PRELIMINARY

CY82C693U

hyperCache™ PCI Peripheral Controller with USB

Features

- PCI to ISA bridge
- PCI Bus Rev. 2.1 compliant
- Supports up to 5 additional PCI masters including the CY82C691
- Integrated DMA controllers with Type A, B, and F support
- Integrated Interrupt controllers
- Integrated timer/counters
- Integrated Real-Time-Clock with 256 bytes of battery-backed SRAM (14 bytes of clock RAM and 242 bytes of CMOS scratch RAM)
- Write-only Register Shadowing
- Integrated Dual-Channel enhanced IDE controller with
  - PCI bus mastering
  - CD ROM support
  - PIO modes 0 through 4 operation
  - Single-word and Multi-word DMA modes 0 through 2
- Integrated Keyboard Controller
- APM compliant power management support through SMM or under hardware control
- Flash PROM support with Write-protection
- Power-on reset circuitry
- QuietBus™ support for the PCI and ISA bus interfaces for better noise immunity
- General-purpose I/O pins and registers
- Provides PCI-ISA/ISA-PCI/IDE-PCI/PCI-IDE post writing
- Provides ISA-PCI pre-reading
- USB Host/Hub controller with 2 USB ports
- Flexible Stand-Alone configuration options
- Packaged in a 208-pin PQFP

System Block Diagram (hC–VX, hC–DX solutions)

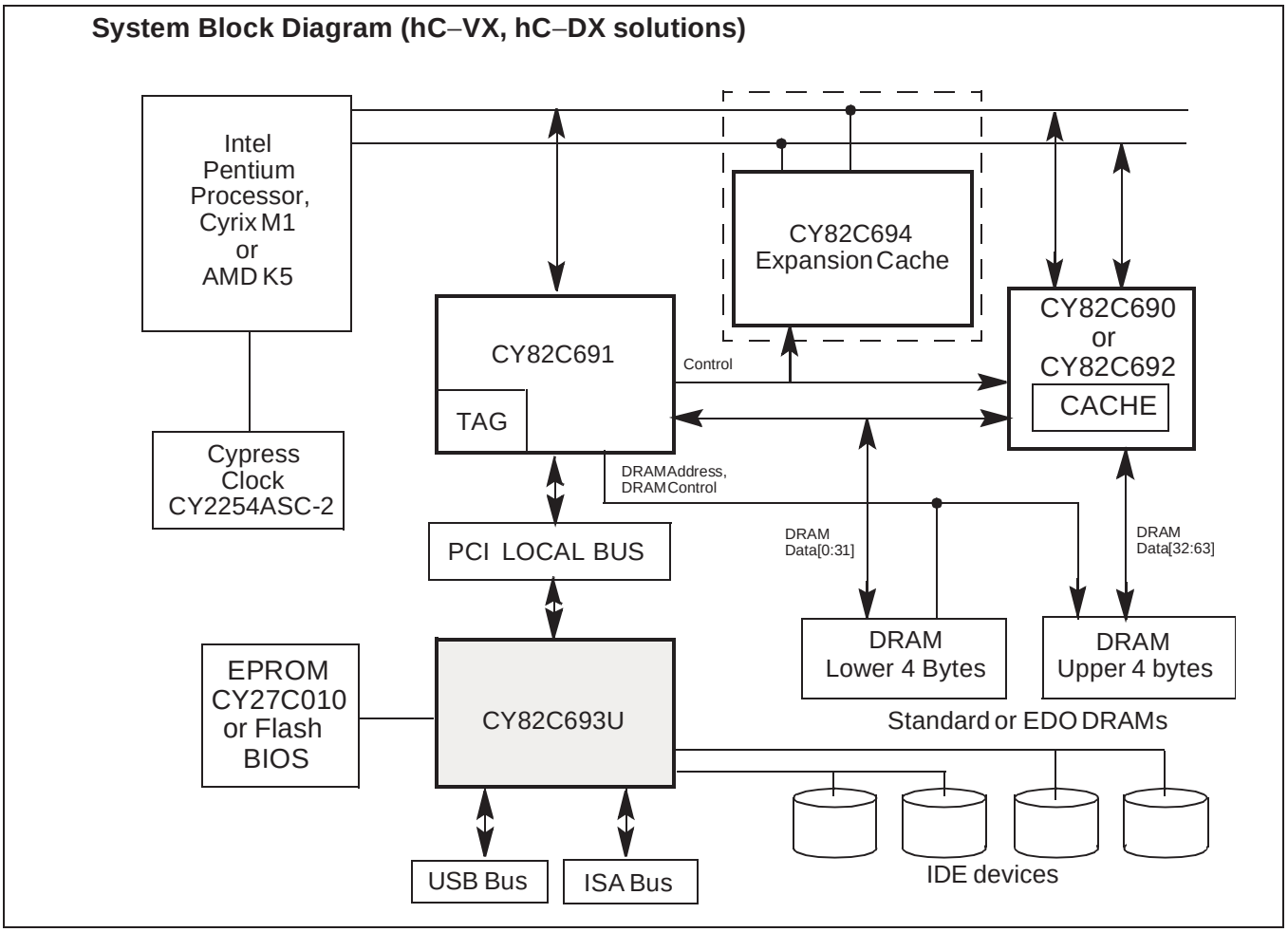


TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

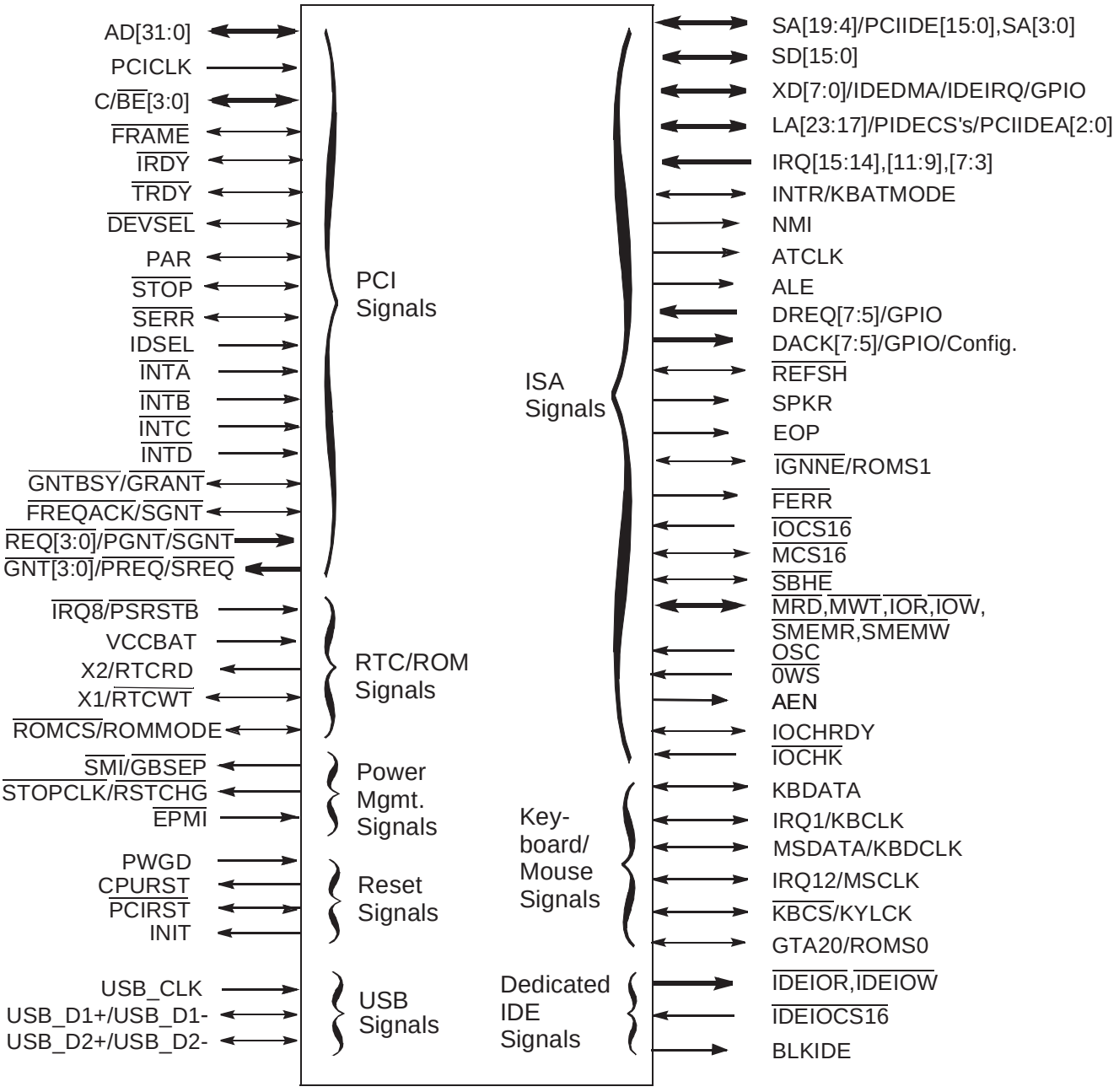
TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

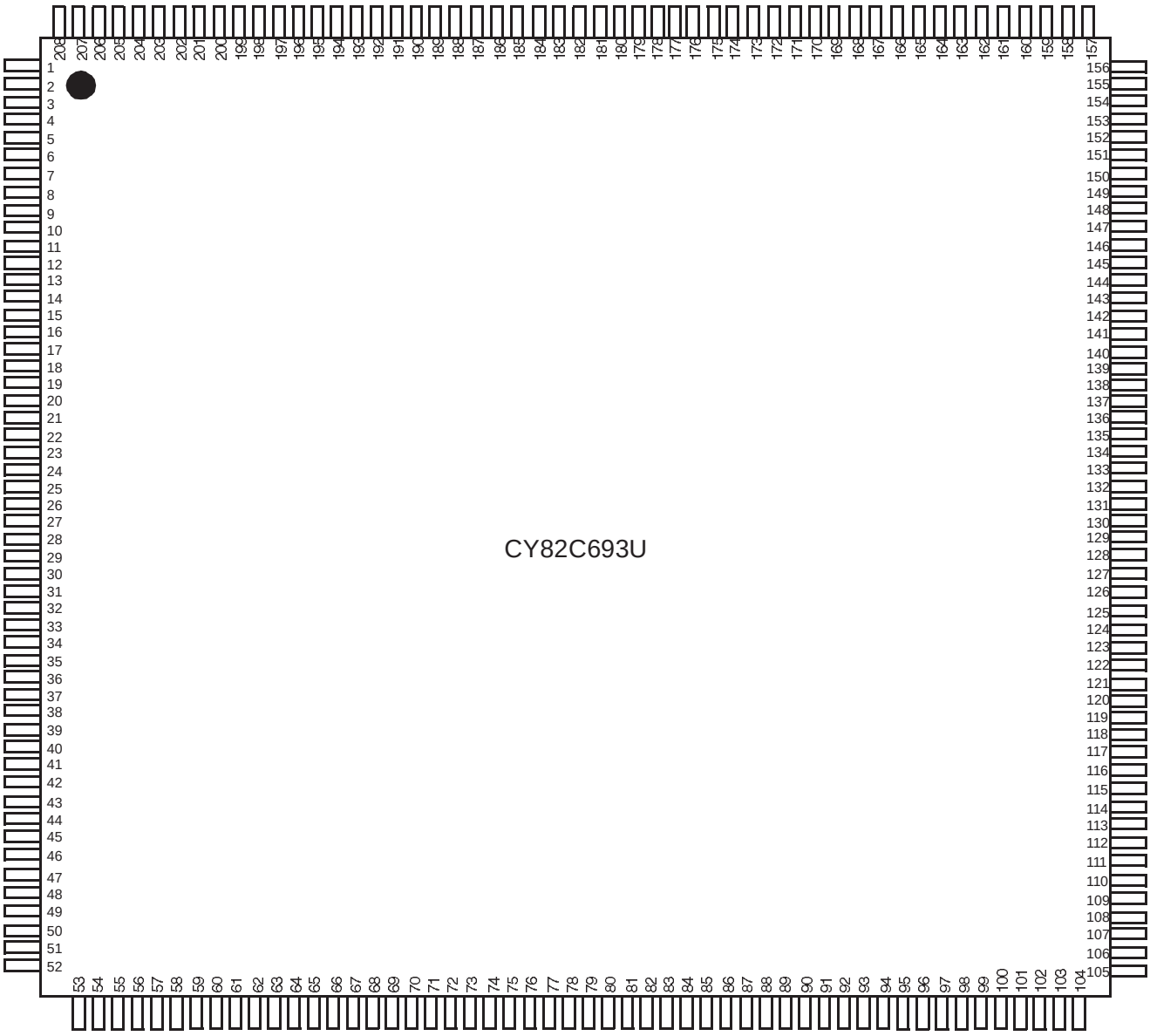
CY82C693U Signals



82C693U–2

Pin Configuration

208-pin PQFP
Top View



82C693U–2

CY82C693U Pin Reference (In Numerical Order by Pin Number)

Pin #	Pin Name	Pin #	Pin Name	Pin #	Pin Name	Pin #	Pin Name	Pin #	Pin Name
1	GND	43	AD9	85	SA7/IDE3	127	X1/RTCWT	169	+3.3V
2	FRAME	44	AD8	86	SA6/IDE2	128	PWGD	170	GTA20/ROMS0
3	IDSEL	45	AD7	87	SA5/IDE1	129	DRQ0	171	GND
4	TRDY	46	AD6	88	SA4/IDE0	130	DRQ5/GPIO6/ PWREN	172	STOPCLK/ RSTCHG
5	IRDY	47	AD5	89	SA3	131	DRQ6/GPIO7/ OVRCUR	173	CPURST
6	DEVSEL	48	AD4	90	SA2	132	DRQ7/GPIO8/ SMIACT	174	INIT
7	SERR	49	AD3	91	SA1	133	OSC	175	NMI
8	PAR	50	AD2	92	SA0	134	SD0	176	INTR/ KBATMODE
9	USB_D1+	51	AD1	93	GND	135	SD1	177	IGNNE/ROMS1
10	USB_D1–	52	AD0	94	ROMCS/ ROMMODE	136	+5V	178	FERR
11	C/BE3	53	GND	95	IRQ7	137	GND	179	SMI/GBSEP
12	C/BE2	54	IOCHK	96	IRQ6	138	SD2	180	USB_D2+
13	C/BE1	55	IOCHRDY	97	IRQ5	139	SD3	181	USB_D2–
14	C/BE0	56	0WS	98	IRQ4	140	SD4	182	EPMI
15	USB_CLK	57	AEN	99	IRQ3	141	SD5	183	MSDATA/ KBDCLK
16	AD31	58	SMEMW	100	IRQ1/KBCLK	142	SD6	184	KBCS/KYLCK
17	AD30	59	SMEMR	101	REFSH	143	SD7	185	KBDATA
18	AD29	60	IOW	102	IRQ9	144	MRD	186	IDEIOCS16
19	AD28	61	IOR	103	ALE	145	MWT	187	IDEIOW
20	+5V	62	LA23/IDECS0	104	+5V	146	SD15	188	IDEIOR
21	GND	63	LA22/IDECS1	105	GND	147	SD14	189	GNTBSY/ GRANT
22	AD27	64	LA21/SIDECS0	106	IRQ10	148	SD13	190	FREQACK/IRQ8
23	AD26	65	+5V	107	ATCLK	149	SD12	191	GNT0/PREQ
24	AD25	66	LA20/SIDECS1	108	IRQ11	150	SD11	192	GNT1/SREQ
25	AD24	67	GND	109	IRQ12/MSCLK	151	GND	193	GNT2/SQWV
26	AD23	68	LA19/IDEA2	110	IRQ14	152	SD10	194	GNT3/DISARB
27	AD22	69	LA18/IDEA1	111	IRQ15	153	SD9	195	+5V
28	AD21	70	LA17/IDEA0	112	DACK0/TSTM	154	SD8	196	REQ0/PGNT
29	AD20	71	GND	113	DACK1/TSTM0	155	XD0/XDIR	197	GND
30	AD19	72	SA19/IDE15	114	DACK2/TSTM1	156	+5V	198	REQ1/SGNT
31	AD18	73	SA18/IDE14	115	DACK3/ DISPSEL	157	BLKIDE	199	REQ2
32	+5V	74	SA17/IDE13	116	DACK5/KB-SEL/GPIO9	158	XD1/XDEN	200	REQ3
33	GND	75	SA16/IDE12	117	DACK6/RTC-SEL/GPIO10	159	XD2/IACK1/GPIO5	201	PCIRST
34	AD17	76	SA15/IDE11	118	DACK7/EXT-BUF/GPIO11	160	XD3/IREQ1/GPIO4	202	INTD
35	AD16	77	SA14/IDE10	119	EOP	161	XD4/IACK0/GPIO3	203	INTC

CY82C693U Pin Reference (In Numerical Order by Pin Number) (continued)

36	AD15	78	SA13/IDE9	120	DRQ1	162	XD5/IREQ0/GPIO2	204	$\overline{\text{INTB}}$
37	AD14	79	SA12/IDE8	121	DRQ3	163	XD6/IDEIRQ1/ GPIO1/$\overline{\text{BUSY}}$	205	$\overline{\text{INTA}}$
38	AD13	80	SA11/IDE7	122	DRQ2	164	XD7/IDEIRQ0/ GPIO0	206	$\overline{\text{STOP}}$
39	AD12	81	GND	123	$\overline{\text{IRQ8}}$/PSRSTB	165	$\overline{\text{SBHE}}$	207	PCICLK
40	AD11	82	SA10/IDE6	124	VCCBAT	166	$\overline{\text{MCS16}}$	208	+5V
41	AD10	83	SA9/IDE5	125	GND	167	$\overline{\text{IOCS16}}$		
42	GND	84	SA8/IDE4	126	X2/$\overline{\text{RTCRD}}$	168	SPKR		

