds low	90	-75	9
9	clk40M high to Neeoe low	35		
10	Data Setup before Dtack	15		10
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nds/Nbas24 low to buffer enable true	40		4
14	Nds/Nbas24 high to buffer enable false	20		5
15	Nbas24 low to Ndtack16 low	565	450	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	clk40M high to Ndtack16 low	60		
20	Nbas24 high to Neeoe high	30		
21	Neeoe high to ROMdata tri-state	110	0	11
22	Neeoe low to Ndtack16 low	460	365	

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) The DIO Bus Specification requires a 65 nS hold time. For Test purposes, a hold time of 43 nS is tested.
9) This is a DIO Bus Specification whose limits are not directly tested.
10) DIO Bus Specification. Neeoe to valid data from EEPROM plus buffer delay getting onto the DIO Bus must meet this spec.
11) Nbas24 high time(max) - Nbas24 to Neeoe high(max)

HEWLETT PACKARD

DIO write of EEPROM

7-17-89



	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite low to Nbas24 low	53	-75	
7	Nwrite hold after Nbas24 or Nds high		7	8
8	Nbas24 low to Nds low		50	9
9	Nwrite low before Nds low		65	
10	Data setup before Nds low		7	
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nwrite low to buffer enable true	20		4
14	Nwrite high to buffer enable false	20		5
15	Nds low to Ndtack16 low	465	300	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Data release after Ndtack16 asserted		85	11
20	Nds low to Neewe low	245	100	7
21	clk40M to Ndtack16 low	60		
22	clk40M to Neeoe change	40		

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) DIO Bus Specification. For test purposes, Nwrite can go high 7 nS before Nbas24 goes high.
9) DIO specifies 2500 ns Max, however with a programmable Bus ERROR timer, this maximum can increase.
10) A write will only take place if the EEPROM's Write Logic is 'ARMed'. See ERS.
11) DIO Bus Specification.

4. PROGRAMMABLE TIMER

4.1 Features

A Programmable Timer Module(which resembles the Motorola MC6840 Programmable Timer Module) has been integrated into BURNNI. BURNNI's Programmable Timer Module(PTM) looks the same to software as Motorola's MC6840 did in earlier Series 300 Workstations. Not all of the Motorola MC6840's features are present, however, the features used by HP-UX, Pascal Workstation(PAWs), BASIC, the BOOT ROM, and the Series 300 Test Stimulus Board Code are present. Additionally, a few new features not available in older designs using the Motorola MC6840 have been added for software testability and for new functionality needed as a result of the HP/Apollo merger. The PTM within BURNNI has the following features:

- Three 16 bit Timers which were implemented using down counters.
- The counters in all Timers can automatically re-cycle, loading their starting count value from latches.
- Two down counting modes,16-bit and dual 8-bit.
- Two of the original 8 PTM Operating Modes
  - Continuous Operating Mode where Write to Latches or PTM Software Reset Causes Counter Initialization.
  - Continuous Operating Mode where PTM Software Reset Causes Counter Initialization.
- Programmable Interrupts on Nir[6].
- Two Selectable Clock sources for Timer #1 and Timer #3. Three Selectable Clock sources for Timer #2.
- Selectable Prescaler(Divide by 8) for Timer #3(Divides either Clock Source).
- Unambiguous read of each 16 bit timer(via 2 Byte Transfers), Unambiguous read of the Timer #2/Timer #3 pair(via 2 Word Transfers).
- Programmable E clock period from .1 micro-second to 25.6 micro-seconds in .1 micro-second increments.

4.2 Power Up Initialization

After a hard system reset(NPuReset asserted), the PTM is set up to look the same as a Motorola MC6840 PTM would have in earlier HP Series 300 Workstations. The output of Timer #3(O3) is connected to the clock input to Timer #2($\overline{C2}$). The PTM itself is "reset" the same as it would have been in older systems:

- All Counter Latches and Counters are preset to $FF.
- All Control Register bits are cleared with the exception of Control Register #1, bit 0(internal reset bit) which is set.
- All Status Register bits are cleared(interrupt flags).
- E clock frequency set to 10 MHz.
- Timer #3's output is connected to Timer #2's Clock($\overline{C2}$) Input.
- Address Decoding for the PTM is enabled, assuming BURNNI is not in "Second LAN support mode(Nkbcs, Nromcs, Nledcs, Neeoe, and Nberr all tied low)".

E-clock's default value of 10 MHz is faster than previous E-clocks, however, there has not been a standard frequency for E-clocks in the past. This is the only major difference between the BURNNI implementation of the Motorola MC6840 PTM circuit after power-up reset and previous designs.

4.3 Memory Map

In the original design, the MC6840 address decode did not look at address lines 4 through 13. As a result, the MC6840 register block could be addressed using multiple addresses. In the BURNNI design, This is still the case except that address line 4 is decoded cutting in half the number of alternate addresses which can be used to access the MC6840. Accessing the BURNNI timer with address line 4 high will result in the transfer being completed normally(with $\overline{\text{DTACK}}$ asserting), the data returned will not be useful data. It is strongly recommended that only the primary addresses shown below be used when accessing the MC6840 register set:

Programmable Timer Module Register Address Map			
REGISTER ADDRESS	NOTES	REGISTER OPERATION WRITE	READ
$5F8001	1	Write Control Register #3	No Operation
	2	Write Control Register #1	
$5F8003		Write Control Register #2	Read Status Register
$5F8005		Write MSB Buffer Register	Read Timer #1 Counter
$5F8007		Write Timer #1 Latch	Read LSB Buffer Register
$5F8009		Write MSB Buffer Register	Read Timer #2 Counter
$5F800B		Write Timer #2 Latch	Read LSB Buffer Register
$5F800D		Write MSB Buffer Register	Read Timer #3 Counter
$5F800F		Write Timer #3 Latch	Read LSB Buffer Register

— Note 1: If Bit 0 of Control Register #2 is set to 0.

— Note 2: If Bit 0 of Control Register #2 is set to 1.

When one of the three Timers is read, the most significant byte of that Timer's count is returned while the least significant byte is latched in a Buffer Register. The least significant byte must be read before reading another timer or it will be overwritten. There is only one Buffer Register which is shared by the three Timers, hence the buffer register can be read at byte addresses $5F8007, $5F800B, or $5F800F. This behavior is consistent with that of Motorola's MC6840.

A new feature based on the Timer read mechanism mentioned above has been added to BURNNI, the ability to do an unambiguous read of the Timer #2/Timer #3 pair. This new feature was added by extending the unambiguous read which already existed for all 16 bit timers. By doing a word read of Timer #2(Using word address $5F8008), Timer #2's Most Significant Byte(MSB) is returned as the Least Significant Byte(LSB) of the word while Timer #3's MSB is returned in the MSB of the word. Immediately doing a word read of the Buffer Register(Using word address $5F800A, $5F8006, or $5F800E) will return Timer #2's LSB as the LSB of the word while Timer #3's LSB will be returned as the MSB of the word. Both of these values were latched when Timer #2 MSB's was read. This feature is useful when connecting Timer #3's output to Timer #2's clock input thus creating a 32 bit counter. The table below illustrates this new feature plus shows some corner cases not mentioned here:

PTM Word Mode Operation(New Feature)				
REGISTER ADDRESS	REGISTER OPERATION			
	WRITE MSB	WRITE LSB	READ MSB	READ LSB
$5F8000	Data Lost	CR #3/CR #1	Undefined Data	Undefined Data
$5F8002	Data Lost	C.R. #2	Undefined Data	Status Register
$5F8004	Data Lost	MSB Buffer Reg.	Timer #3 MSB	Timer #1 MSB
$5F8006	Data Lost	Timer #1 Latch	Timer #3 LSB Latch	LSB Buffer Reg.
$5F8008	Data Lost	MSB Buffer Reg.	Timer #3 MSB	Timer #2 MSB
$5F800A	Data Lost	Timer #2 Latch	Timer #3 LSB Latch	LSB Buffer Reg.
$5F800C	Data Lost	MSB Buffer Reg.	Timer #3 MSB	Timer #3 MSB
$5F800E	Data Lost	Timer #3 Latch	Timer #3 LSB Latch	LSB Buffer Reg.

