A PAL Equations

The **UNIX** PC uses four programmable array logic (PAL) integrated circuits: the disk, arbitor, and memory management unit PALs are shown on sheet 2; the data separator PAL is shown on sheet 12.

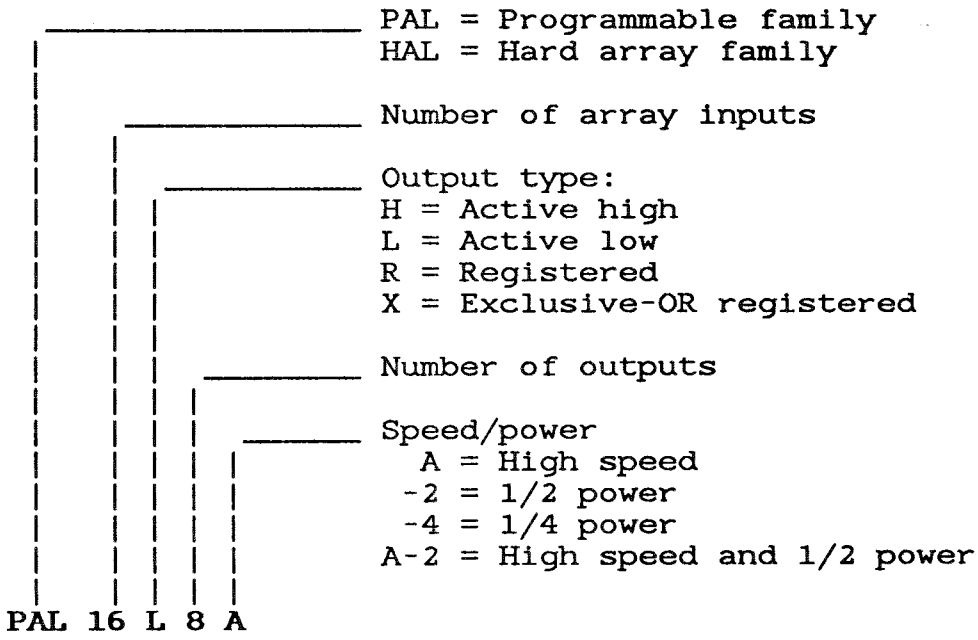
The operation of a PAL is described by a logic equation in a PAL listing. The listing is used to determine the correct logic levels of the outputs with a given set of inputs.

PROM's have been widely used by system designers to implement firmware. The PAL extends this programmable flexibility by utilizing fusible link technology to implement logic functions.

The PAL implements the familiar sum of products logic by using a programmable AND array whose outputs feed a fixed OR array. Since the sum of products form can express any Boolean transfer function, the PAL circuit uses are limited only by the number of terms available in the AND/OR arrays. PALs come in different sizes to allow for effective logic optimization.

PALs can be programmed in most standard PROM programmers with the addition of a PAL personality card.

The PAL identification number is broken down as follows:



Arbitor: PAL 16R8A (Sheet 2: C-6)

Pin Signal Names

Pin Number	Mnemonic	Description
1	PCK*	Input: processor clock
2	RFRQ*	Input: refresh request
3	EXP3RQ*	Input: expansion board 3 request
4	DKRQ*	Input: disk bus request
5	EXP2RQ*	Input: expansion board 2 request
6	EXP1RQ*	Input: expansion board 1 request
7	EXP0RQ*	Input: expansion board 0 bus request
8	BGACK*	Input: bus grant acknowledge
9	ENRAS*	Input: enable row address strobe
10	(Ground)	------
11	OE	Output enable
12	NRF/BGC*	Output: nonrefresh/bus grant common
13	BGC*	Output: bus grant common (any bus master except 68010)
14	EXP0BG*	Output: expansion board 0 bus grant
15	EXP1BG*	Output: expansion board 1 bus grant
16	EXP2BG*	Output: expansion board 2 bus grant
17	DKBG*	Output: disk drive bus grant
18	EXP3BG*	Output: expansion board 3 bus grant
19	RFBG*	Output: refresh bus grant
20	VCO	Voltage-controlled oscillator: +5v

NOTATION

XX	Boolean AND
*	Negation or active low
:=	Register output latched clock
+	Boolean OR