CY82C693U Pin Reference (In Alphabetical Order by Signal Name)

Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.
AD0	52	C/$\overline{\text{BE3}}$	11	INTR/ KBATMODE	176			SD8	154
AD1	51	CPURST	173	$\overline{\text{IOCHK}}$	54			SD9	153
AD2	50	$\overline{\text{DACK0}}$/TSTM	114	IOCHRDY	55	PWGD	128	SD10	152
AD3	49	$\overline{\text{DACK1}}$/TSTM0	114	$\overline{\text{IOCS16}}$	167	$\overline{\text{REFSH}}$	101	SD11	150
AD4	48	$\overline{\text{DACK2}}$/TSTM1	114	$\overline{\text{IOR}}$	61	$\overline{\text{REQ0}}$/ $\overline{\text{PGNT}}$	196	SD12	149
AD5	47	$\overline{\text{DACK3}}$ / DISPSEL	115	$\overline{\text{IOW}}$	60	$\overline{\text{REQ1}}$/ $\overline{\text{SGNT}}$	198	SD13	148
AD6	46	$\overline{\text{DACK5}}$/ KBSEL/GPIO9	116	$\overline{\text{IRDY}}$	5	$\overline{\text{REQ2}}$	199	SD14	147
AD7	45	$\overline{\text{DACK6}}$/RTC-SEL/GPIO10	117	IRQ1/KBCLK	100	$\overline{\text{REQ3}}$	200	SD15	146
AD8	44	$\overline{\text{DACK7}}$/EXT-BUF/GPIO11	118	IRQ3	99	$\overline{\text{ROMCS}}$/ ROMMODE	94	$\overline{\text{SERR}}$	7
AD9	43	$\overline{\text{DEVSEL}}$	6	IRQ4	98			$\overline{\text{SMEMR}}$	59
AD10	41	DRQ0	129	IRQ5	97	SA0	92	$\overline{\text{SMEMW}}$	58
AD11	40	DRQ1	120	IRQ6	96	SA1	91	$\overline{\text{SMI}}$/GBSEP	179
AD12	39	DRQ2	122	IRQ7	95	SA2	90	SPKR	168
AD13	38	DRQ3	121	$\overline{\text{IRQ8}}$/PSRSTB	123	SA3	89	$\overline{\text{STOP}}$	206
AD14	37	DRQ5/GPIO6/ PWREN	130	IRQ9	102	SA4/IDE0	88	$\overline{\text{STOPCLK}}$/ $\overline{\text{RSTCHG}}$	172
AD15	36	DRQ6/GPIO7/ OVRCUR	131	IRQ10	106	SA5/IDE1	87	$\overline{\text{TRDY}}$	4
AD16	35	DRQ7/GPIO8/ SMIACT	132	IRQ11	108	SA6/IDE2	86	USB_CLK	15
AD17	34	EOP	119	IRQ12/MSCLK	109	SA7/IDE3	85	$\overline{\text{USB\_D1+}}$	9
AD18	31	$\overline{\text{EPMI}}$	182	IRQ14	110	SA8/IDE4	84	USB_D1–	10
AD19	30	$\overline{\text{FERR}}$	178	IRQ15	111	SA9/IDE5	83	USB_D2+	180
AD20	29	$\overline{\text{FRAME}}$	2	$\overline{\text{KBCS}}$/KYLCK	184	SA10/IDE6	82	USB_D2–	181
AD21	28	$\overline{\text{FREQACK}}$/ $\overline{\text{IRQ8}}$	190	KBDATA	185	SA11/IDE7	80	VCCBAT	124

CY82C693U Pin Reference (In Alphabetical Order by Signal Name) (continued)

Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.
AD22	27	GND	1, 21, 33, 42, 53, 67, 71, 81, 93, 105, 125, 137, 151, 171,197	LA17/IDEA0	70	SA12/IDE8	79	X1/$\overline{\text{RTCWT}}$	127
AD23	26	$\overline{\text{GNT0}}$/$\overline{\text{PREQ}}$	191	LA18/IDEA1	69	SA13/IDE9	78	X2/RTCRD	126
AD24	25	$\overline{\text{GNT1}}$/$\overline{\text{SREQ}}$	192	LA19/IDEA2	68	SA14/IDE10	77	XD0/XDIR	155
AD25	24	$\overline{\text{GNT2}}$/$\overline{\text{SQWV}}$	193	LA20/SIDECS1	66	SA15/IDE11	76	XD1/$\overline{\text{XDEN}}$	158
AD26	23	$\overline{\text{GNT3}}$/$\overline{\text{DISARB}}$	194	LA21/SIDECS0	64	SA16/IDE12	75	XD2/IACK1/GPIO5	159
AD27	22	$\overline{\text{GNTBSY}}$/ $\overline{\text{GRANT}}$	189	LA22/IDECS1	63	SA17/IDE13	74	XD3/IREQ1/GPIO4	160
AD28	19	GTA20/ROMS0	170	LA23/IDECS0	62	SA18/IDE14	73	XD4/IACK0/GPIO3	161
AD29	18	$\overline{\text{IDEIOCS16}}$	186			SA19/ IDE15	72	XD5/IREQ0/GPIO2	162
AD30	17	$\overline{\text{IDEIOR}}$	188	$\overline{\text{MCS16}}$	166	$\overline{\text{SBHE}}$	165	XD6/IDEIRQ1/ GPIO1/$\overline{\text{BUSY}}$	163
AD31	16	$\overline{\text{IDEIOW}}$	187	$\overline{\text{MRD}}$	144	SD0	134	XD7/IDEIRQ0/ GPIO0	164
AEN	57	IDSEL	3	MSDATA/ KBDCLK	183	SD1	135	$\overline{\text{0WS}}$	56
ALE	103	$\overline{\text{IGNNE}}$/ROMS1	177	$\overline{\text{MWT}}$	145	SD2	138	+3.3V	169
ATCLK	107	INIT	174	NMI	175	SD3	139	+5V	20,32, 65,104, 136,156, 195,208
BLKIDE	157	$\overline{\text{INTA}}$	205	OSC	133	SD4	140		
C/$\overline{\text{BE0}}$	14	$\overline{\text{INTB}}$	204	PAR	8	SD5	141		
C/$\overline{\text{BE1}}$	13	$\overline{\text{INTC}}$	203	PCICLK	207	SD6	142		
C/$\overline{\text{BE2}}$	12	$\overline{\text{INTD}}$	202	$\overline{\text{PCIRST}}$	201	SD7	143		

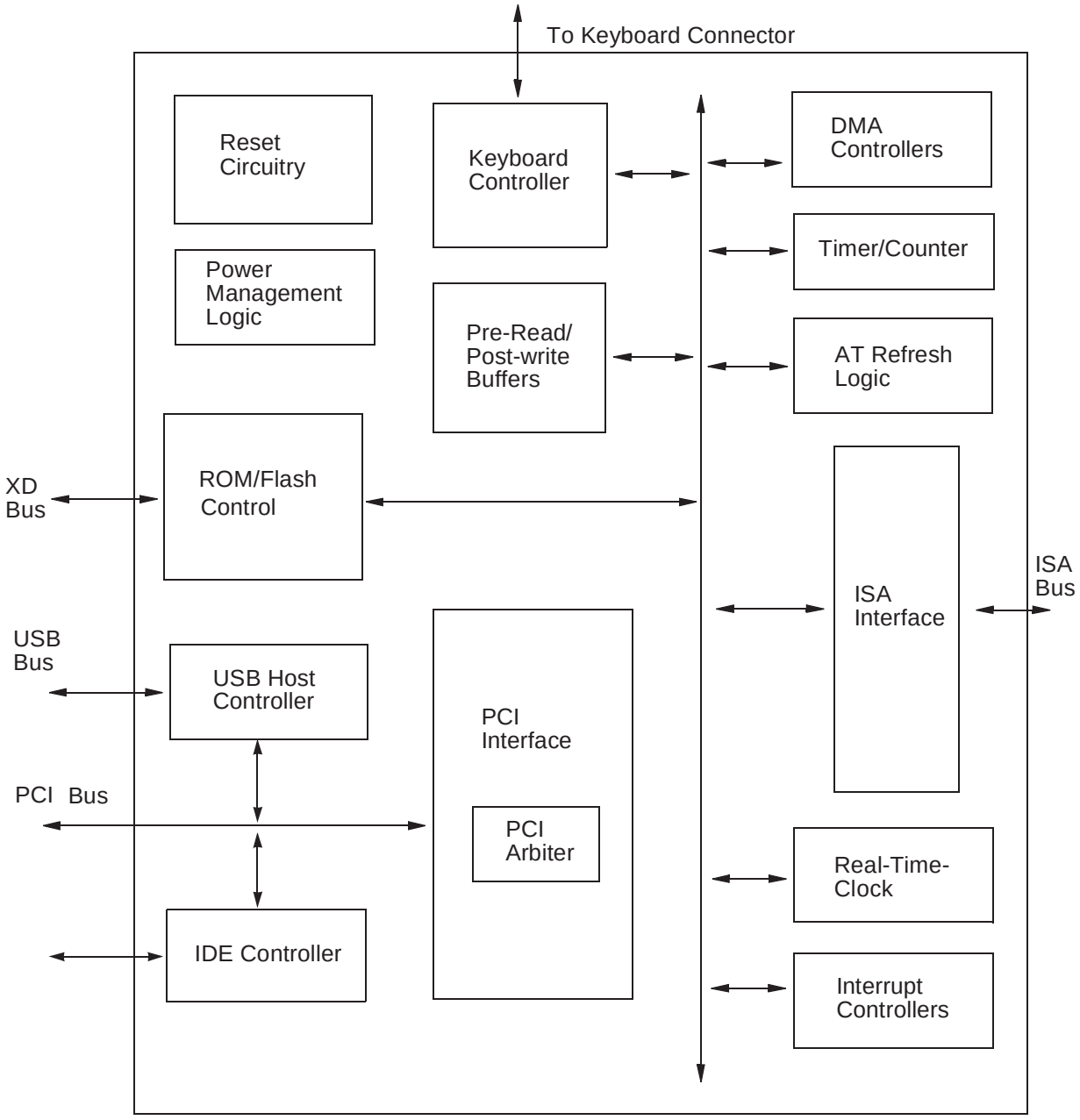


Figure 1. CY82C693U Functional Block Diagram.

Introduction

System Overview

The hyperCache™ family is a family of three chipsets created to provide flexible solutions for today's PC designs. The hC-ZX, hC-VX, hC-DX Chipsets provide all the functions necessary to implement a 3.3V Pentium-class processor based system with the USB (Universal Serial Bus), PCI (Peripheral Component Interconnect), and the ISA (Industry Standard Architecture) buses. System designers can exploit the advantages of the USB and PCI buses while maintaining access to the large base of ISA cards in the marketplace.

The Cypress hyperCache family offers system designers several key advantages. With only three chips, a complete system can be implemented. Cache can be added up to 512 MB with additional CY82C694 devices in 128-KB increments All chipset solutions are pin-compatible and provide flexible upgrade paths through on-board or external cache modules. Six banks of page-mode or EDO DRAM further increase the system designer's options. The chipset also contains concurrent bus support, PCI enhanced IDE with CD-ROM support, integrated RTC, integrated peripheral control (Interrupts/ DMA), and integrated keyboard controller. This chipset is flexible enough to provide the system designer with many cost/performance/function options to provide an optimum solution for a given design.

CY82C693U Introduction

The CY82C693U Peripheral Controller provides a highly integrated peripheral solution for PCI-based motherboards. The CY82C693U contains a PCI to ISA bridge, a PCI IDE controller, DMA controllers, Interrupt controllers, a Real-Time-Clock, Keyboard controller and a USB Host Controller.

Figure 1 shows a block diagram of the CY82C693U.

Functional Overview

The CY82C693U Peripheral Controller contains the following functional blocks:

- PCI Interface
- ISA Interface
- Reset Logic
- Keyboard Controller
- Power Management Logic
- AT Refresh Logic
- Pre-Read/Post-Write Buffers
- BIOS ROM Control.
- Timer/Counter Logic
- DMA Controllers
- Dual-Channel Enhanced IDE Controller
- Real-Time-Clock with 32-kHz Oscillator
- Interrupt Controllers
- USB Host Controller

PCI Bus Interface

The CY82C693U provides a bridge for transactions between the PCI bus, the ISA bus, and IDE peripherals. PCI bus speeds of 25, 30, or 33 MHz are supported. The PCI interface is master/slave (it can initiate or be a target for transactions). The PCI bus interface in the CY82C693U is Revision 2.1 compliant. This standard allows for a multitude of high-speed peripheral cards to be added to the system. PCI is the predominant local bus for Pentium systems.

Master cycles are initiated by driving $\overline{\text{FRAME}}$ LOW with a valid address on AD[31:0], valid even address parity on PAR, and a valid command on C/$\overline{\text{BE}}$[3:0]. Data phases occur when $\overline{\text{IRDY}}$ (initiator ready) and $\overline{\text{TRDY}}$ (target ready) are both active, valid data is placed on AD[31:0], the PAR signal is driven to reflect even parity, and the correct byte enable combination is present on C/$\overline{\text{BE}}$[3:0]. Wait states can be inserted into a transaction if the initiator deasserts $\overline{\text{IRDY}}$ or the target deasserts $\overline{\text{TRDY}}$. A transaction is terminated by the deassertion of $\overline{\text{FRAME}}$ prior to the final data phase. As a PCI master, the CY82C693U will only perform memory read and write transactions. A write cycle consists of a maximum of 4 bytes of data in a single data cycle. A read consists of a maximum of 8 bytes in a two cycle burst.

The CY82C693U uses a subtractive decode strategy to determine if a PCI target transaction is destined for the ISA bus. Potential PCI targets decode any valid address during the first cycle of the transaction ($\overline{\text{FRAME}}$ asserted). If a PCI peripheral device (including the integrated IDE controller in the CY82C693U) detects a transaction to its address space, it will assert $\overline{\text{DEVSEL}}$. If the CY82C693U does not detect the assertion of $\overline{\text{DEVSEL}}$ by another target (or its own IDE controller) within 4 PCI clock cycles, it will claim the transaction by asserting $\overline{\text{DEVSEL}}$. All transactions that are not claimed by a PCI peripheral are, by default, sent to the ISA bus. After asserting $\overline{\text{DEVSEL}}$, the CY82C693U will initiate an ISA cycle and assert $\overline{\text{TRDY}}$ to the PCI bus when valid data is available.

As a PCI slave, the CY82C693U will target-terminate the cycle after the first data transfer by asserting $\overline{\text{STOP}}$ with $\overline{\text{TRDY}}$ if $\overline{\text{FRAME}}$ is not deasserted before the data phase. The CY82C693U does not accept bursts as a target. By not allowing target bursts, PCI bus bandwidth, which would otherwise be quickly consumed by ISA or IDE targets, is conserved.

The CY82C693U supports PCI error reporting through the $\overline{\text{SERR}}$ (system error) signal. Both internally detected and externally generated errors are reported. $\overline{\text{SERR}}$ is asserted for any system error, including address/data parity errors on Special Cycle commands, except for data parity. If the CY82C693U detects the assertion of $\overline{\text{SERR}}$, it will assert NMI (non-maskable interrupt) to the CPU. The CY82C693U will store the source of the NMI ($\overline{\text{SERR}}$) in an internal configuration register to allow the NMI handler software to determine the cause of the error.

The CY82C693U supports the four PCI interrupt signals ($\overline{\text{INTA}}$, $\overline{\text{INTB}}$, $\overline{\text{INTC}}$, and $\overline{\text{INTD}}$). The interrupt lines are open-drain and should be routed to all of the PCI slots. For single-function devices, only $\overline{\text{INTA}}$ should be used. The three other interrupt lines can be connected to any set of functions on a multi-function device. Each interrupt signal can be programmed to be level-sensitive (PCI Compliant) or edge-triggered (not PCI Compliant). The assertion of a PCI interrupt request will cause the INTR signal to be asserted to the processor. When the processor performs an interrupt acknowledge cycle, the CY82C693U will return an interrupt vector based on the level of the PCI interrupt. The PCI interrupt levels are programmable. Interrupt programmability is useful in resolving system conflicts.

The CY82C693U contains the PCI arbiter. The CY82C693U supports up to five PCI masters, including the CY82C691.

There are four dedicated request and grant line pairs, one for each slot, and a special busy and grant for the CY82C691. The CY82C691 is the default owner of the PCI bus. If the CY82C691 requires ownership of PCI and is granted the bus, it will assert the GNTBSY signal. When the CY82C693U sees another master's request asserted, and no higher priority masters are requesting the bus, the CY82C693U will remove the grant from the CY82C691 and give the grant to the new master. The CY82C693U supports rotating priority. In rotating priority, the last master to be granted the bus becomes lowest priority. The CY82C691 is always given highest priority to reduce CPU latency due to arbitration.

ISA Bus Interface

The CY82C693U contains an ISA (Industry Standard Architecture) Bus interface. If no other target in the system claims the transaction, it is passed to the ISA bus. The ISA bus interface in the CY82C693U is full master/slave, allowing a myriad of low-cost peripheral cards to be added to the system.