– Note: The value of Bit 0 in Control Register #2 determines which control register(#3 or #1) will be accessed.

4.4 Control

Control of the PTM within BURNNI is the same as control of the Motorola MC6840 PTM except some features have been eliminated. The reader is encouraged to obtain a copy of Motorola's MC6840 PTM data sheet before writing any software for the PTM within BURNNI. This ERS will not go into detail on how the PTM operates.

What follows now is a list of what control bits are not present within the BURNNI implementation of the PTM. In Control Register #1, bit 7 and bit 5 are not present. Bit 7 is the Timer #1 Counter Output Enable. Bit 5 was CR15, one of three Counter Mode selection bits. Neither of these bits had an effect on past implementations of the PTM.

Control Register #2 also did not implement bits 7 and 5 for the same reasons as were mentioned for control Register #1.

Control Register #3 implemented all bits. Since Timer #3 is the only timer whose output was used in past designs, bits 7 and 5 will have an effect on Timer #3's operation.

There are four operating modes whose operation is based on the Gate inputs available on the Motorola MC6840. All past designs disabled these Gate inputs by tying them low. In BURNNI's PTM implementation, there are no Gate inputs. Thus, Four of the PTM operating modes(2 Frequency Comparison modes and 2 Pulse Width Comparison modes) become "counter disabled" modes. And except for Timer #3, There is no difference between "Continuous Operating" modes and "Single Shot" modes. No testing of "Single Shot" modes for Timer #3 are planned and the user is urged to not use this mode of operation.

All other BURNNI PTM registers have been implemented in a manner compatible with the Motorola MC6840. Interrupts generated by the PTM will be asserted on the "Nir[6]" pin on BURNNI, which is connected to the DIO bus interrupt level 6 signal line.

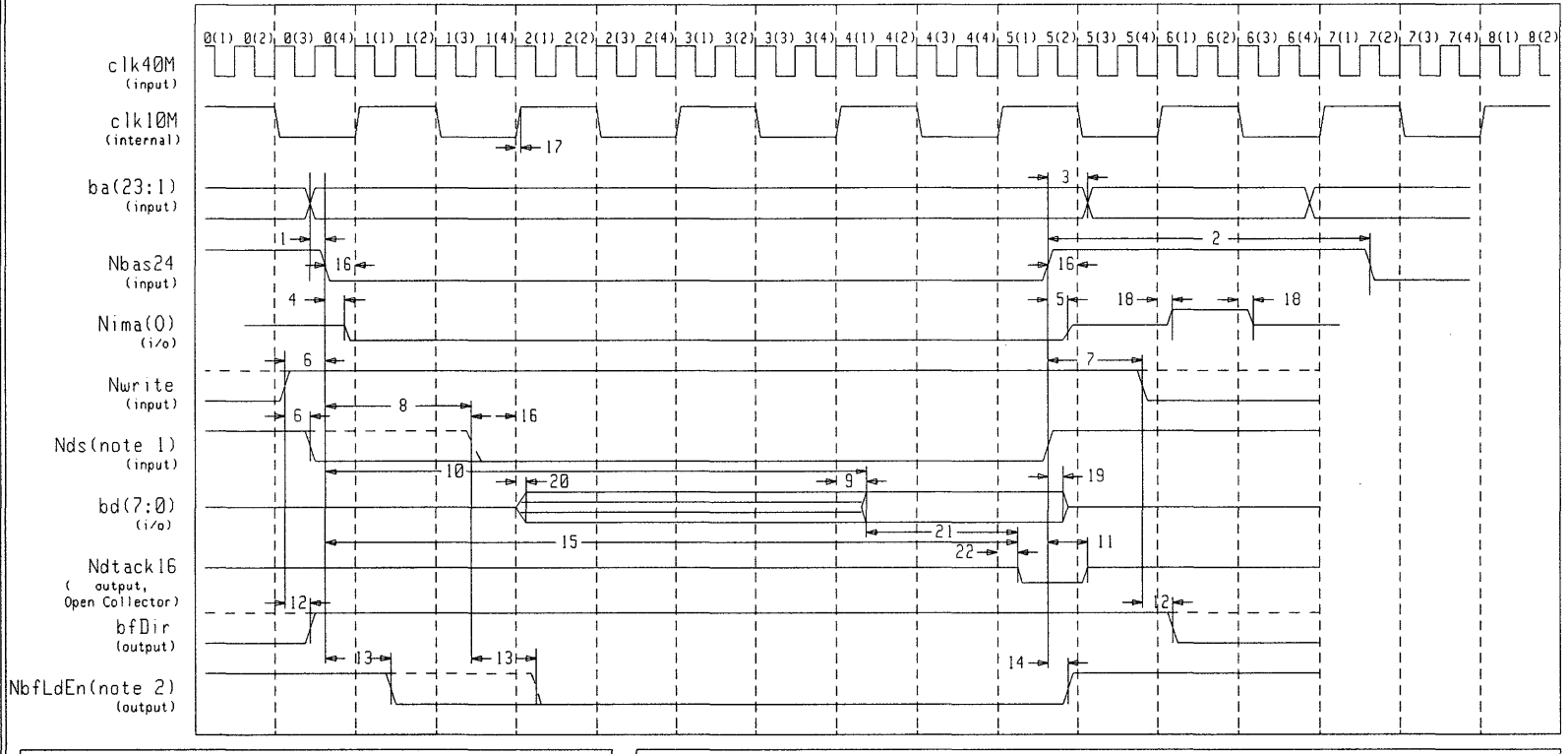
4.5 Timing

The following two figures specify READ and WRITE cycles to BURNNI's PTM. Synchronous timing shown is intended for IC test purposes only. "INTERNAL" signals are also shown to assist in IC test vector generation generation. PC board designers should use the asynchronous timing.