EQUATIONS FOR BUS GRANT

DKBG := (DKRQ X ENRAS* X BGACK* X DKRQ*) + EXP3BG X BGACK)

```
RFBG   := (RFRQ X ENRAS* X BGACK* X DKRQ* X EXP3RQ*)
          + (RFBG X BGACK)

EXP2BG := (EXP2RQ X ENRAS* X BGACK* X DKRQ* X EXP3RQ* X RFRQ*)
           + (EXP2BG X BGACK)

EXP1BG := (EXP1RQ X ENRAS* X BGACK* X DKRQ*  X EXP3RQ* X RFRQ*
          X EXP2RQ*) + (EXP1BG X BGACK)

EXP0BG := (EXP0RQ X ENRAS* X BGACK* X DKRQ* X EXP3RQ* X RFRQ* X
          EXP2RQ* X EXP1RQ*) + (EXP0BG X BGACK)
```

The six preceding logic equations describe the input conditions that produce a DMA bus grant in response to a request from any one of the six devices that requests the bus. The devices are named by the output expression. The equations are listed in order of priority: DKRQ (disk request) is the highest priority; EXP0 (expansion board 0) is the lowest. The 68010 does not appear in these equations, because it receives the bus only when no DMA device is requesting the bus. To the right of the equals sign in each logic equation are four terms in the following form:

```
[bus grant] = [bus request] X [bus not in use] X [priority term]
          + [latch term]
```

The first term is the bus request (RFRQ, EXP3RQ, DKRQ, etc.). This term is ANDed with ENRAS* and BGACK*, which form the "bus not in use" term. The result is that, if a bus cycle is in process when a request is made, the bus grant is not issued until the current cycle is completed.

The third term is the priority term. In the equation for EXP3BG, the priority term is DKRQ*, because disk request has a higher priority than expansion board 3. Thus the bus is not granted to expansion board 3 if a refresh request is pending. Each equation has a priority term that names all the devices of higher priority. The first equation for DKBG has no priority term, because there is no device of higher priority. The last equation for expansion board 0 has a priority term that names all devices that may request the bus except itself, since it is the lowest priority. Thus the bus is not granted for a given line if any one of those above is pending.

The fourth term is the latch term, which contains the output of the expression ANDed with bus grant acknowledge. When a bus request is granted, bus grant acknowledge keeps the bus grant active after the bus request is removed. The grant is removed when bus grant acknowledge is removed.

Equation for Bus Grant Common

```
BGC := (RFRQ + EXP3DRQ + DKRQ + EXP2RQ + EXP1RQ + EXP0RQ) X
       (ENRAS* X BGACK*) + BGC X BGACK

[bus grant common] = [any DMA request] X [bus not in use]
                   + [latch]
```

This equation has the four parts shown above. The first part of this equation contains all the possible DMA devices that could request the bus ANDed with the "bus not in use" term, enable row address strobe, and bus grant acknowledge. Thus, when any DMA device requests the bus, BGC is asserted as soon as the cycle being executed is completed. Completion of the cycle is indicated when ENRAS* and BGACK* are not asserted.

The third term is a latch term consisting of the function output, BGC, ANDed with BGACK. Once the BGC is asserted, it remains asserted as long as the bus grant acknowledge is asserted, and it is deasserted when bus grant acknowledge is deasserted.

Equation for Nonrefresh Bus Grant Common

```
NRFBGC := (EXP3RQ + DKRQ + EXP2RQ + EXP1RQ + EXP0RQ) X (ENRAS* X
          BGACK* X RFQR*) + (NRFBGC X BGACK)
```

This nonrefresh bus grant is similar to bus grant common. It becomes active at the end of the current cycle when any DMA device, except refresh, requests the bus.

The last term is again a latch term to keep the bus grant active while the acknowledge is active after the original request becomes inactive.

Disk Interface: PAL 16R4A (Sheet 2: D-6)

Pin Number	Mnemonic	Description
1	1PCK+	Input: processor clock
2	RD*	Input: read--indicates 68010 or DMA device is reading memory or an I/O device
3	WR*	Input: write--indicates 68010 or DMA device is writing to memory or an I/O device
4	DMAR/W*	Input: direct memory access read/write--high indicates RAM memory is being read; low indicates RAM memory is being written
5	FDDRQL	Input: floppy disk drive data request--latched data from floppy disk controller pin 38 DRQ (data request); indicates that the floppy disk controller data register contains assembled data when a sector of the disk is being read or that it is empty during a sector write operation
6	HDBCS*	Input: hard disk buffer chip select--connected to pin 1 of hard disk controller; active low used to enable reading or writing data from the bus interface when controller data buffer is ready to transfer a byte
7	FDCS*	Input: floppy disk chip select--asserted when the 68010 is writing or reading the floppy disk controller data register
8	HDRE*	Input: hard disk read--asserted when the hard disk controller is reading a byte from the bus interface unit during a Sector Write command or when the 68010 is reading data from the disk controller data register; connected to the hard disk controller pin 6