ISA master cycles are performed by the CY82C693U whenever a PCI master wants to access an ISA peripheral card. The CY82C693U drives the address onto the ISA address lines and asserts BALE. The accessed resource will then respond with its size by the assertion/negation of the IOCS16 or MEMCS16 signals. The appropriate command signals (SMEMR, SMEMW, MEMR, MEMW, IOR, or IOW) will then be asserted by the CY82C693U. The deassertion of the active command signal indicates when valid data is on the bus. Wait states can be inserted into an ISA cycle by the target's negation of the IOCHRDY signal. This forces the command signal to stay on the bus until IOCHRDY is asserted again.

An ISA peripheral requests ownership of the ISA bus by asserting its DMA request signal. The request can be either a normal ISA transaction controlled by the peripheral itself, or a DMA transaction using transfer parameters (such as starting address and block count) previously set up in the DMA controller inside the CY82C693U.

In a normal ISA transaction, the ISA master begins its transfers after receiving DMA acknowledge from the CY82C693U. The CY82C693U differentiates a normal ISA bus request from an ISA DMA request by checking the transfer parameters (in the on-chip DMA controller) corresponding to the above DMA request signal. If no transfer parameters are stored, then the request is a normal ISA transfer.

If the ISA request is a DMA transfer, the CY82C693U outputs the pre-programmed starting address and the control signals onto the ISA bus. It will also assert AEN to the requesting ISA bus master so that it will ignore the address and simply provide the data to be transferred onto the ISA bus. The address on the ISA bus is the memory address used to route the DMA data to the proper memory location. If the target for a DMA/MASTER ISA cycle is on the PCI bus, the PCI bus will be arbitrated for before the CY82C693U will assert the DMA acknowledge to the ISA master.

When either an ISA master request or a DMA request is asserted to the CY82C693U, system data coherency must be maintained. This is accomplished using two sideband signals between the CY82C691 and the CY82C693U (FREQACK and GNTBSY). When the CY82C693U detects the assertion of one of the DMA request signals, it will assert FREQACK for one clock cycle. This is a request to the CY82C691 to flush all write buffers. After driving FREQACK for one cycle, the CY82C693U will drive the signal deasserted (HIGH) for one PCI clock cycle and float the signal (rescind). After the CY82C691 has flushed its write buffers (causing memory to be coherent with respect to the write buffer contents), the CY82C691 will assert FREQACK for one PCI clock cycle and rescind. This informs the CY82C693U that the write buffers have been flushed and it is free to take control of the PCI bus and perform the ISA master/DMA transfer.

GNTBSY performs the PCI arbitration function between the CY82C691 and the CY82C693U. The CY82C691 is the default owner of the PCI bus (GNTBSY is asserted LOW by the CY82C693U in the default state). When the CY82C693U wants access to the PCI bus, it drives GNTBSY HIGH for one PCI clock cycle and rescinds the signal. If the CY82C691 has a PCI bus cycle pending, it will drive GNTBSY LOW until it is finished with its transaction (PCI bus busy). Once the CY82C691 has finished its pending transaction, it will deassert GNTBSY and rescind. When the CY82C693U sees GNTBSY deasserted, it is free to grant the PCI bus to another master. The CY82C693U will leave GNTBSY deasserted as long as a master other than the CY82C691 is on the PCI bus. Once the other PCI masters finish their transactions, the CY82C693U will assert GNTBSY to inform the CY82C691 that it is free to take the PCI bus again.

Because a PCI data transfer is up to 32 bits wide on byte boundaries and ISA data can be eight or sixteen bits wide, the CY82C693U performs bus steering. Based on the address, the state of the C/BE[3:0] signals and the width of the ISA bus resident, the CY82C693U will steer data to/from the PCI data bus from/to the ISA data bus. If a PCI read transaction from the ISA bus requires data larger than the 8/16-bit ISA data width, the CY82C693U can be programmed to perform multiple ISA transactions and pack the data into a single, larger PCI data word. If less than 32 bits are required on the PCI bus (by the assertion of fewer than all 4 byte enables), only the requested data will be packed and presented on the PCI data lines. On an ISA MASTER write, data will be driven onto the PCI bus as it is written (not packed).

The ISA bus interface in the CY82C693U allows for fully synchronous/asynchronous operation. In other words, the ISA clock can be synchronized to a divided-down version of the PCI clock, or the PCI and ISA clocks can be fully independent in frequency and phase. The CY82C693U can switch the ISA clock to a fixed 7.16-MHz clock if the PCI clock is slowed to conserve system power.

The COMMAND recovery time is programmable (either 1.5 or 2.5 ISA clock cycles). Command recovery is the amount of idle ISA cycles between back-to-back ISA transactions. This allows for a trade-off between extra performance and card compatibility. The default is 2.5 ISA clock cycles. However, if the ISA peripheral cards in the system can support 1.5 ISA clocks, the CY82C693U can be programmed to reduce the overall ISA cycle time.

The CY82C693U contains QuietBus logic. Control and data signals will only be passed from PCI to ISA when the CY82C693U determines that the ISA bus is the intended target. This reduces the noise and power consumption associated with switching output buffers unnecessarily. It also eliminates some system EMI (Electro-Magnetic Interference).

Reset Logic

The CY82C693U provides three signals for resetting system components, CPURST, INIT and $\overline{\text{RESET}}$.

CPURST is an active HIGH signal that provides power-on reset functionality for the CPU. CPURST is asserted when the CY82C693U detects PWGD (power good) asserted from the power supply. CPURST forces the processor to begin execution in a known state. When the Pentium processor detects CPURST, it will immediately abort all bus activity and perform its reset sequence. The CY82C693U will assert CPURST for a minimum of 1 ms after PWGD is asserted. This is sufficient to ensure a "cold" or "power-on" reset. The assertion of CPURST will cause all internal processor registers, write-buffers, and caches to be reset. The processor will begin execution by reading from address FFFFFFF0H upon the deassertion of CPURST. CPURST is also used to reset ISA peripherals (functions as SYSRESET).

For a "warm" reset, INIT will be asserted for a minimum of 15 PCI clock cycles. A "warm" reset is performed whenever the CPU writes Port 92H, bit 0, to "1". The keyboard controller within the CY82C693U can also issue a fast "warm" reset if the user writes FE (hex) to port 64 (hex). INIT provides CPURST functionality except INIT leaves the CPU's level 1 cache, internal write-buffers, and floating-point registers intact. Only the processor core is reset. INIT can be used to switch the processor from protected to real mode. Once INIT is sampled active, the processor will begin the initialization sequence on the next instruction boundary. The initialization sequence will continue to completion followed by normal reset execution (read from address FFFFFFF0H). INIT will be asserted for a minimum of two CPU clock cycles and will remain active for three CPU clock cycles prior to the $\overline{\text{BRDY}}$ of an I/O write cycle.

$\overline{\text{RESET}}$ is used to reset all PCI peripherals (functions as $\overline{\text{PCIRST}}$). $\overline{\text{RESET}}$ (like CPURST) will be asserted for a minimum of 1ms after PWGD is asserted.

Keyboard Controller

The CY82C693U integrates the essential functions of an 8042 Keyboard Controller including:

- operating frequencies from 6 to 16 MHz
- support for a PS/2-compatible mouse
- complete operating system independence
- works with MS-DOS®, Microsoft Windows®, OS/2®, and UNIX™

Operating Frequency

The keyboard controller inside the CY82C693U operates at frequencies between 6 and 16 MHz. The clock is internally selectable.

Resetting the Keyboard Controller

The keyboard controller will be reset when the PWGD (power good) signal is negated. Once PWGD is asserted, the keyboard controller will remain in its reset state for 120 keyboard clock cycles before becoming operational.

Host Interface

PCI or ISA masters communicate with the keyboard controller by performing I/O reads and writes to two eight-bit port locations (0064H and 0060H). I/O Port 0064H is the command/status register and I/O Port 0060H is the data register. There are two host signals that the keyboard controller generates (or are functionally emulated within the CY82C693U), GTA20 and INIT. These signals initiate the A20 mask and "warm reset" respectively. There are also four keyboard general purpose I/O registers and ports that are user definable. As inputs, they set bits in internal reserved registers. As outputs, they reflect the value of the internal register bits. The general purpose I/Os are multiplexed with some alternate control signals, the XD bus, IDE DMA control, and IDE Interrupt Request signals. Therefore, these system functions will not be available from the CY82C693U when the pins are programmed as general purpose I/Os. However, not every system will require the additional system functions, and those that do can implement these system functions externally if general purpose I/O is a requirement.

PS/2 Compatible Mouse Support

The CY82C693U supports a PS/2 compatible mouse. MSCLK and MSDATA should be connected directly to the mouse connector. Mouse interrupt requests are generated internally.

Keyboard Interface

The CY82C693U provides the KBCLK and KBDATA signals to connect directly to the keyboard controller. There is also a KEYLOCK pin which should be connected to the KEYLOCK connector. If KEYLOCK is active, the keyboard controller will not respond to inputs from the keyboard. The keyboard interrupt request is internally connected to the CY82C693U's interrupt controller.

Maximum Flexibility

The internal keyboard controller can be disabled if a custom, external keyboard controller solution is desired. All of the signals needed to control and interface to an external keyboard controller are multiplexed with existing keyboard interface signals.

Power Management Logic

The CY82C693U provides flexible power management to help systems conform to governmental system power consumption guidelines. There are 5 timers within the power management logic (a standby timer, a suspend timer, user timer 1, user timer 2, and user timer 3). There are also 10 programmable event detectors.

Events which can be monitored include:

- Keyboard Commands
- Serial Port Commands
- Parallel Port Commands
- Hard Disk Commands
- DMA/ISA MASTER Requests
- A Specific (set of) Interrupt Request(s)
- Video Memory Accesses
- Floppy Drive Accesses
- A PCI Master Request
- A Specific I/O Range

The CY82C693U can monitor any combination or all of the above events.

Hardware power management can be selected to ease the power-down software requirements. In the hardware power management mode, the $\overline{\text{STOPCLK}}$ signal may be pro-

grammed to automatically assert when the suspend and/or standby timers expire. This signal is used to reduce the power consumption of the processors and peripherals when they are idle.

Software power management works with CPU SMM (System Management Mode). All Pentium-class processors have SMM capabilities. SMM is entered through a dedicated System Management Interrupt ($\overline{SMI}$). SMM has its own protected address space, which can only be accessed in System Management Mode. All power management software should be embedded in the SMI handler code stored in SMM space. SMM memory control is handled by the CY82C691 (please see CY82C691 datasheet). However, all $\overline{SMI}$ generation is handled by the CY82C693U.

The CY82C693U supports Full-Speed, Standby, and Suspend power-down states. Other user-defined power-down states can be implemented. Full-Speed is the normal operating state. When power-management is enabled, the Standby timer will begin counting. The terminal count for the Standby timer can be set to various values between 0.2 seconds and 240 minutes. If any of the monitored events occur before the terminal count is reached, the standby timer will reset to zero and begin counting again. If the standby timer expires without detecting any monitored events, the CY82C693U will assert $\overline{SMI}$, allowing the $\overline{SMI}$ handler to transition the system into the Standby state. The $\overline{SMI}$ signal must be cleared (deasserted) by the $\overline{SMI}$ handler by performing a write to the CY82C693U's internal $\overline{SMI}$ clear register (register E7H, bit 6). Once Standby state has been entered, the Suspend timer will begin counting. The terminal counts for the suspend timer can be set to values between 1 second and 480 minutes. If a Standby event is detected by the CY82C693U before the suspend timer expires, the CY82C693U will assert $\overline{SMI}$, reset all timers (only the Suspend timer will be reset if the event is a Suspend event), and set a status register bit to identify the monitored event. The System Management Interrupt handler must determine the source of the $\overline{SMI}$ by reading CY82C693U status registers. The handler code can use the $\overline{SMI}$ source information to determine which state to transition to. If the suspend timer expires before any monitored event occurs, the CY82C693U will once again assert $\overline{SMI}$ so that the handler can transition to the Suspend state

The user-definable timers (User timer 1, User timer 2, and User timer 3) can be used to create additional power-down states. They can be enabled or disabled at any time. The user timer terminal count values can be programmed to values between 1 second and 480 minutes. When any of the timers expire, the CY82C693U will assert $\overline{SMI}$ (until it is cleared in the handler) and set a configuration register bit to indicate which timer(s) expired. The System Management Interrupt handler must read status registers to determine the source of the $\overline{SMI}$. Software can then transition to any power-down state (pre-defined or user-defined). Only User timer 1 is programmable to be reset upon the detection of monitored events.

There are two power control signals used by the CY82C693U ($\overline{STOPCLK}$, and $\overline{EPMI}$). The first signal ($\overline{STOPCLK}$) is register controlled and can be asserted or negated in any state. $\overline{STOPCLK}$ has been traditionally connected to the CPU's $\overline{STOPCLK}$ input to disable the clock to the internal CPU's core. When $\overline{STOPCLK}$ is enabled, it will periodically be asserted and deasserted based on programmable $\overline{STOPCLK}$ period registers. This signal may also be used to power-down peripherals that support a power-down signal. External logic can force the CY82C693U to assert $\overline{SMI}$ by activating the $\overline{EPMI}$ input.

The CY82C693U contains an internal interrupt/event counter. The terminal count for this timer should be programmed to the longest amount of time it takes to handle any interrupt. Whenever a user-selectable event (typically an interrupt) occurs, the $\overline{STOPCLK}$ signal will be deasserted for the time period of the interrupt timer. This allows the processor to handle the interrupt. If no other monitored events occur before the interrupt timer expires, $\overline{STOPCLK}$ will be reasserted.

AT Refresh Logic

The CY82C693U contains logic to support refresh cycles on the ISA bus. An internal refresh request is generated every 15.6 microseconds. Upon detecting the refresh request, the CY82C693U will arbitrate for the ISA bus and generate a refresh cycle. The CY82C693U contains its own internal refresh counter and refresh address counter. During ISA refresh cycles, the CY82C693U will not grant the ISA bus to any peripheral cards. If a PCI transaction is targeting ISA, the CY82C693U will attempt to buffer the transaction. If the internal FIFO buffers are full or the transaction cannot be buffered, the CY82C693U will negate $\overline{TRDY}$ until the refresh is completed.

Pre-Read/Post-Write Buffers

There are 3 FIFOs inside the CY82C693U. PCI/ISA FIFOs are 2 entries deep and 32 bits wide. There is one 4 entries deep by 32 bit wide FIFO for each IDE channel. PCI to ISA or IDE transactions are buffered to improve system performance. ISA or IDE transactions will likewise be buffered for transmission to the PCI bus. PIO IDE transactions go through their own independent buffers. The buffers eliminate some of the latency due to arbitration for resources.

BIOS ROM Control

The CY82C693U provides all of the control signals to support both Flash and conventional ROM. There is also a dedicated XD bus to buffer the ROM data (provided IDE DMA, Independent IDE Interrupt Requests, and several of the general purpose I/O functions are not required). The XD bus can also be externally buffered, which frees pins for other functions. If Flash ROM is used, the CY82C693U will provide the write-protection through an internal Read Only register bit.

Boot-block Flash is supported through hardware strap pins. The size of the ROM and whether or not the upper address bit is inverted (or not) can be strapped. With boot-block Flash, the boot-block can be write protected and can be used to store recovery code (to rebuild the BIOS from disk, network, etc.). Inverted upper address is normal operating mode. This allows BIOS to begin in non-boot-block memory. If the BIOS is corrupted during an update, the CY82C693U should be strapped to not invert the upper ROM address bit, thus operating out of the protected boot-block space.

Timer/Counter Logic

The CY82C693U contains an 8254-compatible timer/counter. It can be used to generate software time delays, count in binary or BCD, generate interrupts, or generate square-wave patterns. There are three independent, 16-bit counters that can operate in the following six modes: Interrupt on terminal count, Hardware retriggerable one-shot, Rate generator, Square wave generator, Software triggered strobe, and Hardware re-

triggerable strobe. The output of Counter 0 is internal only (can be programmed to generate interrupts). The output of Counter 1 is also internal and works with the AT refresh logic to generate ISA refresh cycles. The output of Counter 2 comes out on the SPKR pin and can be used in any of the modes for a variety of functions, including generating a square wave to an external speaker.

DMA Controllers

The CY82C693U contains two 8237-compatible DMA controllers cascaded together to provide seven separate DMA channels. Internally, each controller is a 4-channel DMA device which generates the memory address and control signals necessary to transfer data between an ISA peripheral device and system memory (via the PCI bus). The two DMA controllers (DMACs) are configured to provide four 8-bit DMA channels and three 16-bit DMA channels. Channel 0 of the 16-bit DMAC is used to cascade the two controllers. The DMA controllers support type A, B, and F transfer rates.

The DMACs operate in one of three states at all times: IDLE, PROGRAM, or ACTIVE.

After the DMACs have been initialized, they remain in IDLE until a DMA request signal is asserted. Once a DMA request is seen, the DMAC that receives the request will enter the ACTIVE state, or if the system is programming the internal configuration registers of a DMAC, it will enter the PROGRAM state. In the IDLE state, the DMACs do nothing aside from sampling the DMA request signals and configuration register decode signals. If a request comes in at the same time as a configuration register access, the register access is executed first.