HEWLETT PACKARD

DIO read of 6840 PTM Registers

12-5-89



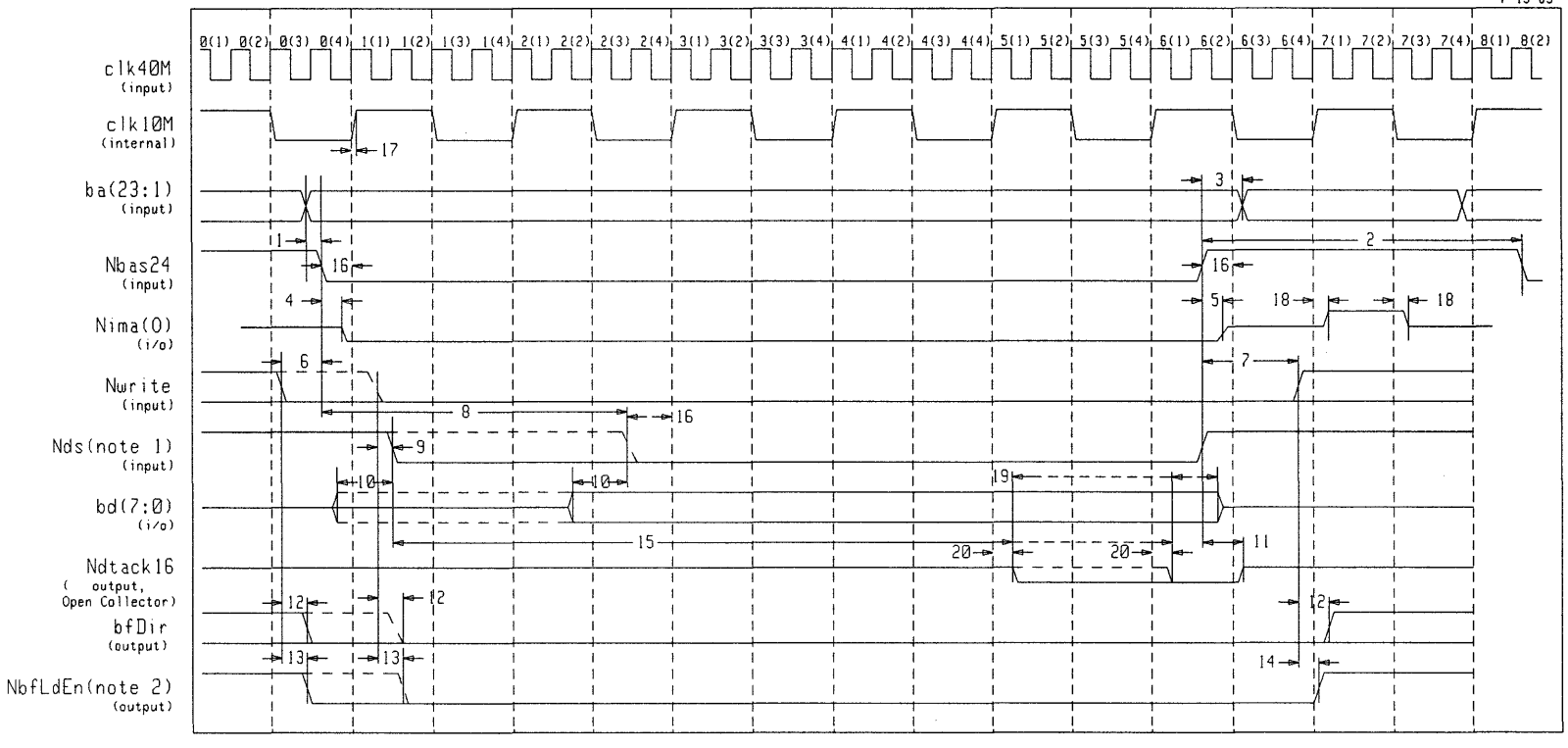
	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite high to Nbas24 or Nds low		7	
7	Nwrite hold after Nbas24 or Nds high		43	8
8	Nbas24 low to Nds low	90	-75	9
9	clk40M high to valid data	60		
10	Nbas24 low to valid data	465	300	7
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nds/Nbas24 low to buffer enable true	40		4
14	Nds/Nbas24 high to buffer enable false	20		5
15	Nbas24 low to Ndtack16 low	565	400	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Nbas24 high to data bus tri-state	25		
20	clk40M to Data Bus driven		0	
21	Data set up before Ndtack16 low		40	
22	clk40M to Ndtack16 low	60		

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) Tester will hold Nwrite for 43nS, which exceeds DIO requirements.
9) This is a DIO Bus Specification whose limits are not directly tested.

HEWLETT PACKARD

DIO write of 6840 PTM Registers

7-13-89



	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite low to Nbas24 low	53	-75	10
7	Nwrite hold after Nbas24 or Nds high		7	8
8	Nbas24 low to Nds low		50	9
9	Nwrite low before Nds low		65	
10	Data setup before Nds low		7	10
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nwrite low to buffer enable true	20		4
14	Nwrite high to buffer enable false	20		5
15	Nds low to Ndtack16 low	465	300	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Data release after Ndtack16 asserted		85	10
20	clk40M to Ndtack16 low	60		

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) DIO Bus Specification. For test purposes, Nwrite can go high 7 nS before Nbas24 goes high.
9) DIO specifies 2500 ns Max, however with a programmable Bus ERROR timer, this maximum can increase.
10) DIO Bus Specification.

5. BOOT ROM and TEST LED Support

5.1 BOOT ROM and TEST LED Signals

BOOT ROM support is provide by one signal, "Nromcs". This signal becomes true(low) whenever a valid BOOT ROM address is decoded, assuming that BOOT ROM address decoding is enabled. Address decoding is disabled by forcing "Nromcs" low during the power-up reset configuration time window(0 to 400 nS after "NPuReset" goes high). If "Nromcs" is high during the configuration time window, address decoding will be enabled. An internal pull-up on the "Nromcs" will cause "Nromcs" to float high during the 120 mS(DIO Bus Specification) time period for which "NPuReset" must be low.

Valid BOOT ROM addresses are dependent on bit 0 of Register 11(Byte Address $0040000B) of the "New I/O Capabilities" Register set. Refer to Section 3.7(PTM/Expanded BOOT ROM Control Register) for more information on this Register. When bit 0 of Register 11 is cleared(default), valid BOOT ROM addresses are $00000000 through $0001FFFF. Setting bit 0 of Register 11 enables additional address space. The new range of valid BOOT ROM addresses becomes $00000000 through $0003FFFF.

BURNNI also generates $\overline{\text{DTACK}}$ for both Reads and Writes to valid BOOT ROM addresses. $\overline{\text{DTACK}}$ timing for both of these cycles can be found in the next section. "Nromcs" DOES NOT assert during Write cycles, however, "Nledcs" may.

"Nledcs" is the TEST LED support signal. The rising edge of this signal is normally used to latch the ODD(or Lower) data byte(DIO data bus lines BD7 through BD0) into a latch whose outputs are tied to LEDs. "Nledcs" is enabled or disabled in the same way "Nromcs" is used to enable or disable BOOT ROM address decoding. Disabling "Nledcs" does NOT necessarily disable $\overline{\text{DTACK}}$ generation for writes to valid TEST LED addresses since these are also valid BOOT ROM addresses. If BOOT ROM address decoding is disabled, then disabling TEST LED decoding will prevent $\overline{\text{DTACK}}$ from being asserted by BURNNI for valid BOOT ROM/TEST LED addresses.

Valid TEST LED addresses are addresses less than or equal to $0001FFFF which also have Address Line BA14 high. If BA14 is low, "Nledcs" will never assert.

BURNNI provides one additional signal for BOOT ROM/TEST LED support, "Nbdrv". "Nbdrv" should be asserted no later than 100 nS after Address Strobe(Nbas24) has asserted. By asserting "Nbdrv", BURNNI's BOOT ROM/TEST LED cycle is aborted and the external data bus buffers are disabled from driving onto the DIO Data bus. "Nbdrv" is the DIO "BUS DRIVE DISABLE($\overline{\text{BDRV}}$)" signal. See the "DIO-II BUS SPECIFICATION"(HP Drawing No: A-5959-0036-1) for more information about the $\overline{\text{BDRV}}$ signal.