Pin Number	Mnemonic	Description
9	HDWE*	Input: hard disk write--asserted when the hard disk controller is writing a byte to the bus interface unit during a Sector Write command or when the 68010 is writing data from the disk controller data register; connected to the hard disk controller pin 7
10	GND	Ground
11	OE	Output enable: connected low through a 330-ohm resistor to ground
12	FDRE*	Output: floppy disk read--connected to the RE pin 4 of the floppy disk controller
13	FDWE*	Output: floppy disk write--connected to the WE pin 2 of the floppy disk controller
14	FDTFER*	Output: floppy disk transfer--not connected outside of PAL used by rest of PAL logic to produce floppy disk read and write on pins 12 and 13 described above
15	QB*	Output: latch B output--not connected outside PAL chip used internally to latch status of FDTFER
16	QC*	Output: latched value of QB--not connected outside of PAL
17	QD*	Output: latched value of QC--not connected outside of PAL
18	TFER*	Output: transfer request--goes to disk DMA bus interface unit; initiates transfer of data from disk controller to bus interface unit and generates a disk bus request
19		Output: not used
20	VCC	+5 volts

LOGIC EQUATIONS

```
FDTFER    := (FDDRQL X FDTFER*) + (FDTFER X QD*)

QB        := FDTFER

QC        := QB

QD        := QC

FDRE      := (FDCS X RD) + (FDTFER X DMARW)

FDWE      := (FDCS X WR) + (FDTFER X DMARW*)

TFER      := (QD X DMARW*) + (QD* X DMARW) + QB + (HBCS X HDRE)
               + (HBCS X HDWE)
```

The transfer request (TFER) is asserted when the hard disk controller or the floppy disk controller is ready to transfer a byte to or from its data register to or from the disk bus interface unit buffers. On every other assertion of TFER, a disk DMA bus request is issued by the interface unit.

MMU and Bus Error: PAL 16L8A (Sheet 2: C-6)

Pin Number	Mnemonic	Description
1	PA22	Input
2	KADDR	Input: kernal access address, A19 A20, and A21 all low
3	SUP+	Input: 68010 in supervisor mode
4	BGC*	Input: bus grant common
5	LPS0*	Input: latched page status bit 0
6	LPS1*	Input: latched page status bit 1
7	R/W*	Input: high = read; low = write
8	LWE+	Input: latched page status bit write enable
9	SPA23	Input: highest address bit
10	GND	Ground
11	T90	Input
12	CASDIS*	Output
13	IODTACK*	Input
14	PAS*	Input
15	BERREN	Output: bus error enable
16	PGF*	Output: page fault
17	MMUERR*	Output: memory management unit error
18	UIE*	Output: user access to address outside memory
19	PPS0	Output: updated processor page status bit 0
20	VCC	+5 volts

Memory Management Unit Error Equation

```
MMUERR    =    BGC* X PA22* X SPA23* X LPS0* X LPS1*
               + BGC X LPS0* X LPS1*
               + BGC* X SUPV* X SPA23* X PA22* X KADDR
               + BGC* X SUPV* X  PA23* X PA22* X RW X LWE*
```

This equation is the logical ORing of four terms, each of which causes a memory management error.

In the first term, BGC*, PA22*, and SPA23* all high indicate a user access to RAM memory. LPS0* and LPS1* both high indicate that the access is to a page that has been designated not physically present.

In the second term, BGC high indicates a DMA access, and LPS0* and LPS1* indicate access is to a page not physically present.

In the third term, SUPV* indicates a user access, 68010 not in supervisory mode, and KADDR indicates the access is to the kernel.

In the fourth term, LWE* indicates the access is to a page not write enabled, and WE* indicates a write cycle.

The four conditions that generate a memory management error are summarized in the following table:

- Processor access to a page not present
- DMA access to a page not present
- User access to the kernel
- User attempt to write to a page not write enabled

MMUERR is used on sheet 6 to generate a level 7 interrupt.