The first thing the DMACs do, after entering the ACTIVE state, is arbitrate for the PCI bus. The CY82C693U must obtain ownership of the PCI bus regardless of whether the target is on PCI or not. After the PCI bus has been won, the CY82C693U will issue DMA acknowledge to the highest priority requestor. The DMA transfer is then free to begin. The DMAC is in control of the ISA command signals and IOCHRDY during DMA.

The DMACs will enter the PROGRAM state any time a PCI address is decoded to I/O Port 22H. This is the Index Register Port (IRP) for DMAC configuration space. A read or write to Port 23H should immediately follow the write to the IRP. Port 23H is the Data Register Port (DRP).

DMA Controller Transfer Modes

The DMACs can be programmed (on a per channel basis) to transfer data in four modes: Single Transfer Mode, Block Transfer Mode, Demand Transfer Mode, or Cascade Mode.

In Single Transfer Mode, the DMACs will transfer one byte/half-word of data (8 or 16 bits) at a time. A DMAC must arbitrate for the PCI bus after each transfer. This ensures that the PCI bus bandwidth will not be completely consumed by an ISA peripheral. The Word Count Register (WCR) will decrement from the block count until zero is reached. If autoinitializing is selected, the channel will reinitialize itself for the next DMA transfer. Otherwise, the channel will be masked until it is initialized.

In Block Transfer Mode, the DMACs will hold onto ownership of the PCI bus and continue performing transfers until the WCR is decremented to zero. This will give the best DMA performance but runs the risk of degrading overall system performance by tying up the PCI bus for long periods. The channel must once again be initialized/autoinitialized after the block transfer has been completed.

In Demand Transfer Mode, the DMACs will continue performing transfers until the WCR contains zero or the DMA request is negated. The peripheral will negate its DMA request when it is not ready to transfer data (e.g., an I/O device with its buffer full). When the device is ready to resume the transfer, it must assert its DMA request again and rearbitrate for the ISA and PCI buses. This allows higher priority bus masters to access the buses while the peripheral involved in the DMA cycle is not ready. However, if the device remains ready to transfer data, it can hold its DMA request active and perform back-to-back transfers without arbitrating between transfers. The channel must be initialized/autoinitialized after the block transfer has been completed.

Seven DMA channels are available in the CY82C693U. If additional channels are required, channels may be placed in Cascade Mode. When a channel is in Cascade Mode, external 8237 HOLDREQ and HLDA signals can be routed to the channel's DMA request and DMA acknowledge signals respectively. The arbitration rotating protocol will be maintained.

Read, Write, and Verify transfers can be performed in any of the transfer modes. Any channel may also be configured to perform autoinitialization. During autoinitialization, the original values are automatically restored to the Base Address Register and Word Count Register from the Current Address Register and Current Word Count Register when the block transfer is complete (zero in the WCR or an EOP is signalled from the peripheral). The channel will not automatically be masked if autoinitialization is selected.

IDE Controller

The CY82C693U contains an integrated, dual-channel PCI to IDE bridge. The IDE controller conforms to ANSI modes 0, 1, 2, 3, and 4 for PIO (Programmed I/O) transfers. Single-word and multi-word DMA transfer modes 0, 1, and 2 are also supported. ATAPI (CD ROM) protocols are allowed with pre-fetching disabled. The controller allows for CHS (Cylinder-Head-Sector) or LBA (Logical Block Address) addressing. Each of the two channels support two devices for a maximum of four IDE devices in the system. The controller allows for PIO pre-fetching, post writing, and DMA PCI mastering in order to increase overall system performance.

Real-Time-Clock

The CY82C693U contains 14 bytes for a complete time-of-day clock with alarm, one hundred year calendar, a programmable periodic interrupt generator, and 242 bytes of battery-backed "scratch" RAM. The entire Real-Time-Clock (RTC) remains operational under normal or battery power. A 32 kHz oscillator is integrated as part of the RTC. Only an external crystal and a battery are required to complete the clock circuit.

RTC Address Map

The internal RTC contains 256 bytes of battery-backed RAM (14 bytes for clock data and 242 bytes of user-definable RAM). *Figure 2* shows the address map.

		Hex Address (Index)
Accessed Using Ports 70H–index and 71H–data	Seconds	00
	Seconds Alarm	01
	Minutes	02
	Minutes Alarm	03
	Hours	04
	Hours Alarm	05
	Day of the week	06
	Numerical date of the month	07
	Month	08
	Year	09
	RegisterA	0A
	RegisterB	0B
	RegisterC	0C
	RegisterD	0D
	114 Bytes of User-Defined RAM (Configuration Storage)	0E–7F
Ports 72H–index and 73H–data	128 Bytes of additional User-Defined RAM (Configuration Storage)	80–FF

Figure 2. Real-Time-Clock Address Map

Update cycles are performed once per second. The update cycle increments the seconds byte and increments the minutes byte on an overflow (followed by hour, day, month, year, etc.). Alarm value comparisons are also made. During the update cycle, data is undefined. However, the CY82C693U contains an UIP (update in progress) bit to inform the system of updates. If the processor, upon polling the appropriate RTC status register, sees the UIP bit set, an access to the RTC clock data should not be performed. An update-ended interrupt may also be programmed to inform the processor that the update cycle has ended and RTC data is valid.

External RTC Control

To allow for maximum flexibility, an external RTC can be used. The control signals for an external RTC are only used when the internal RTC is disabled.

Interrupt Controllers

The CY82C693U contains two interrupt controllers (INTC1 and INTC2) that provide 82C59A functionality. There are fifteen separate interrupt request inputs (although some of the interrupt requests are only available internally). The two controllers are cascaded to maintain AT interrupt priorities. Interrupt arbitration can be programmed to be rotating or fixed. When interrupt requests come in from the system, the CY82C693U will store all requests, evaluate the priority, and respond with the appropriate acknowledge vector for the CPU's first interrupt acknowledge cycle. Then, if automatic end-of-interrupt (AEOI) is selected, the internal interrupt pending bit will be reset on the second CPU interrupt acknowledge cycle. If AEOI is not selected, it is the responsibility of the software to generate the appropriate end-of-interrupt (EOI) sequence at the end of the interrupt service routine to clear down the interrupt (for example, if a clock interrupt is requested, the system designer may wish the pending interrupt request to remain active through the entire service routine. This allows higher priority interrupts to prematurely force an exit from the clock service routine without causing the loss of the clock interrupt request. If AEOI were programmed, the request to the CPU and the interrupt pending bit would automatically be reset before the clock service routine was entered. If a higher priority interrupt forced the clock routine to be exited, the clock interrupt would be lost.)

The interrupt controllers (INTCs) are initialized and programmed using special command words issued by the CPU. Initialization Command Words (ICW1 through ICW4) bring the INTCs to a known state when the system is first powered-up or reset. Operational Command Words (OCW1 through OCW3) setup the various operating modes.

The following events occur automatically in the initialization sequence:

- The edge sense circuit is reset. Therefore, a request must make a LOW-to-HIGH transition to be detected. If the interrupt request is programmed to be edge-detected, the rising edge must be used (LOW-to-HIGH). High-to-Low transitions will be ignored.
- The interrupt mask register is cleared. Interrupts are enabled.
- Interrupt Request (IR) 7 is set to the lowest priority on each INTC.
- Special Mask Mode is reset.
- Status Read is set to return the value in the Interrupt Request Register (IRR).

The following options are programmable through the ICWs and OCWs:

- Vector Mode operation: The CY82C693U can generate the vector for an interrupt acknowledge cycle.
- The call address interval: The number of cycles to generate an interrupt vector can be chosen to be 4 or 8.
- Edge vs. Level Sensitive Interrupt Requests: The interrupt request lines can be programmed to be detected on a rising edge or a fixed level on an individual request basis.
- Vector Address Byte: The value can be programmed.
- Automatic End-Of-Interrupt: The pending interrupt can be programmed to clear automatically with the acknowledge cycle. If AEOI is not programmed, interrupts must be cleared within the interrupt service routine by the software.
- Interrupt Nesting: Interrupt nesting is a programmable option. If interrupt nesting is allowed, interrupts can be asserted within interrupt service routines.
- Interrupt Masking: The bits of the Interrupt Mask Register can be set to mask (ignore) certain interrupt requests.
- Interrupt Priority: Priority can be programmed to be rotating or fixed.
- Polling: When Polling is selected, Interrupt Requests will not issue an interrupt to the CPU. They will merely set bits within the Interrupt Pending Register (IPR). It is the responsibility of the CPU to periodically read (poll) the IPR to determine if an interrupt is pending.

PCI and IDE interrupts are fully routable internally to independent interrupt levels. The interrupt level of each PCI or IDE interrupt is controlled with configuration registers.

Stand-Alone Operation

The CY82C693U was designed to be used as part of the hyperCache chipset or as a stand-alone PCI peripheral controller. The stand-alone option can be used in any PCI system. The system CPU need not be X86-based, provided the PCI specification is followed. Special options were added to the CY82C693U to make stand-alone operation very flexible. The options include: the ability to disable the integrated PCI arbiter; the ability to split the GNTBSY signaling protocol onto separate GRANT and BUSY signals; the ability to reset both the CY82C693U and the ISA bus when PCIRST is driven active from an external source; the ability to bypass the FREQACK protocol for DMA transfers into memory; and the ability to decode 32-bit I/O space.

Use With An External PCI Arbiter

When using an external PCI arbiter, GNT3/DISARB (pin 194) should be tied LOW through a 1K Ohm pull-down resistor. This will disable the PCI arbiter that is contained in the CY82C693U.

Pin 191 becomes the PREQ output when the internal arbiter is disabled. PREQ is the primary PCI request signal. In the CY82C693U, PREQ is used to request the PCI bus for ISA DMA/Master transactions, Bus Master IDE transactions, and USB bus master transactions. PREQ should be connected to one of the request inputs of the external arbiter.

Pin 196 becomes the PGNT input when the internal arbiter is disabled. PGNT is the primary PCI grant signal. The external arbiter should assert PGNT to allow the CY82C693U to take ownership of the PCI bus. PGNT should be connected to the grant output that corresponds to the PREQ request input. If PGNT is asserted to the CY82C693U without a pending PREQ, the CY82C693U will follow bus parking rules as specified in the PCI rev. 2.1 specification.

Pin 192 becomes the SREQ input when the internal arbiter is disabled. SGNT is the secondary PCI grant signal. Pin 198 becomes the SGNT output when the internal arbiter is disabled. They are used as optional USB arbitration signals. SREQ and SGNT can be programmed to provide separate arbitration for the USB Host Controller. By default, the USB Host Controller requests use of the PCI bus and is granted the bus through PREQ and PGNT. When programmed for separate arbitration (PCI Configuration Registers, Function 0, index=4DH, bit 7), SREQ and SGNT are used to arbitrate for USB, and PREQ and PGNT are used for other bus masters (IDE and ISA DMA).

The integrated Real-Time-Clock contains logic to generate a square wave. When the internal PCI arbiter is enabled, the RTC square wave is not available outside of the CY82C693U. When the internal PCI arbiter is disabled, the RTC square wave is driven on pin 193.

Splitting GNTBSY

When the PCI arbiter inside the CY82C693U is enabled, GNTBSY functions as the arbitration signal between the CY82C693U and the CPU-to-PCI bridge. The CPU-to-PCI bridge is normally granted use of the PCI bus (to reduce the arbitration latency that can hurt CPU performance). The CY82C693U holds GNTBSY asserted to signal that the bus is granted to the CPU-to-PCI bridge. When another PCI master requests use of the bus, the CY82C693U takes away the CPU-to-PCI bridge's grant by deasserting GNTBSY for one clock cycle and then placing GNTBSY in a high-impedance state. The CPU-to-PCI bridge has two PCI clock cycles after sampling GNTBSY deasserted, to signal that it is not finished with the PCI bus (bus busy) by reasserting GNTBSY. If GNTBSY is not sampled asserted in the third clock cycle after the original deassertion, the CY82C693U will grant the PCI bus to the highest priority requesting master. However, if GNTBSY is reasserted by the CPU-to-PCI bridge, the CY82C693U will not grant the bus to any other PCI masters until GNTBSY is deasserted again by the CPU-to-PCI bridge. If the PCI bus has not been signalled busy by the CPU-to-PCI bridge, all pending requests will be serviced before GNTBSY is asserted again by the CY82C693U to grant the bus back to the CPU-to-PCI bridge.

The GNTBSY protocol is implemented using a single signal. This may cause problems when attempting to interface the CY82C693U to a CPU-to-PCI bridge that does not support single signal handshaking. Therefore, the CY82C693U provides the option to separate GNTBSY into a GRANT signal and a BUSY signal.

If pin number 179 (SMI/GBSEP) is pulled down through a 1K Ohm resistor, the GNTBSY signalling protocol is split into separate GRANT and BUSY signals. Pin 189 becomes a constantly driving GRANT output. GRANT will be asserted to grant ownership of the PCI bus to the CPU-to-PCI bridge. Pin 163 becomes the BUSY input. BUSY is asserted by the CPU-to-PCI bridge to signal that the PCI bus is busy and should not be granted to any other PCI master.

External Reset Control

System reset is normally controlled by the CY82C693U. When the PWGD signal is deasserted, the CY82C693U's internal circuitry is reset and CPURST is driven active. PCIRST is placed in high-impedance until the CY82C693U samples pin 172 HIGH (default with internal pull-up). Then PCIRST is driven active. When PWGD goes active, the CY82C693U will continue to drive CPURST and PCIRST active for a minimum of 1 ms.

If an external reset agent is desired, pin 172 should be pulled to ground through a 1K Ohm resistor. If pin 172 is sampled LOW while PWGD is deasserted, the PCIRST output buffer will be placed in high-impedance. This requires the external agent to drive PCIRST (internal pull-up resistor is used). Once PWGD is asserted, the state of pin 172 will be internally latched into the CY82C693U. If pin 172 is pulled LOW, PCIRST becomes an input to the CY82C693U. CPURST will still be driven active for a minimum of 1ms after the assertion of PWGD. After PWGD goes active, the PCIRST input can be driven active by an external agent to reset the internal circuitry of the CY82C693U. PCIRST will also be inverted and driven out on the CPURST output. ISA bus is also reset by the assertion of PCIRST. PCIRST is a synchronous input (sampled on PCI clock) so the minimum guaranteed CPURST would be 1 PCI clock cycle (if PCIRST were only driven active for 1 PCI clock cycle).

FREQACK Bypassing

The CY82C693U uses a flush request/acknowledge handshake. The flush request/acknowledge insures that there is a coherent path whenever ISA or IDE DMA masters attempt to access system memory. Data coherency may be violated anytime there is a temporary storage element (such as a post write buffer) that does not monitor transfers of data to provide the

most up-to-date copy. In other words, data may be stored in a post-write buffer which is more current than the data in DRAM memory. An ISA/IDE DMA master may get the wrong data if the access is allowed to proceed to main memory "around the post-write buffer."

To eliminate a data coherency problem, the CY82C693U will issue a "flush request" before allowing an ISA/IDE DMA transfer to proceed. Flush request is signalled when the CY82C693U asserts FREQACK for one PCI clock cycle. The CY82693U will then deassert FREQACK, place the output in high-impedance, and begin monitoring the input. When all post-write buffers have been emptied, the system controller will assert FREQACK for one PCI clock cycle. This signifies an acknowledge. The CY82C693U will proceed with the DMA transfer.

If a coherent path to memory can be guaranteed without flushing storage elements (i.e., the post-write buffers "snoop" transactions and provide current data, or there are no post-write buffers), the flush request/acknowledge protocol may be bypassed. If PCI configuration register 4DH, bit 6 is set to "1", the flush request will be internally acknowledged. The FREQACK signal (pin 190) will be unused and placed in a high-impedance state.

When the flush request/acknowledge protocol is bypassed, the RTC interrupt request (IRQ8) may be optionally driven on pin 190. This can be used to provide immediate, non-masked servicing for Real-Time-Clock interrupts. External IRQ8 generation is controlled by PCI configuration register 4DH, bit 5. If bit 5 is "0", IRQ8 from the RTC circuit is only routed internally to the integrated interrupt controller. It is not available on an external CY82C693U pin. If, however, register 4DH, bit 5 is set to "1", the IRQ8 output from the RTC circuitry is masked to the internal interrupt controller, and IRQ8 is driven on pin 190 of the CY82C693U. NOTE: If the flush request/acknowledge protocol is not bypassed, FREQACK is driven on pin 190 and if register 4DH, bit 5 is set to "1" the RTC interrupt will only be masked (IRQ8 will not be available externally).

32-Bit I/O Space Decode

As X86 processors only support 64 Kbytes of I/O space, the CY82C693U's IDE controllers only decode the lower 16 bits of the PCI address. This is sufficient for PC chipset applications. However, for stand-alone operation, it may become necessary to decode 32 address bits (4GB) of I/O space. If PCI configuration register 4DH, bit 4 is set to "1", 32-bits are decoded for I/O space. This effects the format of IDE Base Address Registers in PCI configuration space. When register 4DH, bit 4 is "0", the upper 16-bits of the IDE Base Address Registers (10H, 14H, and 20H) are hardwired to 0000H. When register 4DH, bit 4 is "1", the upper 16-bits are readable and writable. The value written corresponds to the decoded address range (as specified in the PCI specification, revision 2.1).

1 Mbyte ROM Decode

The CY82C693U can support Extended ROM decoding up to 1 MB at the top of memory space. If bit 3 of the PCI configuration register 4DH is set to "0", 512 KB Extended ROM address decoding is supported (FFF80000H to FFFFFFFFH). If bit 3 is set to "1" 1 MB Extended ROM address decoding is enabled (FFF00000H to FFFFFFFFH).