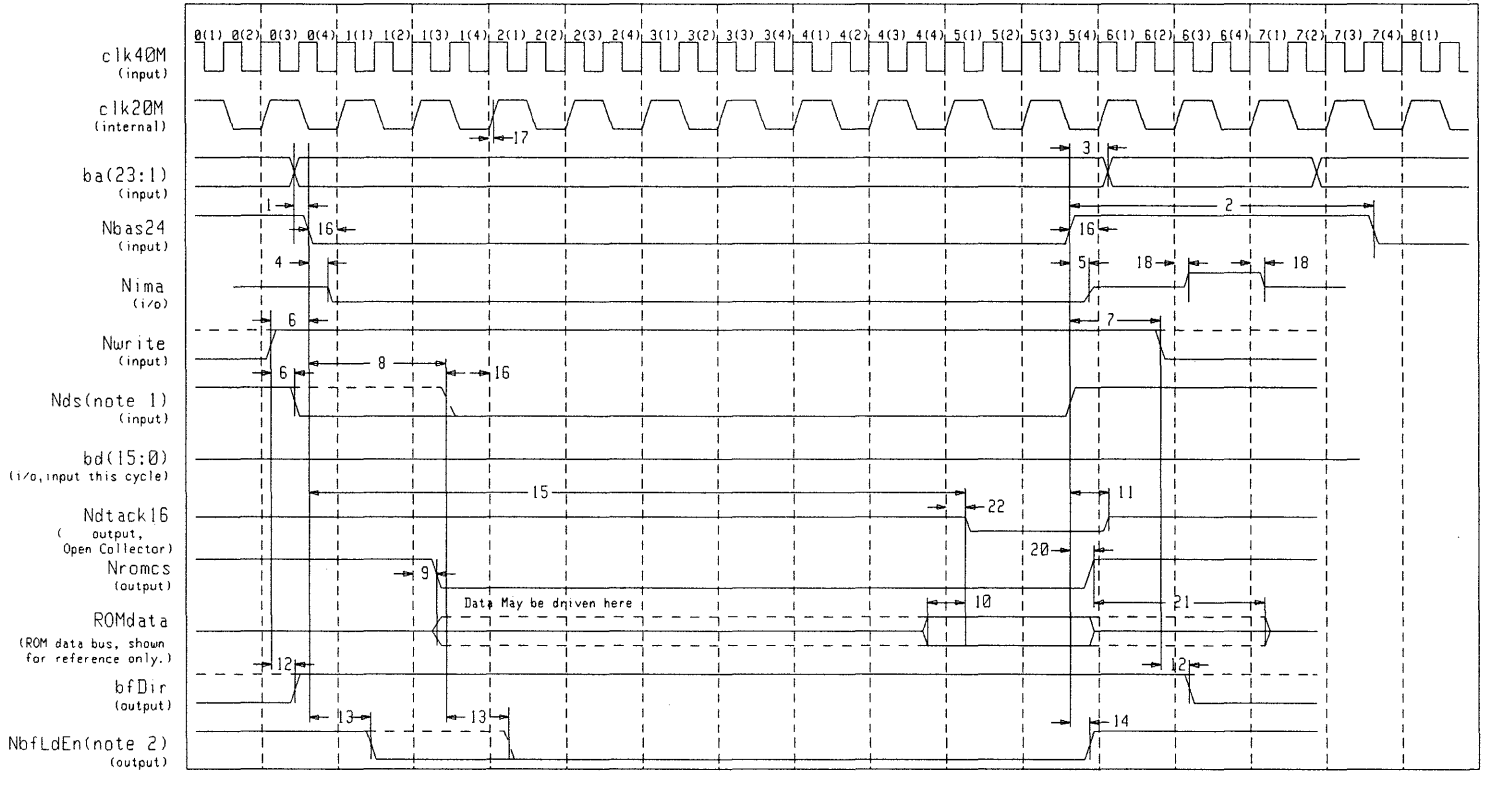
5.2 Timing

The following two figures specify READ and WRITE cycles supported by BURNNI to a ROM whose starting address is $00000000. WRITE cycles can be of two types: 1) To a latch which is used to display system status on LEDs and 2) To no device(used by the BOOT process for exception processing. Synchronous timing shown is intended for IC test purposes only. "INTERNAL" signals are also shown to assist in IC test vector generation generation. PC board designers should use the asynchronous timing.

HEWLETT PACKARD

DIO read of BOOT ROM

7-7-89



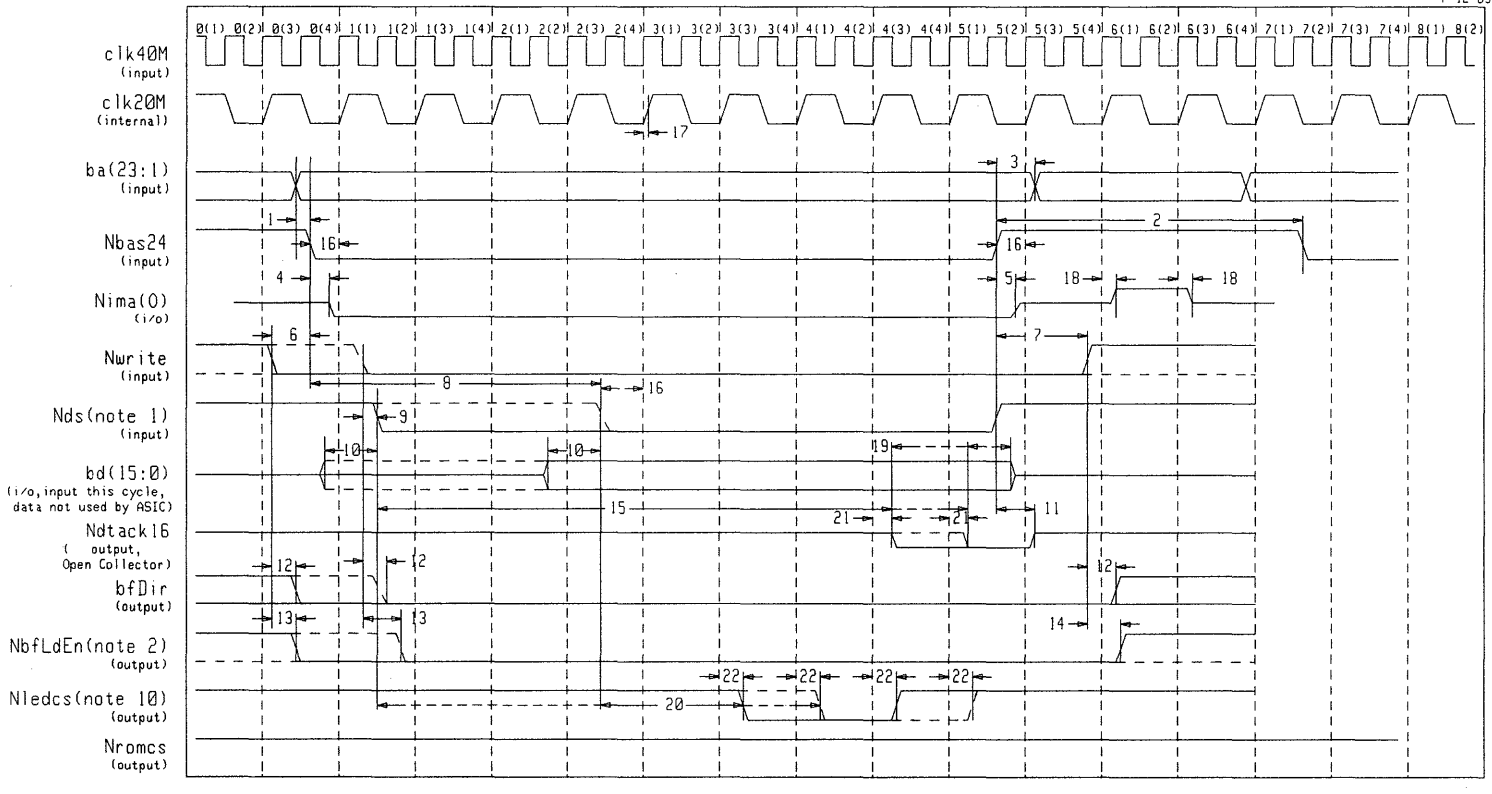
	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite high to Nbas24 or Nds low		7	
7	Nwrite hold after Nbas24 or Nds high		43	8
8	Nbas24 low to Nds low	90	-75	9
9	clk40M high to Nromcs low	35		
10	Data Setup before Dtack	15		10
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nds/Nbas24 low to buffer enable true	40		4
14	Nds/Nbas24 high to buffer enable false	20		5
15	Nbas24 low to Ndtack16 low	530	400	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	clk40M high to data bus tri-state	25		
20	Nbas24 high to Nromcs high	30		
21	Nromcs high to ROMdata tri-state	110	0	11
22	clk40M to Ndtack16 low	60	0	

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low true) also represents the timing of bfHdEn(High true) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) The DIO Bus Specification requires a 65 nS hold time.
9) This is a DIO Bus Specification whose limits are not directly tested.
10) DIO Bus Specification which the BOOT ROM will need to meet.
11) This Specification is to be met by the BOOT ROM.