Page Fault Equation

```
PGF  =    BGC* X PA22 X SPA23* X LPSO* X LPS1*
          + (BGC X LPS0* X LPS1*)
```

Page fault is the ORing of two terms. In the first, BGC* indicates a 68010 access, and LPS0* and LPS1 indicate access is to a page not present. In the second, BGC indicates a DMA access. Thus page fault occurs when either the 68010 or a DMA device tries to access a page not present.

User Nonmemory Location Error Equation

```
UIE = (BGC* X SUPV* X ENRAS * X PA22) + (BGC* X SUPV* X ENRAS*)
```

In this equation, SUPV* indicates a user access; PA22 and ENRAS* indicate that the access is to a nonmemory location.

Bus Error Enable Equation

```
BERREN=   BGC* X PA22* X SPA23* X LPS0* X LPS1* X T90 X PAS
        + BGC* X SUPV* X SPA23* X PA22* X KADDR X T90 X PAS
        + BGC* X SUPV* X SPA23* X PA22* X RW X LWE* X T90
        + BGC* X SUPV* X  SPA23*X PA22 X IODTACK
        + BGC* X SUPV* X SPA23* X DTACK
```

This equation is the ORing of five terms. The condition to assert each term is:

- User access to page not present
- User access to kernel
- User access to page not write enabled
- User access to address not in memory
- User access to address not in memory

BERREN is ANDed with EE* (the error enable bit) and connected to BERR- pin 22 of the 68010.

Column Address Disable Strobe Equation

```
CASDIS =  BGC* X PA22* X SPA23* X LPS0* X LPS1* X RW
        + BGC X LPS0* X LPS1* X RW
        + BGC* X SUPV* X SPA23* X PA22* X KADRR X RW
        + BGC* X SUPV* X SPA23* X PA22* X RW X LWE* X RW
```

The column disable bit prevents writing to RAM memory during a memory management unit error. Thus the terms of these equations contain the same conditions that generate the MMUERR ANDed with the RW write bit.

Page Status Bit 0 Equation

```
PPS0 =    RW* X LPS1* X LPS0 X MMUERR*
        + RW* X LPS1 X LPS0* X MMUERR*
        + LPS0* X MMUERR
```

This equation lists the following three conditions for setting bit 0 of page status to 1:

- Writing to a page that is present but has never been written to, when there is no MMUERR

- Writing to a page that is present and has been read but not previously written to, when there is no MMUERR
- A MMUERR exists and PSO was previously written to, when there is no MMUERR
- A MMUERR exists and PSO was previously 0.

Hard Disk Data Separator: PAL 16R4 (Sheet 26)

Pin Number	Mnemonic	Description
1	MUXCLK*	Input clock for multiplex clock output
2	PLLCLK	Output of VCO
3	DATA0	Data and clock from drive 0
4	DATA1	Grounded
5	DDRIVE0*	Grounded
6	HDRGATE	High when disk controller is inspecting data
7	PCLK*	10 MHz
8	TEST	Grounded
9	REF	Grounded
10	GND	
11	OE*	Grounded
12	RCLK	1/2 PLL
13	DCLK	No external connection
14	FFA*	No external connection
15	OSCENB*	Enables VCO
16	FFB*	No external connection
17	FFC*	No external connection
18	CLR*	Clears pullup/pulldown flipflops
19	MUX*	Multiplex clock output
20	VCC	

Equations

```
FFA     :  =    HDRGATE

FFB     := FFA

FFC     := FFB

OSCENB := HDRGATE* FFC* + HDDRGATE X FFC

CLR      = HDRGATE* X FFC + HDRGATE X FFC* + OSCENB*

DCLK*    = PLLCLK X RCLK + PLLCLK* X DCLK* + RCLK X DCLK* + TEST
```

RCLK* = PLLCLK* X DCLK* + PLLCLK X RCLK* + DCLK* X RCLK*

MUX = HDRGATE X DATA0* X DDRIVE0 + HDRGATE X DATA1*
X DDRIVE0* + HDRGATE* X PCLK*

If HDRGATE is high, MUX pin 19 outputs drive 0 data coming in on pin 3. If HDRGATE is low, MUX outputs PCK* coming in on pin 7.

When HDRGATE is high, PLLK is phase-locked to DATA0 at pin 3 at twice the frequency. DATA0 is 5 MHz and PLL is 10 MHz.

When HDRGATE is low, PLLK is 10 MHz, phase locked to PCK* at pin 7.

RCLK is always half the frequency of PLLCLK.