Universal Serial Bus (USB) Host Controller

The CY82C693U contains a USB host controller. The Host Controller integrates a root hub with two USB ports; therefore, two USB peripheral devices can be connected to the CY82C693U without any extra logic. An external hub can be connected to any of the two integrated ports in case more devices are required. The Host Controller is connected to system memory via the PCI bus and has bus mastering capability. The USB Host Controller's PCI configuration registers are located in PCI function 3 configuration space in CY82C693U.

The CY82C693U USB Host Controller fully complies with the USB Open Host Controller Interface (OpenHCI) standard, thus allowing it to be compatible with the available OpenHCI standard software drivers. *Figure 3* illustrates different hierarchical domains of a USB system.

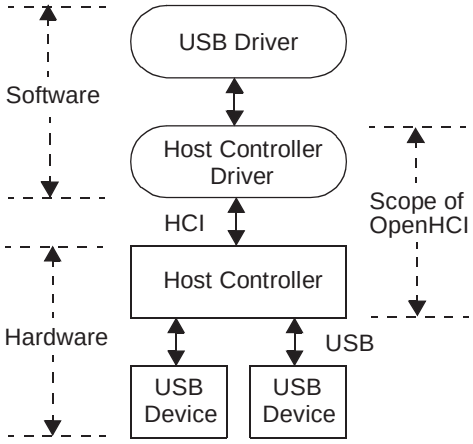


Figure 3. USB System Domains

The domains are the USB Driver (USBD), Host Controller Driver (HCD), Host Controller (HC), and USB Device. The USB Driver and the Host Controller Driver are part of system software, and HCD is typically provided by the operating system. The Host Controller and the USB devices are implemented in hardware. The interface between the software layer (HCD) and the hardware (HC) is called Host Controller Interface (HCI), which is specified in the OpenHCI standard. The basic building block for communication across the interface is called Transfer Descriptor (TD). The Transfer Descriptor contains all necessary information for the Host Controller to process packet transaction with USB devices. The HCD software is responsible for communicating information between the USBD software layer (also typically located in the operating system) and the Host Controller. The HCD receives Transfer Descriptors from the USBD, reformats them as necessary, schedules their execution and enqueues them for the processing by the Host Controller. When the Host Controller finishes the processing of the Transfer Descriptors, the HCD reformats them as necessary, and sends the results back to the USBD. The HCD is also responsible for status and error conditions monitoring.

The Host Controller moves data between system memory and devices on the USB by processing Transfer Descriptors enqueued by the HCD and generating transactions on USB. When the Host Controller sends information to a USB device, the data from system memory pointed to by a Transfer De-

scriptor is converted to the USB serial protocol and transferred on the USB with the appropriate headers. When the Host controller receives information from a USB device, the data is converted from the USB serial protocol and stored in the system memory location pointed to by a Transfer Descriptor. The Host Controller is also responsible for reporting the status of transactions on USB to the HCD. Some other tasks performed by the Host Controller include the maintenance of USB frame generation and the reporting of USB device connection activity to the HCD.

For more detailed information on the USB Host Controller, see USB Open Host Controller Interface (OpenHCI) Specification, revision 1.0 available from Microsoft, Compaq and National Semiconductor. For additional information on USB, see USB Specification revision 1.0 from USB Implementers Forum.

CY82C693U Signal Description

The CY82C693U signals are divided into eight functional areas: Reset signals, PCI Interface signals, ISA Interface signals, Power Management signals, Keyboard Interface signals, IDE Interface signals, USB interface signals, and Miscellaneous signals.

Reset Signals

Name	I/O	Description
PWGD	I	Power Good: This signal is driven active from a combination of the external power supply's power good signal and the external reset switch. This signal is used to qualify initialization signals and reset the internal state of the CY82C693U.
CPURST	O	CPU Reset: This signal resets the CPU and the ISA bus.
$\overline{\text{PCIRST}}$	O	PCI Reset: This signal functions as the PCI bus reset. During normal operation, this signal is an output only (used by the CY82C693U to reset PCI bus residents). If pin 172 is pulled LOW through a 1K Ohm resistor, $\overline{\text{PCIRST}}$ becomes an input that is used to initiate a CY82C693U reset and an ISA reset. See the text description for Stand-Alone Operation. This signal will power up in a high-impedance state with an internal pull-up resistor. When pin 172 is sampled, this signal will either begin driving (pin 172 HIGH) or remain a high-Z input (pin 172 LOW).
INIT	O	CPU Initialization: This signal is used to reset the core of the CPU without disturbing the state of internal caches or write-buffers. This signal can be used to switch the processor from protected mode to real mode.

PCI Interface Signals

Name	I/O	Description
AD[31:0]	I/O	PCI Address/Data Bus: Multiplexed bidirectional address/data lines on the PCI bus. The CY82C693U either drives or samples these lines during PCI cycles.
PCICLK	I	PCI Clock: PCI Clock Input. This signal is used to synchronize the CY82C693U to the PCI bus. The clock must be within the range 25 MHz to 33 MHz.
C/$\overline{\text{BE}}$[3:0]	I/O	PCI Command & Byte Enables: C/$\overline{\text{BE}}$[3:0] are driven by the current bus master during the address phase to define the transaction and during the data phase as the byte enables. These signals are outputs when the CY82C693U is a master and inputs when the CY82C693U is a slave.
$\overline{\text{FRAME}}$	I/O	Cycle Frame: Driven by the current bus master to indicate the start and duration of a transaction.
$\overline{\text{IRDY}}$	I/O	Initiator Ready: The assertion of $\overline{\text{IRDY}}$ indicates the current bus master's ability to complete the current data phase of the transaction. Used in conjunction with $\overline{\text{TRDY}}$ from the target.
$\overline{\text{TRDY}}$	I/O	Target Ready: The assertion of $\overline{\text{TRDY}}$ indicates the current target's ability to complete the current data phase of the transaction. Works in conjunction with $\overline{\text{IRDY}}$ from the master.
$\overline{\text{DEVSEL}}$	I/O	Device Select: Indicates that a PCI device has decoded that it is the target of the transaction. The target has three options for decoding: fast decoding, medium decoding, or slow decoding. The CY82C693U will sample $\overline{\text{DEVSEL}}$, and if it is not asserted by a target within the timeout period, will assert $\overline{\text{DEVSEL}}$ to claim the cycle.
PAR	I/O	Parity: An even parity bit across AD[31:0] and C/$\overline{\text{BE}}$[3:0]. As a master the CY82C691 generates even parity on PCI write cycles. On read cycles, the CY82C693U checks parity by sampling PAR.

PCI Interface Signals (continued)

Name	I/O	Description
$\overline{\text{STOP}}$	I/O	Stop: Indicates that the current target is requesting the master to stop the current transaction. $\overline{\text{STOP}}$ is used in conjunction with $\overline{\text{DEVSEL}}$ and $\overline{\text{TRDY}}$ to indicate disconnect, target abort, and retry cycles.
$\overline{\text{SERR}}$	I/O	System Error: System error may be asserted by any agent for reporting address parity errors or any other types of errors besides data parity. $\overline{\text{SERR}}$ will cause the CY82C693U to assert NMI to the processor.
IDSEL	I	PCI ID Select: This signal should be connected to a unique PCI address. It is used to select the CY82C693U during PCI configuration cycles.
$\overline{\text{INTA/B/C/D}}$	I	PCI Interrupt Requests: These signals allow PCI peripherals to interrupt the processor.
$\overline{\text{GNTBSY/GRANT}}$	I/O	CY82C691 Busy/Grant: This signal is the PCI arbitration signal used by the CY82C691. The CY82C691 is the default owner of the PCI bus. In the default state, the CY82C693U will actively drive this signal asserted (LOW). If another external master requests the bus, the CY82C693U will deassert this signal for a single clock cycle and then let the signal float. If the CY82C691 asserts this signal within two clock cycles after this signal is seen deasserted, this signal becomes $\overline{\text{BUSY}}$ to the central arbiter. $\overline{\text{BUSY}}$ tells the CY82C693U that the CY82C691 owns the PCI bus. The CY82C693U will not grant the PCI bus to any other PCI master until $\overline{\text{BUSY}}$ is deasserted by the CY82C691. When no requests are pending on the PCI bus, this signal reverts to 691 $\overline{\text{GRANT}}$ and is driven active (LOW) to allow the CY82C691 to take possession of the PCI bus. If pin 179 is strapped LOW through a 1K Ohm resistor, this signal becomes a dedicated $\overline{\text{GRANT}}$ output. $\overline{\text{GRANT}}$ is constantly driven by the CY82C693U.
$\overline{\text{FREQACK/IRQ8}}$	I/O	CY82C691 Flush Request/Acknowledge or RTC Interrupt Request: This signal is used to facilitate DMA transfers. The CY82C693U will drive this signal LOW for one clock cycle to request the CY82C691 flush its internal buffers before a DMA transfer. Once the CY82C691 has flushed its buffers, it will drive this signal for one clock cycle to acknowledge the flush. The CY82C693U is then free to perform its DMA transfer. If PCI configuration register 4DH, bit 6 is set to “1”, all flush requests are immediately acknowledged inside the CY82C693U. This signal becomes high-impedance. If PCI configuration register 4DH, bit 5 is also set to “1”, the Real-Time-Clock interrupt request ($\overline{\text{IRQ8}}$) is masked to the internal interrupt controller and driven out on this pin.
$\overline{\text{REQ}}$[3:2]	I	PCI Bus Requests: These signals are connected to the individual bus requests from each PCI peripheral. When a combination of the bus requests is asserted, the CY82C693U will resolve the priority and give the grant to the highest priority master. If the internal PCI arbiter is disabled (by strapping pin 194 LOW through a 1K Ohm resistor), these signals become UNUSED.
$\overline{\text{REQ[1]/SGNT}}$	I/O	PCI Bus Request 1/ Secondary PCI Grant: If the internal PCI arbiter is enabled, this signal is connected to the individual bus request from PCI peripheral number 1. When a combination of the bus requests is asserted, the CY82C693U will resolve the priority and give the grant to the highest priority master. If the internal PCI arbiter is disabled (by strapping pin 194 LOW through a 1K Ohm resistor), this signal becomes a RESERVED input. In the CY82C693UU this signal will optionally become the secondary grant input to grant the PCI bus to the USB Host Controller.
$\overline{\text{REQ[0]/PGNT}}$	I/O	PCI Bus Request 0/ Primary PCI Grant: If the internal PCI arbiter is enabled, this signal is connected to the individual bus request from PCI peripheral number 0. When a combination of the bus requests is asserted, the CY82C693U will resolve the priority and give the grant to the highest priority master. If the internal PCI arbiter is disabled (by strapping pin 194 LOW through a 1K Ohm resistor), this signal becomes the $\overline{\text{PGNT}}$ input. $\overline{\text{PGNT}}$ is driven active by the external PCI arbiter to grant ownership of the PCI bus to the CY82C693U.
$\overline{\text{GNT[3]/DISARB}}$	I/O	PCI Bus Grant 3/ PCI Arbiter Disable: This signal is connected to bus grant of PCI peripheral #3. When a combination of the bus requests is asserted, the CY82C693U will resolve the priority and assert grant to the highest priority master. During power-up, this signal acts as a strapping option to disable the PCI arbiter within the CY82C693U. An external PCI arbiter must be provided. When the internal arbiter is disabled, several other signals change function to provide the request/grant signals from the CY82C693U to the external arbiter.

PCI Interface Signals (continued)

Name	I/O	Description
$\overline{\text{GNT[2]}}$/SQWV	O	PCI Bus Grant 2/ RTC Square Wave Output: This signal is connected to bus grant of PCI peripheral #3. When a combination of the bus requests is asserted, the CY82C693U will resolve the priority and assert grant to the highest priority master. If the internal PCI arbiter is disabled (pin 194 strapped LOW during power up), this pin drives the output of the square wave generator in the Real-Time-Clock.
$\overline{\text{GNT[1]}}$/$\overline{\text{SREQ}}$	O	PCI Bus Grant 1/ Secondary PCI Request: If the internal PCI arbiter is enabled, this signal is connected to the individual bus grant to PCI peripheral number 1. When a combination of the PCI bus requests is asserted, the CY82C693U will resolve the priority and assert grant to the highest priority master. If the internal PCI arbiter is disabled (by strapping pin 194 LOW through a 1K Ohm resistor), this signal becomes a RESERVED output. In the CY82C693UU this signal will optionally become the secondary request output to request ownership of the PCI bus for the USB Host Controller.
$\overline{\text{GNT[0]}}$/$\overline{\text{PREQ}}$	O	PCI Bus Grant 0/ Primary PCI Request: If the internal PCI arbiter is enabled, this signal is connected to the individual bus grant to PCI peripheral number 0. When a combination of the bus requests is asserted, the CY82C693U will resolve the priority and assert grant to the highest priority master. If the internal PCI arbiter is disabled (by strapping pin 194 LOW through a 1K Ohm resistor), this signal becomes the $\overline{\text{PREQ}}$ output. $\overline{\text{PREQ}}$ is driven active by the CY82C693U to request ownership of the PCI bus for ISA or IDE DMA and USB Host Controller (if no separate arbitration used) cycles.

ISA Interface Signals

Name	I/O	Description
SA[19:4]/ IDE[15:0]	I/O	ISA Address Bus/PCI IDE Data Bus: These signals provide most of the address for the ISA bus as well as the data path for IDE.
SA[3:0]	I/O	ISA Address: These signals provide the rest of the ISA address.
SD[15:0]	I/O	System Data Bus: These signals connect directly to the ISA data bus.
XD7/IDEIRQ0 GPIO0	I/O	XD Bus, Bit 7/Programmable Interrupt Request 0/General Purpose I/O 0: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 7 of the data bus. Otherwise, this is user-selectable to provide support for a Programmable Interrupt Request or a general purpose I/O. As a GPIO, this signal can either reflect the contents of an internal register bit (output) or set an internal register bit to the value driven on the line (input).
XD6/IDEIRQ1 GPIO1/$\overline{\text{BUSY}}$	I/O	XD Bus, Bit 6/Programmable Interrupt Request 1/General Purpose I/O 1/PCI Bus Busy: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 6 of the data bus. Otherwise, this is user-selectable to provide support for a Programmable Interrupt Request or a general purpose I/O. As a GPIO, this signal can either reflect the contents of an internal register bit (output) or set an internal register bit to the value driven on the line (input). If pin 179 is strapped LOW during power-up ($\overline{\text{GNTBSY}}$ split), this signal becomes the $\overline{\text{BUSY}}$ input. $\overline{\text{BUSY}}$ is asserted by the CPU-to-PCI bridge to tell the PCI arbiter in the CY82C693U that the PCI bus is currently busy and may not be granted to any other PCI master.
XD5/IREQ0/ GPIO2	I/O	XD Bus, Bit 5/IDE DMA Request 0/General Purpose I/O 2: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 5 of the data bus. Otherwise, this is user-selectable to provide support for IDE DMA or a general purpose I/O. For IDE DMA support, this signal is DMA request 0 (primary IDE channel DMA request). As a GPIO, this signal can either reflect the contents of an internal register bit (output) or set an internal register bit to the value driven on the line (input).

ISA Interface Signals (continued)

Name	I/O	Description
XD4/$\overline{\text{IACK0}}$/ GPIO3	I/O	XD Bus, Bit 4/IDE DMA Acknowledge 0/General Purpose I/O 3: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 4 of the data bus. Otherwise, this is user-selectable to provide support for IDE DMA or a general purpose I/O. For IDE DMA support, this signal is DMA acknowledge 0 (primary IDE channel DMA acknowledge). As a GPIO, this signal can either reflect the contents of an internal register bit (output) or set an internal register bit to the value driven on the line (input).
XD3/IREQ1/ GPIO4	I/O	XD Bus, Bit 3/IDE DMA Request 1/General Purpose I/O 4: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 3 of the data bus. Otherwise, this is user-selectable to provide support for IDE DMA or a general purpose I/O. For IDE DMA support, this signal is DMA request 1 (secondary IDE channel DMA request). As a GPIO, this signal can either reflect the contents of an internal register bit (output) or set an internal register bit to the value driven on the line (input).
XD2/$\overline{\text{IACK1}}$/ GPIO5	I/O	XD Bus, Bit 2/IDE DMA Acknowledge 1/General Purpose I/O 5: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 2 of the data bus. Otherwise, this is user-selectable to provide support for IDE DMA or a general purpose I/O. For IDE DMA support, this signal is DMA acknowledge 1 (secondary IDE channel DMA acknowledge). As a GPIO, this signal can either reflect the contents of an internal register bit (output) or set an internal register bit to the value driven on the line (input).
XD1/$\overline{\text{XDEN}}$	I/O	XD Bus, Bit 1/External XD Bus Buffer Enable: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 1 of the data bus. Otherwise, this pin provides the enable for a buffer between XD[7:0] and SD[7:0].
XD0/XDIR	I/O	XD Bus, Bit 0/External XD Bus Direction Control: If zero TTL is desired for XD bus support (BIOS ROM data, external keyboard controller data, and external RTC data), this is bit 0 of the data bus. Otherwise, this pin provides the direction control for a buffer between XD[7:0] and SD[7:0].
LA23/$\overline{\text{IDECS0}}$	I/O	Latched Address 23/ IDE Chip Select0: This signal connects to LA23 on the ISA bus and is used to provide access up to 16 MB. During IDE accesses, this signal provides the chip select 0 for the primary channel.
LA22/$\overline{\text{IDECS1}}$	I/O	Latched Address 22/ IDE Chip Select1: This signal connects to LA22 on the ISA bus and is used to provide access up to 16 MB. During IDE accesses, this signal provides the chip select 1 for the primary channel.
LA21/$\overline{\text{SIDECS0}}$	I/O	Latched Address 21/ Secondary IDE Chip Select0: This signal connects to LA21 on the ISA bus and is used to provide access up to 16 MB. During IDE accesses, this signal provides the chip select 0 for the secondary channel.
LA20/$\overline{\text{SIDECS1}}$	I/O	Latched Address 20/ Secondary IDE Chip Select1: This signal connects to LA20 on the ISA bus and is used to provide access up to 16 MB. During IDE accesses, this signal provides the chip select 1 for the secondary channel.
LA[19:17]/ IDEA[2:0]	I/O	Latched Address 19 Through 17/ PCI IDE Register Select Address 2 Through 0: These signals connect to LA19 through LA17 on the ISA bus and are used to provide access up to 16 MB. During IDE accesses, these signals provide the register select (or address) to the IDE connectors.
IRQ1/KBCLK	I/O	Keyboard Clock/Interrupt Request 1: This signal is the keyboard clock connected to the keyboard connector if the internal keyboard controller is used. Otherwise, it provides IRQ1 (the keyboard controller interrupt input) if an external keyboard controller is desired.
IRQ[15:14], IRQ[11:9] IRQ[7:3]	I	Interrupt Request Inputs: These signals provide interrupt requests for CPU interrupters.
IRQ12/MSCLK	I/O	Mouse Clock/Interrupt Request 12: This signal is the mouse clock connected to the mouse connector if the internal keyboard controller is used. Otherwise, it provides IRQ12 (the mouse controller interrupt input) if an external keyboard controller is desired.