HEWLETT PACKARD

DIO write of TEST LEDs or BOOT ROM address space

7-12-89



	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite low to Nbas24 low	53	-75	
7	Nwrite hold after Nbas24 or Nds high		7	8
8	Nbas24 low to Nds low		50	9
9	Nwrite low before Nds low		65	
10	Data setup before Nds low		7	
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nwrite low to buffer enable true	20		4
14	Nwrite high to buffer enable false	20		5
15	Nds low to Ndtack16 low	410	200	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Data release after Ndtack16 asserted		85	11
20	Nds low to Nledcs low	285	100	
21	clk40M to Ndtack16 low	60	0	
22	clk40M to Nledcs change	35	0	

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn also represents the timing of NbfHdEn for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) DIO Bus Specification. For test purposes, Nwrite can go high 7 nS before Nbas24 goes high.
9) DIO Bus Specifies 2500 ns Max, however with a programmable Bus ERROR timer, this maximum can increase.
10) Nledcs remains high if the address written is not a 'Test LED' address(which is part of the BOOT ROM address space).
11) DIO Bus Specification.

HEWLETT PACKARD

6. Keyboard System Support

6.1 Keyboard Signals

BURNNI provides support for keyboard by providing interface signals for an Intel 8042(or family member) Peripheral Processor. In Series 300 systems, the 8042 Peripheral Processor is the system keyboard interface, as well as the battery backed real time clock interface and the sound generator or beeper interface. The 8042 Peripheral Processor also has a variety of timers and/or counters programmed into it plus a time-of-day clock(which gets initialized from the battery backed real time clock). For more information on the 8042 Peripheral Processor programmed as the Series 300 keyboard interface, see document A-1820-4784-2.

Interface signals provided for the 8042 Peripheral Processor are:

- clk5M -- A 5 MHz clock.
- Nclk5M -- Complement of the clk5M 5 MHz clock.
- kbRs0 -- 8042 Register Select 0(Latched DIO address BA1).
- Nkbcs -- Chip Select for 8042.
- NkbIor -- Read Strobe for 8042.
- NkbIow -- Write Strobe for 8042.
- NkbReset -- Reset for 8042.

These signals meet Intel's timing requirements for the 8042. The next section contains timing diagrams for all the above signals except "NkbReset". "NkbReset" is triggered by "Nreset" asserting or "NPuReset" asserting. "NkbReset", once asserted, is guaranteed to stay asserted a minimum of 25 micro-Seconds, which meets the 8042 peripheral's requirements.

To disable the keyboard support(Address Decoding), tie "Nkbcs" low. The logic value on this pin is sampled from 200 nS to 400 nS after "NPuReset" goes high. If "Nkbcs" is low, then address decoding for the 8042 is disabled and "Nkbcs" will remain tri-stated. If "Nkbcs" is allowed to float high(with help from an internal pull up resistor built into the I/O pad), keyboard address space decoding will be enabled and "Nkbcs" will start driving high from 200 nS to 400 nS after "NPuReset" goes high. In this case, a valid keyboard address($420000 through $42FFFF) will cause "Nkbcs" to assert as shown in the timing diagrams in the next section. "NPuReset" asserting always tri-states "Nkbcs".

Although a number of signals are provided, these are not all the signals needed to support the 8042 keyboard interface. The keyboard interface can generate two different interrupts, one intended to be driven on DIO interrupt level 1 and the other intended to be driven on DIO interrupt level 7. The signal intended to be driven on DIO interrupt level 1 is HIGH true and must be inverted before being driven using an open collector gate(an inverting open collector gate would do both functions). The signal intended to be driven on DIO interrupt level 7 is, however, LOW true. This signal must be driven on DIO interrupt level 7 with an open collector gate. Additionally, the value of this signal must be reported in bit location 2 of the internal HP-IB interface Register #5(Byte Address $470005). Another ASIC code named NIKKI(1LY5) implemented the internal HP-IB interface and hence has a pin which can be connected to the 8042's interrupt level 7 interrupt. If NIKKI is not used in the system, Register #5 of the internal HP-IB interface must be constructed using external logic.

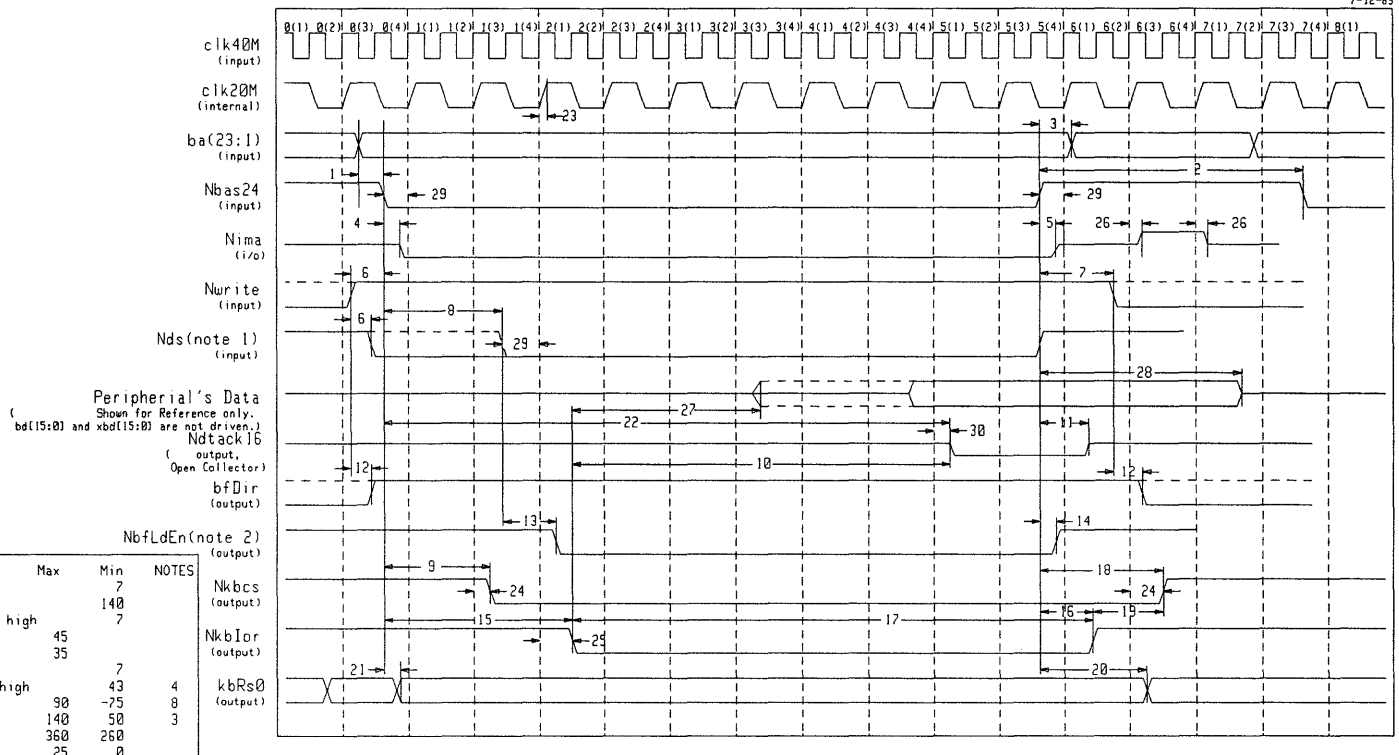
A WORD OF CAUTION: When constructing Register #5, be careful with address decoding. Register #5 is shown as DIO address $470005, however, Series 300 software really uses DIO address $478005. In other words, DO NOT decode DIO address lines BA5 through BA15.

6.2 Timing

The following two figures specify READ and WRITE cycles supported by BURNNI to an Intel 8042 Peripheral Processor. Synchronous timing shown is intended for IC test purposes only. "INTERNAL" signals are also shown to assist in IC test vector generation generation. PC board designers should use the asynchronous timing.

HEWLETT PACKARD

DIO read of 8042



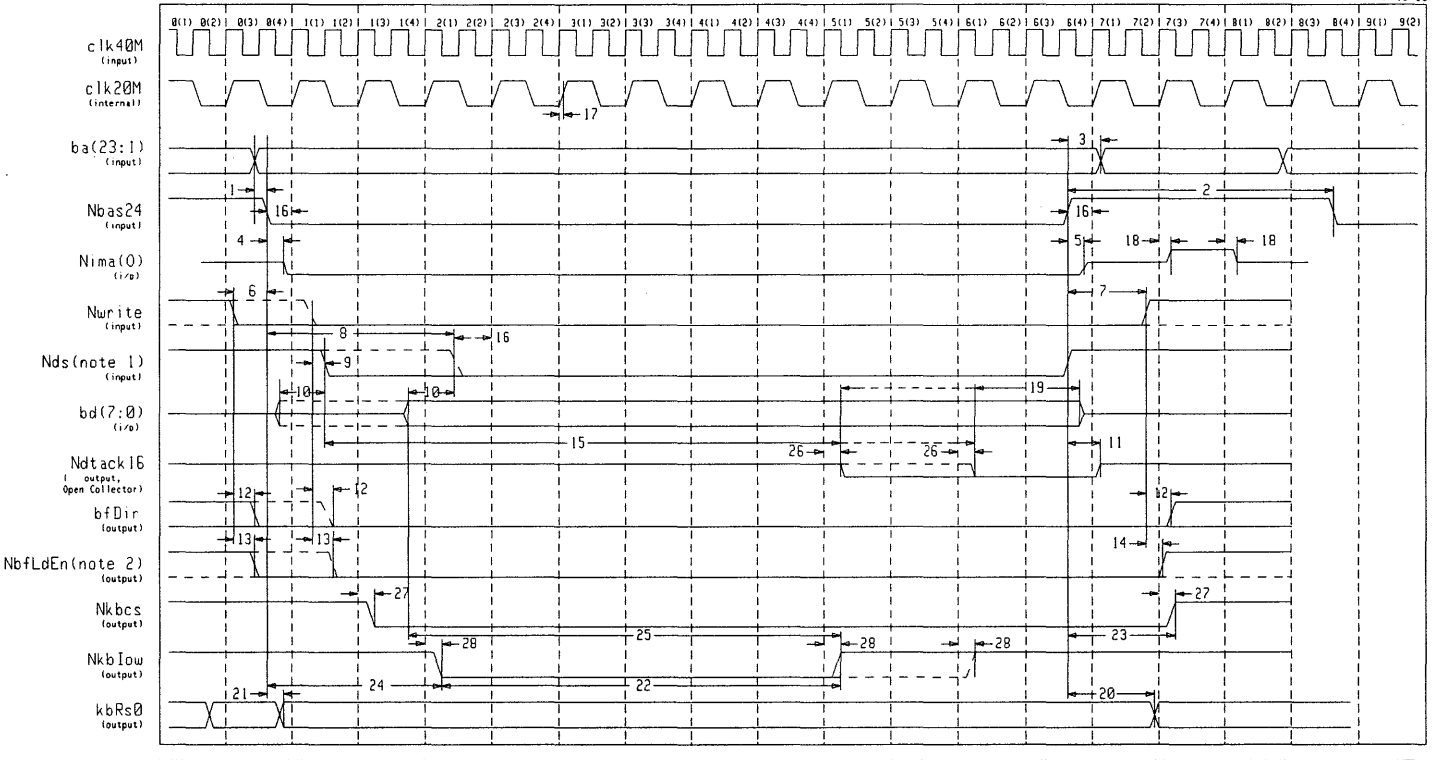
	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite high to Nbas24 or Nds low		7	
7	Nwrite hold after Nbas24 or Nds high		43	4
8	Nbas24 low to Nds low	90	-75	8
9	Nbas24 low to Nkbcs low	140	50	3
10	NkbIor low to Ndtack low	360	260	
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir valid	20		
13	Nds/Nbas24 low to buffer enable true	40		
14	Nds/Nbas24 high to buffer enable false	20		
15	Nbas24 low to NkbIor low	195	100	3
16	Nbas24 high to NkbIor high	45		
17	NkbIor low time		400	
18	Nbas24 high to chip enable high	135	50	
19	NkbIor high to Nkbcs high		5	
20	kbRs0 hold after Nbas24 high	130	50	6
21	Nbas24 to valid kbRs0	20		
22	Nbas24 low to Ndtack16 low	515	400	3
23	clk40M to clk20M skew	15	0	
24	clk40M to Nkbcs change	35		
25	clk40M to NkbIor low	40		
26	clk40M to signal change	30		
27	NkbIor to 8042 valid data	130		9
28	Nbas24 high to 8042 Data tri-state	130		5
29	Asynchronous Set Up Time		15	7
30	clk40M to Ndtack16 low	60		

NOTES:
1) Nds is the logical 'or' or Nlds and Nuds
2) NbfLdEn(Low True) also represents the timing of bfHdEen(High True) for word and upper byte accesses.
3) This parameter is specified for system designers who will not reference their signals to clock edges.
4) The DIO Bus Specification requires a 65 nS hold time. For Test purposes, a hold time of 43 nS is tested.
5) This spec. is the time from Nbas24 going high to NkbIor going high plus 85 nS for the 8042 to tri-state.
6) The max spec. is the worst case time it takes for the address latch to become transparent(assuming synchronizer miss).
7) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
8) This is a DIO Bus Specification whose limits are not directly tested.
9) 8042 parameter shown for reference only.

HEWLETT PACKARD

DIO write to 8042

7-13-89



	TIMING PARAMETER(nS)	Max	Min	NOTES
1	Address setup before Nbas24 low		7	
2	Nbas24 high time		140	
3	Address hold after Nbas24 or Nds high		7	
4	Nbas24 low to Nima low	45		
5	Nbas24 high to Nima tri-state	35		
6	Nwrite low to Nbas24 low	53	-75	
7	Nwrite hold after Nbas24 or Nds high		7	8
8	Nbas24 low to Nds low		50	9
9	Nwrite low before Nds low		65	
10	Data setup before Nds low		7	
11	Nbas24 high to Ndtack16 release	25	0	
12	Nwrite change to bfDir change	20		
13	Nwrite low to buffer enable true	20		4
14	Nwrite high to buffer enable false	20		5
15	Nds low to Ndtack16 low	465	350	7
16	Asynchronous Set Up Time		15	3
17	clk40M to Internal clock skew	15		6
18	clk40M to signal change	30		
19	Data release after Ndtack16 asserted		85	10
20	kbRs0 hold after Nbas24 goes high	130	50	11
21	Nbas24 low to valid kbRs0	20		
22	NkbIow low time		230	
23	Nbas24 high to Nkbcs high	140	50	7
24	Nbas24 low to NkbIow low	190	100	7
25	Data setup before NkbIow high		300	7
26	clk40M to Ndtack16 low	60		
27	clk40M to Nkbcs change	35		
28	clk40M to NkbIow change	35		

NOTES:
1) Nds is the logical 'or' of Nlds and Nuds.
2) NbfLdEn(Low True) also represents the timing of bfHdEn(High True) for word and upper byte accesses.
3) This parameter is only specified for test purposes. The ASIC will synchronize this normally asynchronous signal.
4) This delay is from which ever signal went low last, Nbas24 or Nds(See Note 1).
5) This delay is from which ever signal went high first, Nbas24 or Nds(See Note 1).
6) This parameter not directly measurable.
7) This parameter for designers not referencing their signals to the clock.
8) DIO Bus Specification. For test purposes, Nwrite can go high 7 nS before Nbas24 goes high.
9) DIO Bus Specifies 2500 ns Max, however with a programmable Bus ERROR timer, this maximum can increase.
10) DIO Bus Specification.
11) The max. spec. is the worst case time it takes for the address latch to become transparent(assuming synchronizer miss).