ISA Interface Signals (continued)

Name	I/O	Description
IRQ8/PSRSTB	I	Power Strobe/Interrupt Request 8: If the internal Real-Time-Clock (RTC) is used, this is the power strobe input. If an external RTC is desired, this is interrupt request 8 (traditionally the RTC interrupt).
INTR/KBATMODE	I/O	CPU Interrupt/Keyboard AT Mode: This is the INTR (interrupt request) signal that is connected directly to the CPU's INTR pin. When interrupt requests come in to the CY82C693U, it will assert INTR to the CPU. On power-up this signal should be pulled-down through a 1K Ohm resistor to ground if PS/2 mouse support is desired.
NMI	O	Non-maskable CPU Interrupt: This is the NMI (non-maskable interrupt request) signal that is connected directly to the CPU's NMI pin. When the CY82C693U detects a fatal error (such as a PCI parity error), it will assert NMI to the CPU.
ATCLK	O	AT Clock: This signal can be used to provide the system ISA (AT) bus clock. It can be internally programmed to generate different ISA frequencies.
ALE	I/O	Bus Address Latch Enable: This signal is the ISA BALE signal used by peripherals to latch the cycle address, AEN, and SBHE.
DRQ7/GPIO8/ SMIACT	I/O	ISA DMA/Master Request Input 7/General Purpose I/O 8/USB Host Controller SMIACT Input: This signal is used by external ISA peripherals to request mastership of the ISA bus for DMA or ISA MASTER cycles. They can be programmed to be GPIO if the DREQ signals are connected to an external ('157 type) TTL device to time multiplex (using the ATCLK as the selector signal) the DMA requests onto DRQ[3:0]. When USB Host Controller is enabled, this signal becomes SMI Active input, which is used to mask the SMI output while in SMM. Pull-up is required for SMIACT if unused.
DRQ6/GPIO7/ OVRCUR	I/O	ISA DMA/Master Request Input 6/General Purpose I/O 7/USB Host Controller OVR-CUR input: This signal is used by external ISA peripherals to request mastership of the ISA bus for DMA or ISA MASTER cycles. They can be programmed to be GPIO if the DREQ signals are connected to an external ('157 type) TTL device to time multiplex (using the ATCLK as the selector signal) the DMA requests onto DRQ[3:0]. When USB Host Controller is enabled, this signal becomes Over-current Detection input. OVRCUR is asserted when downstream ports exceed their current allocation. This input causes power to be disabled and is reported through the hub and port status register. Pull-up required for OVRCUR if unused.
DRQ5/GPIO6/ PWREN	I/O	ISA DMA/Master Request Input 5/General Purpose I/O 6/USB Host Controller PWREN output: This signal is used by external ISA peripherals to request mastership of the ISA bus for DMA or ISA MASTER cycles. They can be programmed to be GPIO if the DREQ signals are connected to an external ('157 type) TTL device to time multiplex (using the ATCLK as the selector signal) the DMA requests onto DRQ[3:0]. When USB Host Controller is enabled, this signal becomes Port Power Enable output. The global power to USB ports is controlled by this signal. If NoPowerSwitching in HcRhDescriptorA (USB Host Controller Operational Register) is set, this signal is always active.
DRQ[3:0]	I	ISA DMA/Master Request Inputs: These signals are used by external ISA peripherals to request mastership of the ISA bus for DMA or ISA MASTER cycles. These signals can also be programmed to be time multiplexed (based on ATCLK) to free DRQ[7:5] for GPIO. If they are multiplexed, the multiplexed version of DREQ3 and DREQ7 (with DREQ3 passed through when ATCLK is LOW) should be connected to the DRQ3 pin, the multiplexed version of DREQ1 and DREQ6 (with DREQ1 active when ATCLK is LOW) should be tied to the DRQ1 pin, the multiplexed version of DREQ0 and DREQ5 (with DREQ0 active when ATCLK is LOW) should be tied to the DRQ0 pin, and DREQ2 should be tied directly to the DRQ2 pin.
DACK0/TSTM	I/O	ISA DMA/Master 0 Acknowledge/Test Mode Enable: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor to enable test mode. For normal operation, this pin should not be pulled-down. This signal can also be programmed to generate a DMA acknowledge code that can be sent to the Input A input of a '138 type TTL 1 of 8 decoder. By programming this input to generate a coded output, DACK[7:5] are free to become GPIO.

ISA Interface Signals (continued)

Name	I/O	Description
$\overline{\text{DACK1}}$/TSTM0	I/O	ISA DMA/Master 1 Acknowledge Input/Test Mode Select 0: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor or not pulled-down to select between the different test modes. This signal can also be programmed to generate a DMA acknowledge code that can be sent to the Input B input of a '138 type TTL 1 of 8 decoder. By programming this input to generate a coded output, $\overline{\text{DACK}}$[7:5] are free to become GPIO.
$\overline{\text{DACK2}}$/TSTM1	I/O	ISA DMA/Master 2 Acknowledge/Test Mode Select 1: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor not pulled-down to select between the different test modes. This signal can also be programmed to generate a DMA acknowledge code that can be sent to the Input C input of a '138 type TTL 1 of 8 decoder. By programming this input to generate a coded output, $\overline{\text{DACK}}$[7:5] are free to become GPIO.
$\overline{\text{DACK3}}$/DISPSEL	I/O	ISA DMA/Master 3 Acknowledge Input/Display Type Select: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor or left floating to select between CGA and Monochrome monitors (acts as the keyboard controller Mono/Color pin). This signal can also be programmed to generate an enable to a '138 type TTL 1 of 8 decoder. By programming this input to generate the enable, glitches are prevented when the code signals switch.
$\overline{\text{DACK5}}$/KBSEL /GPIO9	I/O	ISA DMA/Master 5 Acknowledge Input/Internal Keyboard Controller Enable/General Purpose I/O 9: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor to disable the internal keyboard controller if an external keyboard controller is desired. This signal can be used as a GPIO if DACK[3:0] are programmed to connect to an external '138 type TTL 1 of 8 decoder.
$\overline{\text{DACK6}}$/RTCSEL /GPIO10	I/O	ISA DMA/Master 6 Acknowledge Input/Internal Real Time Clock Enable/General Purpose I/O 10: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor to disable the internal RTC if an external RTC is desired. This signal can be used as a GPIO if DACK[3:0] are programmed to connect to an external '138 type TTL 1 of 8 decoder.
$\overline{\text{DACK7}}$/EXTBUF /GPIO11	I/O	ISA DMA/Master 7 Acknowledged Input/Internal IDE Controller Enable/General Purpose I/O 11: This signal is used by the CY82C693U to grant ISA bus mastership to external ISA peripherals for DMA or ISA MASTER cycles. At power-up, this pin should be pulled-down through a 1K Ohm resistor to disable the internal IDE controller if an external IDE controller is desired. This signal can be used as a GPIO if DACK[3:0] are programmed to connect to an external '138 type TTL 1 of 8 decoder.
$\overline{\text{REFSH}}$	I/O	AT Refresh: Driven when an ISA refresh cycle is in progress. As an input, this signal will force a refresh cycle to be initiated.
SPKR	O	Speaker Output: This signal is the output of counter 2 in the timer/counter logic. It can be used to drive a speaker.
EOP	O	End of Process: This signal is driven by the CY82C693U to signal the end of a block transfer during a DMA cycle.

ISA Interface Signals (continued)

Name	I/O	Description
$\overline{\text{IGNNE}}$/ROMS1	I/O	Ignore Numerical Error: This signal is driven by the CY82C693U to the CPU and should be connected to the CPU's $\overline{\text{IGNNE}}$ pin. When this signal is asserted, the processor will ignore numerical errors and continue executing non-control, floating-point instructions. At power-up, this signal acts as ROMS1. ROMS1 is used in conjunction with ROMS0 (GTA20) and ROMMODE ($\overline{\text{ROMCS}}$) to implement boot-block Flash recovery straps. The strapping is defined as follows: ROMMODE / ROMS0 / ROMS1 / Result 1 / X / X / No ROM address bits are inverted. (EPROM or boot-block recovery mode). 0 / 1 / 1 / ROM address bit 16 is inverted (Normal operation w/ 16Kx8 boot-block Flash) 0 / 1 / 1 / ROM address bit 17 is inverted (Normal operation w/ 32Kx8 boot-block Flash) 0 / 0 / 1 / ROM address bit 18 is inverted (Normal operation with 64Kx8 boot-block Flash) 0 / 0 / 0 / ROM address bit 15 is inverted (Normal operation with 8Kx8 boot-block Flash)
$\overline{\text{FERR}}$	O	Floating Point Error: This signal informs the processor that a coprocessor error has occurred.
GTA20/ROMS0	I/O	Gate A20: This signal forces memory to wrap-around 1 MB. It is implemented as a fast gate A20. At power-up, this signal acts as ROMS0. ROMS0 is used in conjunction with ROMS1 ($\overline{\text{IGNNE}}$) and ROMMODE ($\overline{\text{ROMCS}}$) to implement boot-block Flash recovery straps. The strapping is defined as follows: ROMMODE / ROMS0 / ROMS1 / Result 1 / X / X / No ROM address bits are inverted. (EPROM or boot-block recovery mode) 0 / 1 / 1 / ROM address bit 16 is inverted (Normal operation w/ 16Kx8 boot-block Flash) 0 / 1 / 0 / ROM address bit 17 is inverted (Normal operation with 32Kx8 boot-block Flash) 0 / 0 / 1 / ROM address bit 18 is inverted (Normal operation with 64Kx8 boot-block Flash) 0 / 0 / 0 / ROM address bit 15 is inverted (Normal operation with 8Kx8 boot-block Flash)
X2/RTCRD	O	RTC Oscillator Out/RTC Output Enable: If the integrated RTC is used, this signal is the 32.768-kHz crystal output. If an external RTC is desired, this signal is the RTC output enable.

ISA Interface Signals (continued)

Name	I/O	Description
X1/$\overline{\text{RTCWT}}$	I/O	RTC Oscillator In/RTC Write Enable: If the integrated RTC is used, this signal is the 32.768-kHz crystal input. If an external RTC is desired, this signal is the RTC write enable.
$\overline{\text{IOCS16}}$	I	16-Bit I/O Chip Select: This signal is driven active by any ISA I/O target that can support 16-bit accesses.
$\overline{\text{MCS16}}$	I/O	16-Bit Memory Chip Select: This signal is driven active by any ISA MEMORY target that can support 16-bit accesses. During ISA Master cycles, the CY82C693U will drive this signal.
$\overline{\text{SBHE}}$	I/O	System Byte High Enable: This signal is driven active by the current ISA bus master to indicate that valid data resides on SD[15:8].
$\overline{\text{MRD}}$	I/O	ISA Memory Read Command Signal: This signal is driven by the current ISA bus owner to request a memory resource to drive data onto the bus during a cycle.
$\overline{\text{MWT}}$	I/O	ISA Memory Write Command Signal: This signal is driven by the current ISA bus owner to request a memory resource to accept data presented on the bus during a cycle.
$\overline{\text{IOR}}$	I/O	ISA I/O Read Command Signal: This signal is driven by the current ISA bus owner to request an I/O resource to drive data onto the bus during a cycle.
$\overline{\text{IOW}}$	I/O	ISA I/O Write Command Signal: This signal is driven by the current ISA bus owner to request an I/O resource to accept data presented on the bus during a cycle.
$\overline{\text{SMEMR}}$	I/O	ISA System Memory Read Command Signal: This signal is driven by the current ISA bus owner to request a memory resource to drive data onto the bus during a cycle. This signal is only active within the first 1 MB of memory space.
$\overline{\text{SMEMW}}$	I/O	ISA System Memory Write Command Signal: This signal is driven by the current ISA bus owner to request a memory resource to accept data presented on the bus during a cycle. This signal is only active within the first 1 MB of memory space.
OSC	I	Oscillator Input: This signal is the 14.318-MHz reference clock input, used by the timer and DMA controller.
$\overline{\text{0WS}}$	I	Zero Wait State Input: When this signal is driven by an ISA peripheral, the CY82C693U will shorten the ISA cycle to zero wait states and ignore IOCHRDY.
AEN	O	Address Enable: This signal is driven when a valid address is present on the ISA bus. It is deasserted during DMA cycles so that ISA peripherals do not respond to the DMA memory cycle.
IOCHRDY	I/O	I/O Channel Ready: When deasserted, this signal indicates to the ISA bus owner that additional cycle time is required to complete the transaction. Used to add ISA wait states.
$\overline{\text{IOCHK}}$	I	I/O Channel Check: This signal is driven active if the system detects an ISA bus parity error.

Power Management Signals

Name	I/O	Description
$\overline{\text{SMI}}$/GBSEP	I/O	System Management Interrupt/$\overline{\text{GNTBSY}}$ Separate: Used by the CY82C693U to send the highest-level, process-transparent interrupt to the CPU. At power up, this signal is sampled to determine the grant/busy protocol for CPU-to-PCI bridge arbitration. If this signal is left floating (sampled HIGH), $\overline{\text{GNTBSY}}$ arbitration is performed on a single signal. If this signal is strapped LOW through a 1K Ohm resistor, $\overline{\text{GRANT}}$ is driven out on pin 189 and $\overline{\text{BUSY}}$ is sampled on pin number 163.

Power Management Signals (continued)

Name	I/O	Description
$\overline{\text{STOPCLK}}$/ $\overline{\text{RSTCHG}}$	I/O	CPU Stop Clock/External PCI Reset Generator Present: This signal is used to stop the clock to the CPU's core to reduce power-consumption during idle periods. At power up, this signal is sampled to determine the source of PCI reset in the system. If this signal is left floating (sampled HIGH), PCI reset and CPU/ISA reset signals are output from the CY82C693U. They are asserted when PWGD is sampled deasserted and will be driven asserted for a minimum period of 1ms after PWGD goes active. If $\overline{\text{STOPCLK}}$/$\overline{\text{RSTCHG}}$ is strapped LOW through an external 1K Ohm resistor, PCIRST becomes an input. When $\overline{\text{PCIRST}}$ is sampled asserted, all registers/FIFOs/state within the CY82C693U are cleared to a known reset state. CPURST is driven active for the duration of $\overline{\text{PCIRST}}$ active. There is no minimum guaranteed assertion time for CPURST in this mode.
$\overline{\text{EPMI}}$	I	External Power Management Input: This signal is used to force the assertion of an $\overline{\text{SMI}}$ by an external source (e.g., a power-down switch).

Keyboard Interface Signals

Name	I/O	Description
KBDATA	I/O	Keyboard Data: This is the serial data line to/from the keyboard connector.
$\overline{\text{KBCS}}$/KYLCK	I/O	Keyboard Chip Select/Key Lock Input: This signal is the key lock input to lock the keyboard (for system security) if the internal keyboard controller is used. If an external keyboard controller is desired, this is the keyboard controller chip select.
MSDATA/KBDCLK	I/O	Mouse Data/External Keyboard Controller Clock Output: If the internal keyboard controller is used, this is the serial mouse data line to/from the keyboard connector. If an external keyboard controller is desired, this signal drives the state machine clock input (normally ATCLK) of the external keyboard controller.