HEWLETT PACKARD

7. Buffer Control

7.1 Control Signals

For BURNNI, all host interface signals assume DIO/DIO-II(HP Drawing No.: A-5959-0036-1) bus timing. This was done in order to allow BURNNI to be designed into different products with different processor clock frequencies. The DIO/DIO-II bus has the only set of common signals and timing used by the initial set of products for which BURNNI was designed.

With DIO/DIO-II timing assumed, data bus buffer control was necessary for BURNNI to provide a more complete interface. Four data bus buffer control signals are provided, one for direction control and three for buffer enables. Two additional input signals are also provided so that buffer control will work with other I/O devices(including DMA) attached to the local data bus.

"bfDir" is the data bus buffer direction control signal. This signal is asynchronously generated from two signals, the DIO/DIO-II bus signal BR/ $\overline{W}$ ("Nwrite" pin) and "NdmaActive", one of the two additional input signals needed for DMA support. The logic equation(in PALASM format with pin names in upper case characters) for "bfDir" is:

```
BFDIR = NWRITE * NDMAACTIVE
     + /NWRITE * /NDMAACTIVE
```

"bfDir" will lag changes in "Nwrite" and/or "NdmaActive" by 20 nS maximum.

"NbfLdEn" is the lower DIO data bus(BD0 through BD7) buffer enable. This signal is LOW true and is asynchronously generated. The logic equation for "NbfLdEn in PALASM(pin names in upper case characters) format is:

```
NBFLDEN = NLDS * NWRITE * NDMAACTIVE
        +  NLIMA * NIIMA * NWRITE * NDMAACTIVE
        +  NWRITE * NBAS24 * NBAS32 * NDMAACTIVE
        + /NWRITE * NBAS24 * NBAS32 * /NDMAACTIVE
```

NLIMA , and NIIMA are internal signals which are related to the signal on the "Nima" pin. NLIMA is the LOW true "local" I aM Addressed. This signal will be driven out on the "Nima" pin. NIIMA is the actual signal on the "Nima" pin. "Nima" is an I/O pin which is asserted when a local I/O device is addressed. BURNNI pre-charges this signal high from 50 to 75 nS after "Nbas24" or "Nbas32" goes high. If an I/O device within BURNNI or directly supported by BURNNI is addressed, BURNNI will drive "Nima" low. If a device other than BURNNI is addressed, it should assert "Nima" low in order for BURNNI to properly control the data bus buffers. This device must quit driving "Nima" within 50 nS after either "Nbas24" or "Nbas32" goes high.

"bfHdEn" is the upper DIO data bus(BD8 through BD15) buffer enable. This signal is HIGH true and is asynchronously generated. This signal is high true to support DMA, which requires that "bfHdEn" be gated with $\overline{FOLD}$ (using a 2-input NAND gate). The LOW true logic equation for "bfHdEn" in PALASM(pin names in upper case characters) format is:

```
NBFHDEN = NUDS * NWRITE * NDMAACTIVE
        +  NLIMA * NIIMA * NWRITE * NDMAACTIVE
        +  NWRITE * NBAS24 * NBAS32 * NDMAACTIVE
        + /NWRITE * NBAS24 * NBAS32 * /NDMAACTIVE
```

"NbfXdEn" is the DIO-II data bus(XBD0 through XBD15) buffer enable. This signal is LOW true and is

asynchronously generated. The logic equation for "NbfXdEn" in PALASM(pin names in upper case characters) format is:

```
NBFXDEN = NLIMA * NIIMA * NWRITE * NDMAACTIVE
        +  NWRITE * NBAS24 * NBAS32 * NDMAACTIVE
        +  NWRITE * NLWORD * NDMAACTIVE
        + /NWRITE * NBAS24 * NBAS32 * /NDMAACTIVE
        + /NWRITE * NLWORD * /NDMAACTIVE
```

HEWLETT PACKARD

8. TEST MODES

8.1 Introduction

In addition to the normal mode of operation, there are two test modes supported by BURNNI. In one mode, a number of BURNNI's flip-flops are connected in a serial chain so that their state information can be accessed serially. ATG(Automatic Test Generation) software was used to test BURNNI in this mode. The other test mode is aimed more at the ability to test PC boards into which BURNNI is loaded. In this mode, most of the outputs are tri-stated thus allowing the PC board tester to supply it's own signals. The "Scan Path" and "PC Board Test Support" sections will describe both test modes in more detail.

8.2 Initialization

Normally, asserting "NPuReset" is sufficient to put BURNNI in a known state, however, there are self synchronizing state machines which do not see "NPuReset". These state machines are responsible for converting 40 MHz to 20 MHz, 10 MHz, and finally 5 MHz which is driven outside BURNNI. 20 MHz is also used internally to "filter" the NPuReset signal so that the chances of a "glitch" causing a RESET is minimized. It is necessary for these state machines to operate while "NPuReset" is asserted(external logic usually requires a valid clock signal in order to be properly RESET). This causes a problem for IC testers(or testers in general). In order to test BURNNI, a tester must be able to put all flip-flops in a known state. As far as the tester is concerned, the state machines which generate the clocks can never be in a known state.

As a simple example, take a 2-state state machine for dividing 40 MHz down to 20 MHz. This simple state machine has two states, "1" and "0". If this state machine powered up in the "0" state, the next state will be "1". If the state machine powered up in the "1" state, the next state will be "0". As far as the real world is concerned, ether starting state will produce a correct signal(assuming 40 MHz clock cycle number to phase of 20 MHz clock is not important!). For a tester, the state machine powers up in the "X" state. The next state will be " $\overline{X}$ " and the following state will be "X" again. BURNNI solved this problem by allowing these state machines to be initialized in TEST mode("Ntest" low) only.

The following figure shows how a tester should initialize BURNNI. This figure also introduces the 40 MHz clock numbering scheme used on all other timing diagrams. This scheme numbers each 40 MHz clock cycle with two integers, the first integer is the 10 MHz clock cycle number followed in parenthesis by the 40 MHz clock cycle number within that 10 MHz cycle(note that the second number takes on only values 1, 2, 3, or 4). All test vectors must keep track of this information!