IDE Interface Signals

Name	I/O	Description
$\overline{\text{IDEIOR}}$	O	IDE I/O Read: This signal directly drives the $\overline{\text{IDEIOR}}$ signal on the IDE connector.
$\overline{\text{IDEIOW}}$	O	IDE I/O Write: This signal directly drives the $\overline{\text{IDEIOW}}$ signal on the IDE connector.
$\overline{\text{IDEIOCS16}}$	I	IDE I/O Chip Select 16: This signal directly drives the $\overline{\text{IDEIOCS16}}$ signal on the IDE connector. Determines whether the IDE transaction is 8 bits or 16 bits wide.
BLKIDE	O	Block IDE: This signal may be used in conjunction with an external Quad OR gate to block the IDE chip selects to the IDE drives during ISA transfers. This prevents older IDE devices from hanging if multiple IDE chip selects are asserted. Newer drives will not have a problem because IDE chip selects will only be sampled when an IDE COMMAND signal ($\overline{\text{IDEIOW}}$ or $\overline{\text{IDEIOR}}$) is asserted.

USB Interface Signals

Name	I/O	Description
USB_CLK	I	USB 48MHz Clock Input
USB_D1+, USB_D1–	I/O	USB port 1: This signal pair comprises the differential signal for USB port 1.
USB_D2+, USB_D2–	I/O	USB port 2: this signal pair comprises the differential signal for USB port 2.

Miscellaneous Signals

Name	I/O	Description
$\overline{\text{ROMCS}}$ /ROMMODE	I/O	System ROM Chip Select: This signal is used to enable the on-board BIOS ROM. At power-up, this signal acts as ROMMODE. ROMMODE is used in conjunction with ROMS0 (GTA20) and ROMS1 ($\overline{\text{IGNNE}}$) to implement boot-block Flash recovery straps. The strapping is defined as follows: ROMMODE / ROMS0 / ROMS1 / Result 1 / X / X / No ROM address bits are inverted. (EPROM or boot-block recovery mode). 0 / 1 / 1 / ROM address bit 16 is inverted (Normal operation w/ 16Kx8 boot-block Flash) 0 / 1 / 0 / ROM address bit 17 is inverted (Normal operation with 32Kx8 boot-block Flash) 0 / 0 / 1 / ROM address bit 18 is inverted (Normal operation with 64Kx8 boot-block Flash) 0 / 0 / 0 / ROM address bit 15 is inverted (Normal operation with 8Kx8 boot-block Flash)
VBATT	I	Battery Backup Power: This pin should be connected directly to a battery-driven power source. Used to retain and maintain RTC functionality when the main power supply is disconnected.
+5V	I	V_{CC}: These are the +5V power supply pins for the device. They should be maintained within the proper operating limits.
+3.3V	I	3.3-Volt V_{CC}: This is the +3.3V power supply pin for the device. It should be maintained within the proper operating limits.
GND	I	GROUND: These are the 0V power supply pins for the device. They should be maintained within the proper operating limits.

hyperCache Memory and I/O Map

The hyperCache memory and I/O mapping diagrams are shown in this section. The detailed description of I/O registers shown in the I/O map can be found in the next sections.

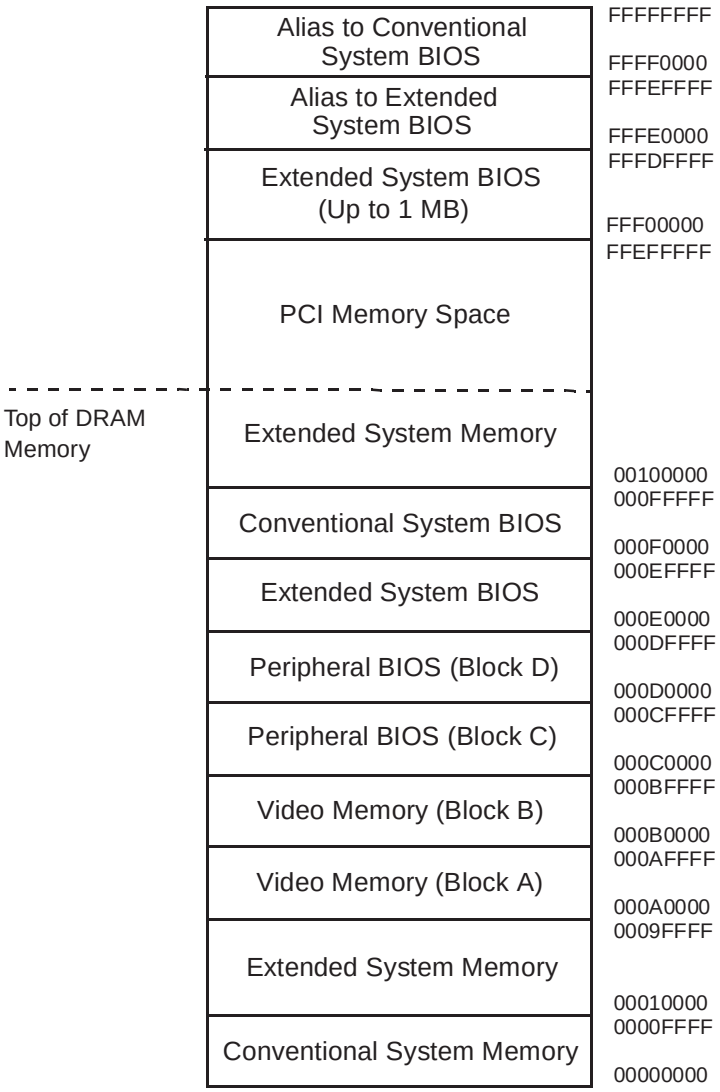


Figure 4. hyperCache Memory Map Diagram

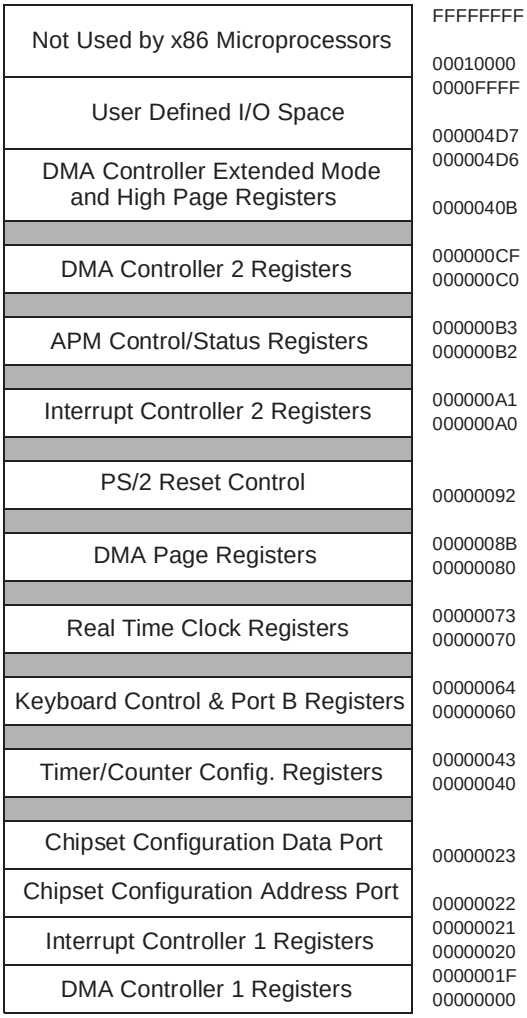


Figure 5. hyperCache CY82C693U I/O Map Diagram

CY82C693U Control Registers

The control registers for the CY82C693U are defined in this section. The registers can be accessed through I/O Ports 22H and 23H (PCI I/O Reads or Writes to address 22H and 23H). To access each register, the user must first write the index number of the register into Port 22, which forces the internal decoding logic to point to the selected register. Data can be accessed by then reading/writing to/from Port 23.

Register 1: Peripheral Configuration Register #1 (Read/Write) — Index=01H

Bit	Function	Default
7:6	I/O Wait State Control: 00: 1 Wait State 01: 2 Wait States 10: 3 Wait States 11: 4 Wait States This register field is used to aid in ISA compatibility. ISA cards that cannot respond fast enough may require wait states to be inserted into each ISA cycle. Wait states are inserted by the deassertion of the IOCHRDY signal. IOCHRDY will remain deasserted for the number of wait states (AT clock cycles) programmed into this register field.	11
5:4	16-bit DMA Wait State Control 00: 1 Wait State 01: 2 Wait States 10: 3 Wait States 11: 4 Wait States This bit field allows wait state control on 16-bit accesses. Wait states are inserted by the deassertion of the IOCHRDY signal. IOCHRDY will remain deasserted for the number of wait states (AT clock cycles) programmed into this register field.	00
3:2	8-bit DMA Wait State Control 00: 1 Wait State 01: 2 Wait States 10: 3 Wait States 11: 4 Wait States This bit field allows wait state control on 8-bit accesses. Wait states are inserted by the deassertion of the IOCHRDY signal. IOCHRDY will remain deasserted for the number of wait states (AT clock cycles) programmed into this register field.	00
1	MEMR Leading Edge Delay Control: 0: 1 DMA clock delay 1: No delay This bit allows the assertion of the $\overline{\text{MEMR}}$ signal to be delayed during DMA transactions. Normally $\overline{\text{MEMR}}$ is asserted one DMA clock cycle later than $\overline{\text{IOR}}$ during DMA. This delay may be removed so that $\overline{\text{MEMR}}$ and $\overline{\text{IOR}}$ are asserted at the same time.	0
0	DMA Controller Clock Speed Control: 0: DMA clock is ATCLK divided by two 1: DMA clock is ATCLK This bit allows the clock that controls the DMA controllers to be sped by running directly off of the ATCLK. Normally, the DMA clock is the AT clock divided by two. When this bit changes, internal synchronization logic prevents short clock pulses.	0

Register 2: Peripheral Configuration Register #2 (Read/Write) – Index=02H

Bit	Function	Default
7:3	Reserved	000
2	Peripheral Controller Test Mode Control: 0: Normal Operation 1: Put Peripheral Controller Into Test Mode	0
1	Interrupt Request Configuration Control: 0: Normal Operation 1: Enable Interrupt Request Level/Edge Control When this bit is set to 1, interrupt request inputs can be programmed (on an individual basis) to be Edge-triggered (non-sharable) or Level-sensitive (sharable). The selection is made in registers 3 and 4.	0
0	DMA Extended Mode Control 0: Enable DMA Extended Mode 1: Disable DMA Extended Mode	0

Register 3: Interrupt Request Level/Edge Control Register #1 (Read/Write) – Index=03H

Bit	Function	Default
7	IRQ7 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
6	IRQ6 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
5	IRQ5 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
4	IRQ4 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
3	IRQ3 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
2:0	Reserved	000

Register 4: Interrupt Request Level/Edge Control Register #2 (Read/Write) – Index=04H

Bit	Function	Default
7	IRQ15 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
6	IRQ14 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
5	Reserved, Must be 0	0
4	IRQ12 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
3	IRQ11 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
2	IRQ10 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
1	IRQ9 Edge-Triggered/Level-Sensitive Selector: 0: Edge-Triggered 1: Level-Sensitive	0
0	Reserved, Must be 0	0

Register 5: Real-Time-Clock Configuration Register (Read/Write) – Index=05H

Bit	Function	Default
7:3	Reserved	00000
2:0	Value programmed into this register is used to fine-tune the 32-kHz Oscillator for greater RTC accuracy.	000

Write-Only Shadow Registers

The following registers contain shadowed values for the internal write-only registers. The shadowed values may be used to read, mask-off, and then modify certain fields within the write-only registers.

Register 80: DMA1 Write Request Shadow Register (Read/Write) – Index=80H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA1's DMA request register. (DMA Register 9)	00000000

Register 81: DMA1 Write Single Mask Bit Shadow Register (Read/Write) – Index=81H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA1 write single mask bit register. (DMA Register 10)	00000000

Register 82: DMA1 Write Mode Shadow Register (Read/Write) – Index=82H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA1 write mode register. (DMA Register 11)	00000000

Register 83: DMA1 Clear Byte Pointer Shadow Register (Read/Write) – Index=83H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA1 Address Space Expansion Flip-Flop Control Register. (DMA Register 12)	00000000

Register 84: DMA1 Master Clear Shadow Register (Read/Write) – Index=84H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA1 master clear register. (DMA Register 13)	00000000

Register 85: DMA1 Clear Mask Shadow Register (Read/Write) – Index=85H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA1 mask clear register. (DMA Register 14)	00000000

Register 86: Timer Counter 1 Command Mode Shadow Register (Read/Write) – Index=86H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into timer counter 1 command mode register. (Timer/Counter Register 0)	00000000

Register 87: CMOS Battery-Backed RAM Address and NMI Mask Registers Shadow Register (Read/Write) – Index=87H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into the NMI mask register and the last accessed address to the CMOS RAM.	00000000

Register 88: DMA2 Write Request Shadow Register (Read/Write) – Index=88H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA2's write request register. (DMA Register 25)	00000000

Register 89: DMA2 Write Single Mask Bit Shadow Register (Read/Write) – Index=89H

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA2's write single mask bit register. (DMA Register 26)	00000000

Register 8A: DMA2 Write Mode Shadow Register (Read/Write) – Index=8AH

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA2's write mode register. (DMA Register 27)	00000000

Register 8B: DMA2 Clear Byte Pointer Shadow Register (Read/Write) – Index=8BH

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA2's Address Space Expansion Flip-Flop Control register. (DMA Register 28)	00000000

Register 8C: DMA2 Mask Clear Shadow Register (Read/Write) – Index=8CH

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA2's mask clear register. (DMA Register 29)	00000000

Register 8D: DMA2 Clear Mask Shadow Register (Read/Write) – Index=8DH

Bit	Function	Default
7:0	This register contains the shadowed value that was written into DMA2's clear register. (DMA Register 30)	00000000

Register 8E: Coprocessor Error Shadow Register (Read/Write) – Index=8EH

Bit	Function	Default
7:0	This register contains the shadowed value that was written into the coprocessor error register.	00000000

Register 8F: Extended CMOS RAM address Shadow Register (Read/Write) – Index=8FH

Bit	Function	Default
7:0	This register contains the shadowed value that was written into the extended CMOS RAM address register.	00000000

General Purpose I/O Registers

The following registers control the operation of GPIO.

Register 90: General Purpose I/O Control Register A (Read/Write) – Index=90H

Bit	Function	Default
7	Internal Keyboard Controller GPIO Port Output Control: 0: Internal Keyboard Controller Ports not available on outputs of the CY82C693U. 1: Internal Keyboard Controller Ports available on outputs of the CY82C693U.	0
6	Reserved	0
5	GPIO5 Control: This bit value will be driven onto the GPIO5 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO5 pin if it is programmed as an input.	0
4	GPIO4 Control: This bit value will be driven onto the GPIO4 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO4 pin if it is programmed as an input.	0
3	GPIO3 Control: This bit value will be driven onto the GPIO3 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO3 pin if it is programmed as an input.	0
2	GPIO2 Control: This bit value will be driven onto the GPIO2 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO2 pin if it is programmed as an input.	0
1	GPIO1 Control: This bit value will be driven onto the GPIO1 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO1 pin if it is programmed as an input.	0
0	GPIO0 Control: This bit value will be driven onto the GPIO0 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO0 pin if it is programmed as an input.	0

Register 91: General Purpose I/O Input/Output Control Register A (Read/Write) – Index=91H

Bit	Function	Default
7:6	Reserved	00
5	GPIO5 Control: 0: Input 1: Output	0
4	GPIO4 Control: 0: Input 1: Output	0
3	GPIO3 Control: 0: Input 1: Output	0
2	GPIO2 Control: 0: Input 1: Output	0
1	GPIO1 Control: 0: Input 1: Output	0
0	GPIO0 Control: 0: Input 1: Output	0

Register 92: General Purpose I/O Control Register B (Read/Write) – Index=92H

Bit	Function	Default
7	DMA Request Mode: 0: Normal. DMA request pins cannot be used for GPIO. 1: Multiplexed. If external multiplexer is used, several of the DMA Request signals can be used for GPIO.	0
6	DMA Acknowledge Mode: 0: Normal. DMA acknowledge pins cannot be used for GPIO. 1: Encoded. If external priority decoder is used, several of the DMA Acknowledge signals can be used for GPIO.	0
5	GPIO11 Control: This bit value will be driven onto the GPIO11 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO11 pin if it is programmed as an input.	0
4	GPIO10 Control: This bit value will be driven onto the GPIO10 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO10 pin if it is programmed as an input.	0
3	GPIO9 Control: This bit value will be driven onto the GPIO9 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO9 pin if it is programmed as an input.	0
2	GPIO8 Control: This bit value will be driven onto the GPIO8 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO8 pin if it is programmed as an input.	0
1	GPIO7 Control: This bit value will be driven onto the GPIO7 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO7 pin if it is programmed as an input.	0
0	GPIO6 Control: This bit value will be driven onto the GPIO6 pin if it is programmed as an output. This bit will contain the value driven onto the GPIO6 pin if it is programmed as an input.	0

Register 93: General Purpose I/O Input/Output Control Register B (Read/Write) – Index=93H

Bit	Function	Default
7:6	Reserved	00
5	GPIO11 Control: 0: Input 1: Output	0
4	GPIO10 Control: 0: Input 1: Output	0
3	GPIO9 Control: 0: Input 1: Output	0
2	GPIO8 Control: 0: Input 1: Output	0
1	GPIO7 Control: 0: Input 1: Output	0
0	GPIO6 Control: 0: Input 1: Output	0

Power Management Control Registers

The following registers control the operation of the power management logic within the CY82C693U.