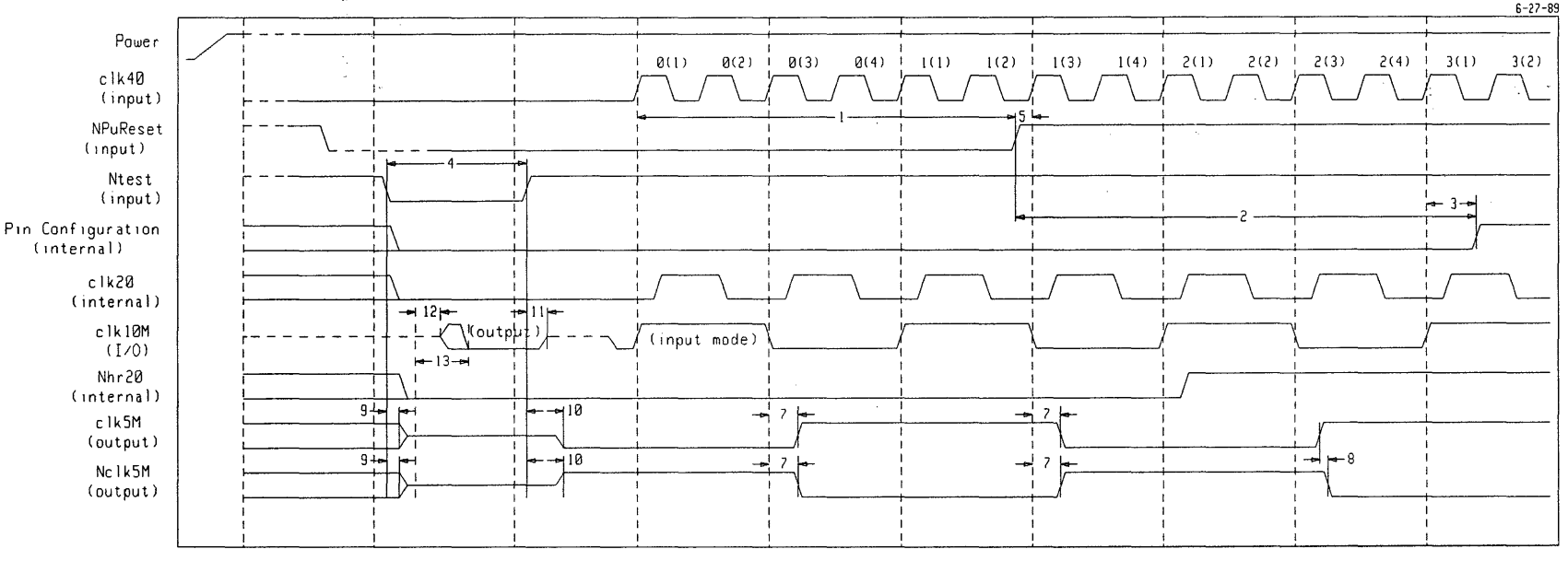
8.3 Scan Path

The "Scan Path" test mode is used to initialize all flip-flops in the scan chain by serially shifting in the initialization(present state) pattern. Next state information is then generated by temporarily reverting back to parallel mode and clocking once. The next state information is now serially shifted out while new present state information is serially shifted in. ATG(Automatic Test Generation) software is available which uses this mode of operation to build a test vector set with a high degree of test coverage.

When in "Scan Path" test mode, BURNNI can not be operated at its maximum speed. This is partially due to the delay caused by the logic involved in forcing all clocks to be the same. It is also due to the delays built into the serial data path between logic blocks which normally have different clocks to prevent data set-up and hold problems. A large amount of logic functionality which would be difficult(sometimes impossible) to test with parallel vectors is tested by ATG vectors.

HEWLETT PACKARD

Tester Initialization of BURNNI



	TIMING PARAMETER(nS)	Max	Min	
1	Power Up Reset Assertion Time	-	5clk+10nS	
2	Pin Configuration after Nponin	400	200	
3	Input clock edge to pin state change	30	0	
4	Ntest low time		50	
5	Asynchronous set-up time		15	
6	Clock rising Edge to 10 MHz Clock Edge	30	0	(Not Shown, See Note 3)
7	Clock rising edge to 5 MHz Clock Edge	35	0	
8	clk5M Clock edge to Nclk5M Clock edge skew	+-5	0	
9	Ntest low to tri-state(becomes an input)	40	0	
10	Ntest high to signal driven	40	0	
11	Ntest high to tri-state(becomes an input)	40	0	
12	Ntest low to signal driven	40	0	
13	Ntest low to clk40M signal valid on clk10M pin	50	0	

Notes: 1) Some output pins become inputs when NPuReset is asserted. The Logic value on these pins is then sampled with the timing shown for "Pin Configuration" above. The sampled pin will then remain an input or will become an output. Pins whose function is determined by the logic value on them at "Pin Configuration" time are Nkbcs, Nromcs, Nledcs, Neeoe, and Nberr. InAle is also sampled by the "Pin Configuration" signal resulting in LAN support being disabled if this pin is low.

2) "Pin Configuration" is the internal signal NDhr20, the synchronized NPuReset signal delayed one 20 MHz state.

3) clk10M is normally an input. When Ntest is driven low, Ntest becomes an output which drives whatever signal is present on clk40M.

Five pins are used to enable and control BURNNI in "Scan Path" test mode. Four of these pins have other functions when not in "Scan Path" test mode. These pins are described in the following paragraphs. Two additional pins are not normally asserted while in "Scan Path" mode doing "Scan Path Testing", "oe" and "NPuReset". "oe" tri-states most of the output pins(some LAN circuit pins are not tri-stated) while "NPuReset" asynchronously clears many of the flip-flops(this feature used during tester initialization).

Asserting "Ntest"(low) puts BURNNI into "Scan Path" test mode. As part of being in this mode, all internal clocks within BURNNI are forced to reflect the signal on the "clk40M" pin. Additionally, "clk10M"(normally an input) becomes an output driving the scan clock(signal applied to "clk40M") off chip and also driving the scan clock back into the LAN support logic. Multiplexer circuits are used in other locations to switch between the normal clock and the scan clock. Care must be taken when "Ntest" changes logic states since the Multiplexer circuits can cause a "glitch" to occur. With the "clk10M" pin also changing from 'receive only' to both 'driving' and 'receiving', the LAN support circuits may also see an invalid clock.

There are other effects of "Ntest" being asserted(driven low). One of these effects is that all "highbias" inputs to the pads are driven low thus disabling the pad pull-ups. A second effect is that the meaning of "Nbdrv" changes such that it becomes "scanin", the serial input data port. "Nclk5M" and "clk5M", which are normally outputs, become inputs with "clk5M" becoming the "masken" signal. "Nclk5M", although becoming an input, is not connected to any logic.

Once in "Scan Path" mode("Ntest" low), "Ndtkall" becomes the "scan enable"("scanen") signal. Taking this pin high puts BURNNI into the serial mode of operation which is used to serially shift data into or out from the flip-flops in the scan chain. Taking "Ndtkall" low puts BURNNI into the parallel mode of operation in which all flip-flops are no longer connected in a serial chain. The flip-flops are now connected as they would be during normal operation(except all flip-flops now share a common clock due to "Ntest" being low).

When "Ndtkall" is high, "Nbdrv" is the input data port for the scan chain("scanin" signal) and "NkbIow" is the output data port for the scan chain("scanout" signal). When "Ndtkall" is low, "NkbIow" reverts back to its original meaning. "Nbdrv" has no function when "Ndtkall" is low.

The last "Scan Path" mode signal is "masken", which is tied to the "clk5M" pin. Driving a 1 on this pin forces "NkbIow"("scanout") high thus masking an "UNKNOWN" or "DON'T CARE"("X") signal which the tester may think is present on the "NkbIow" pin. This is desirable since some(all?) testers take longer to test "X" logic values than it does to test either a "1" or a "0". Driving a "0" on "clk5M" allows the real logic value "NkbIow" to present on the "NkbIow" pin(either "NkbIow" or "scanout" information).

8.4 PC Board Test Support

PC Board test support is provided by the "oe" signal. During normal operation, "oe" is either tied high through a resistor(or allowed to float high, the "oe" pin has the "highbias" input to the pad enabled). When a PC board tester needs to override signals from BURNNI, it takes the "oe" low and most outputs are asynchronously tri-stated. A few LAN subsystem signals are exceptions. The following signals are not guaranteed to tri-state when "oe" is driven low:

- "Sad" bus
- "lad" bus(See Note below)
- "NlnBusEn"
- "lnread"

- "Nlance"
- "Nlnrdy"

Note: If the LAN state machine has seen a few "clk10M" cycles, "oe" will tri-state the "lad" bus. However, there is a corner case where it is possible to get the LAN state machines into a state for which "oe" does not tri-state the "lad" bus.