Register 40: Standby Timer Event Detection Control (Read/Write) – Index=40H

Bit	Function	Default
7	PCI Master Request Detection Control: 0: Disable PCI Master Request Detection 1: Enable PCI Master Request Detection	0
6	Video Memory Access (Blocks A, B, or 3b0H–3dfH) Detection Control: 0: Disable Video Memory Access Detection 1: Enable Video Memory Access Detection	0
5	Parallel Port Access (278H–27fH, or 378H–37fH) Detection Control: 0: Disable Parallel Port. Access Detection 1: Enable Parallel Port Access Detection	0
4	COM Port 2/4 Access (2f8H–2ffH, or 2e8H–2efH) Detection Control: 0: Disable COM Port 2/4 Access Detection 1: Enable COM Port 2/4 Access Detection	0
3	COM Port 1/3 Access (3f8H–3ffH, or 3e8H–3efH) Detection Control: 0: Disable COM Port 1/3 Access Detection 1: Enable COM Port 1/3 Access Detection	0
2	Hard Disk Access (170H–177H, or 1f0H–1f7H) Detection Control: 0: Disable Hard Disk Access Detection 1: Enable Hard Disk Access Detection	0
1	Floppy Disk Access (3f5H) Detection Control: 0: Disable Floppy Disk Access Detection 1: Enable Floppy Disk Access Detection	0
0	Keyboard Controller Access (60H or 64H) Detection Control: 0: Disable Keyboard Controller Access Detection 1: Enable Keyboard Controller Access Detection	0

Register 41: Standby Timer Interrupt Request Detection Control #1 (Read/Write) – Index=41H

Bit	Function	Default
7	IRQ7 Monitor Control: 0: Disable IRQ7 Monitoring 1: Enable IRQ7 Monitoring	0
6	IRQ6 Monitor Control: 0: Disable IRQ6 Monitoring 1: Enable IRQ6 Monitoring	0
5	IRQ5 Monitor Control: 0: Disable IRQ5 Monitoring 1: Enable IRQ5 Monitoring	0
4	IRQ4 Monitor Control: 0: Disable IRQ4 Monitoring 1: Enable IRQ4 Monitoring	0
3	IRQ3 Monitor Control: 0: Disable IRQ3 Monitoring 1: Enable IRQ3 Monitoring	0
2	INTR Monitor Control (Any Unmasked Interrupt Request): 0: Disable INTR Monitoring 1: Enable INTR Monitoring	0
1	IRQ1 Monitor Control: 0: Disable IRQ1 Monitoring 1: Enable IRQ1 Monitoring	0
0	IRQ0 Monitor Control: 0: Disable IRQ0 Monitoring 1: Enable IRQ0 Monitoring	0

Register 42: Standby Timer Interrupt Request Detection Control #2 (Read/Write) – Index=42H

Bit	Function	Default
7	IRQ15 Monitor Control: 0: Disable IRQ15 Monitoring 1: Enable IRQ15 Monitoring	0
6	IRQ14 Monitor Control: 0: Disable IRQ14 Monitoring 1: Enable IRQ14 Monitoring	0
5	IRQ13 Monitor Control: 0: Disable IRQ13 Monitoring 1: Enable IRQ13 Monitoring	0
4	IRQ12 Monitor Control: 0: Disable IRQ12 Monitoring 1: Enable IRQ12 Monitoring	0
3	IRQ11 Monitor Control: 0: Disable IRQ11 Monitoring 1: Enable IRQ11 Monitoring	0
2	IRQ10 Monitor Control: 0: Disable IRQ10 Monitoring 1: Enable IRQ10 Monitoring	0
1	IRQ9 Monitor Control: 0: Disable IRQ9 Monitoring 1: Enable IRQ9 Monitoring	0
0	IRQ8 Monitor Control: 0: Disable IRQ8 Monitoring 1: Enable IRQ8 Monitoring	0

Register 43: Standby Timer DMA Request Detection Control #1 (Read/Write) – Index=43H

Bit	Function	Default
7	DREQ7 Monitor Control: 0: Disable DREQ7 Monitoring 1: Enable DREQ7 Monitoring	0
6	DREQ6 Monitor Control: 0: Disable DREQ6 Monitoring 1: Enable DREQ6 Monitoring	0
5	DREQ5 Monitor Control: 0: Disable DREQ5 Monitoring 1: Enable DREQ5 Monitoring	0
4	ISA DMA/MASTER Monitor Control (Any DREQ): 0: Disable ISA DMA/MASTER Monitoring 1: Enable ISA DMA/MASTER Monitoring	0
3	DREQ3 Monitor Control: 0: Disable DREQ3 Monitoring 1: Enable DREQ3 Monitoring	0
2	DREQ2 Monitor Control: 0: Disable DREQ2 Monitoring 1: Enable DREQ2 Monitoring	0
1	DREQ1 Monitor Control: 0: Disable DREQ1 Monitoring 1: Enable DREQ1 Monitoring	0
0	DREQ0 Monitor Control: 0: Disable DREQ0 Monitoring 1: Enable DREQ0 Monitoring	0

Register 44: Suspend Timer Event Detection Control (Read/Write) – Index=44H

Bit	Function	Default
7	PCI Master Request Detection Control: 0: Disable PCI Master Request Detection 1: Enable PCI Master Request Detection	0
6	Video Memory Access (Blocks A, B, or 3b0H–3dfH) Detection Control: 0: Disable Video Memory Access Detection 1: Enable Video Memory Access Detection	0
5	Parallel Port Access (278H–27fH, or 378H–37fH) Detection Control: 0: Disable Parallel Port Access Detection 1: Enable Parallel Port Access Detection	0
4	COM Port 2/4 Access (2f8H–2ffH, or 2e8H–2efH) Detection Control: 0: Disable COM Port. 2/4 Access Detection 1: Enable COM Port 2/4 Access Detection	0
3	COM Port 1/3 Access (3f8H–3ffH, or 3e8H–3efH) Detection Control: 0: Disable COM Port 1/3 Access Detection 1: Enable COM Port 1/3 Access Detection	0
2	Hard Disk Access (170H–177H, or 1f0H–1f7H) Detection Control: 0: Disable Hard Disk Access Detection 1: Enable Hard Disk Access Detection	0
1	Floppy Disk Access (3f5H) Detection Control: 0: Disable Floppy Disk Access Detection 1: Enable Floppy Disk Access Detection	0
0	Keyboard Controller Access (60H or 64H) Detection Control: 0: Disable Keyboard Controller Access Detection 1: Enable Keyboard Controller Access Detection	0

Register 45: Suspend Timer Interrupt Request Detection Control #1 (Read/Write) – Index=45H

Bit	Function	Default
7	IRQ7 Monitor Control: 0: Disable IRQ7 Monitoring 1: Enable IRQ7 Monitoring	0
6	IRQ6 Monitor Control: 0: Disable IRQ6 Monitoring 1: Enable IRQ6 Monitoring	0
5	IRQ5 Monitor Control: 0: Disable IRQ5 Monitoring 1: Enable IRQ5 Monitoring	0
4	IRQ4 Monitor Control: 0: Disable IRQ4 Monitoring 1: Enable IRQ4 Monitoring	0
3	IRQ3 Monitor Control: 0: Disable IRQ3 Monitoring 1: Enable IRQ3 Monitoring	0
2	INTR Monitor Control (Any Unmasked Interrupt Request): 0: Disable INTR Monitoring 1: Enable INTR Monitoring	0
1	IRQ1 Monitor Control: 0: Disable IRQ1 Monitoring 1: Enable IRQ1 Monitoring	0
0	IRQ0 Monitor Control: 0: Disable IRQ0 Monitoring 1: Enable IRQ0 Monitoring	0

Register 46: Suspend Timer Interrupt Request Detection Control #2 (Read/Write) – Index=46H

Bit	Function	Default
7	IRQ15 Monitor Control: 0: Disable IRQ15 Monitoring 1: Enable IRQ15 Monitoring	0
6	IRQ14 Monitor Control: 0: Disable IRQ14 Monitoring 1: Enable IRQ14 Monitoring	0
5	IRQ13 Monitor Control: 0: Disable IRQ13 Monitoring 1: Enable IRQ13 Monitoring	0
4	IRQ12 Monitor Control: 0: Disable IRQ12 Monitoring 1: Enable IRQ12 Monitoring	0
3	IRQ11 Monitor Control: 0: Disable IRQ11 Monitoring 1: Enable IRQ11 Monitoring	0
2	IRQ10 Monitor Control: 0: Disable IRQ10 Monitoring 1: Enable IRQ10 Monitoring	0
1	IRQ9 Monitor Control: 0: Disable IRQ9 Monitoring 1: Enable IRQ9 Monitoring	0
0	IRQ8 Monitor Control: 0: Disable IRQ8 Monitoring 1: Enable IRQ8 Monitoring	0

Register 47: Suspend Timer DMA Request Detection Control #1 (Read/Write) – Index=47H

Bit	Function	Default
7	DREQ7 Monitor Control: 0: Disable DREQ7 Monitoring 1: Enable DREQ7 Monitoring	0
6	DREQ6 Monitor Control: 0: Disable DREQ6 Monitoring 1: Enable DREQ6 Monitoring	0
5	DREQ5 Monitor Control: 0: Disable DREQ5 Monitoring 1: Enable DREQ5 Monitoring	0
4	ISA DMA/MASTER Monitor Control (Any DREQ): 0: Disable ISA DMA/MASTER Monitoring 1: Enable ISA DMA/MASTER Monitoring	0
3	DREQ3 Monitor Control: 0: Disable DREQ3 Monitoring 1: Enable DREQ3 Monitoring	0
2	DREQ2 Monitor Control: 0: Disable DREQ2 Monitoring 1: Enable DREQ2 Monitoring	0
1	DREQ1 Monitor Control: 0: Disable DREQ1 Monitoring 1: Enable DREQ1 Monitoring	0
0	DREQ0 Monitor Control: 0: Disable DREQ0 Monitoring 1: Enable DREQ0 Monitoring	0

Register 48: User Timer 1 Event Detection Control (Read/Write) – Index=48H

Bit	Function	Default
7	PCI Master Request Detection Control: 0: Disable PCI Master Request Detection 1: Enable PCI Master Request Detection	0
6	Video Memory Access (Blocks A, B, or 3b0H–3dfH) Detection Control: 0: Disable Video Memory Access Detection 1: Enable Video Memory Access Detection	0
5	Parallel Port Access (278H–27fH, or 378H–37fH) Detection Control: 0: Disable Parallel Port. Access Detection 1: Enable Parallel Port Access Detection	0
4	COM Port 2/4 Access (2f8H–2ffH, or 2e8H–2efH) Detection Control: 0: Disable COM Port 2/4 Access Detection 1: Enable COM Port 2/4 Access Detection	0
3	COM Port 1/3 Access (3f8H–3ffH, or 3e8H–3efH) Detection Control: 0: Disable COM Port 1/3 Access Detection 1: Enable COM Port 1/3 Access Detection	0
2	Hard Disk Access (170H–177H, or 1f0H–1f7H) Detection Control: 0: Disable Hard Disk Access Detection 1: Enable Hard Disk Access Detection	0
1	Floppy Disk Access (3f5H) Detection Control: 0: Disable Floppy Disk Access Detection 1: Enable Floppy Disk Access Detection	0
0	Keyboard Controller Access (60H or 64H) Detection Control: 0: Disable Keyboard Controller Access Detection 1: Enable Keyboard Controller Access Detection	0

Register 49: User Timer 1 Interrupt Request Detection Control #1 (Read/Write) – Index=49H

Bit	Function	Default
7	IRQ7 Monitor Control: 0: Disable IRQ7 Monitoring 1: Enable IRQ7 Monitoring	0
6	IRQ6 Monitor Control: 0: Disable IRQ6 Monitoring 1: Enable IRQ6 Monitoring	0
5	IRQ5 Monitor Control: 0: Disable IRQ5 Monitoring 1: Enable IRQ5 Monitoring	0
4	IRQ4 Monitor Control: 0: Disable IRQ4 Monitoring 1: Enable IRQ4 Monitoring	0
3	IRQ3 Monitor Control: 0: Disable IRQ3 Monitoring 1: Enable IRQ3 Monitoring	0
2	INTR Monitor Control (Any Unmasked Interrupt Request): 0: Disable INTR Monitoring 1: Enable INTR Monitoring	0
1	IRQ1 Monitor Control: 0: Disable IRQ1 Monitoring 1: Enable IRQ1 Monitoring	0
0	IRQ0 Monitor Control: 0: Disable IRQ0 Monitoring 1: Enable IRQ0 Monitoring	0

Register 4A: User Timer 1 Interrupt Request Detection Control #2 (Read/Write) – Index=4AH

Bit	Function	Default
7	IRQ15 Monitor Control: 0: Disable IRQ15 Monitoring 1: Enable IRQ15 Monitoring	0
6	IRQ14 Monitor Control: 0: Disable IRQ14 Monitoring 1: Enable IRQ14 Monitoring	0
5	IRQ13 Monitor Control: 0: Disable IRQ13 Monitoring 1: Enable IRQ13 Monitoring	0
4	IRQ12 Monitor Control: 0: Disable IRQ12 Monitoring 1: Enable IRQ12 Monitoring	0
3	IRQ11 Monitor Control: 0: Disable IRQ11 Monitoring 1: Enable IRQ11 Monitoring	0
2	IRQ10 Monitor Control: 0: Disable IRQ10 Monitoring 1: Enable IRQ10 Monitoring	0
1	IRQ9 Monitor Control: 0: Disable IRQ9 Monitoring 1: Enable IRQ9 Monitoring	0
0	IRQ8 Monitor Control: 0: Disable IRQ8 Monitoring 1: Enable IRQ8 Monitoring	0

Register 4B: User Timer 1 DMA Request Detection Control #1 (Read/Write) – Index=4BH

Bit	Function	Default
7	DREQ7 Monitor Control: 0: Disable DREQ7 Monitoring 1: Enable DREQ7 Monitoring	0
6	DREQ6 Monitor Control: 0: Disable DREQ6 Monitoring 1: Enable DREQ6 Monitoring	0
5	DREQ5 Monitor Control: 0: Disable DREQ5 Monitoring 1: Enable DREQ5 Monitoring	0
4	ISA DMA/MASTER Monitor Control (Any DREQ): 0: Disable ISA DMA/MASTER Monitoring 1: Enable ISA DMA/MASTER Monitoring	0
3	DREQ3 Monitor Control: 0: Disable DREQ3 Monitoring 1: Enable DREQ3 Monitoring	0
2	DREQ2 Monitor Control: 0: Disable DREQ2 Monitoring 1: Enable DREQ2 Monitoring	0
1	DREQ1 Monitor Control: 0: Disable DREQ1 Monitoring 1: Enable DREQ1 Monitoring	0
0	DREQ0 Monitor Control: 0: Disable DREQ0 Monitoring 1: Enable DREQ0 Monitoring	0

Register 4C: Throttle Timer Event Detection Control (Read/Write) – Index=4CH

Bit	Function	Default
7	PCI Master Request Detection Control: 0: Disable PCI Master Request Detection 1: Enable PCI Master Request Detection	0
6	Video Memory Access (Blocks A, B, or 3b0H–3dfH) Detection Control: 0: Disable Video Memory Access Detection 1: Enable Video Memory Access Detection	0
5	Parallel Port Access (278H–27fH, or 378H–37fH) Detection Control: 0: Disable Parallel Port Access Detection 1: Enable Parallel Port Access Detection	0
4	COM Port 2/4 Access (2f8H–2ffH, or 2e8H–2efH) Detection Control: 0: Disable COM Port 2/4 Access Detection 1: Enable COM Port 2/4 Access Detection	0
3	COM Port 1/3 Access (3f8H–3ffH, or 3e8H–3efH) Detection Control: 0: Disable COM Port. 1/3 Access Detection 1: Enable COM Port 1/3 Access Detection	0
2	Hard Disk Access (170H–177H, or 1f0H–1f7H) Detection Control: 0: Disable Hard Disk Access Detection 1: Enable Hard Disk Access Detection	0
1	Floppy Disk Access (3f5H) Detection Control: 0: Disable Floppy Disk Access Detection 1: Enable Floppy Disk Access Detection	0
0	Keyboard Controller Access (60H or 64H) Detection Control: 0: Disable Keyboard Controller Access Detection 1: Enable Keyboard Controller Access Detection	0

Register 4D: Throttle Timer Interrupt Request Detection Control #1 (Read/Write) – Index=4DH

Bit	Function	Default
7	IRQ7 Monitor Control: 0: Disable IRQ7 Monitoring 1: Enable IRQ7 Monitoring	0
6	IRQ6 Monitor Control: 0: Disable IRQ6 Monitoring 1: Enable IRQ6 Monitoring	0
5	IRQ5 Monitor Control: 0: Disable IRQ5 Monitoring 1: Enable IRQ5 Monitoring	0
4	IRQ4 Monitor Control: 0: Disable IRQ4 Monitoring 1: Enable IRQ4 Monitoring	0
3	IRQ3 Monitor Control: 0: Disable IRQ3 Monitoring 1: Enable IRQ3 Monitoring	0
2	INTR Monitor Control (Any Unmasked Interrupt Request): 0: Disable INTR Monitoring 1: Enable INTR Monitoring	0
1	IRQ1 Monitor Control: 0: Disable IRQ1 Monitoring 1: Enable IRQ1 Monitoring	0
0	IRQ0 Monitor Control: 0: Disable IRQ0 Monitoring 1: Enable IRQ0 Monitoring	0

Register 4E: Throttle Timer Interrupt Request Detection Control #2 (Read/Write) – Index=4EH

Bit	Function	Default
7	IRQ15 Monitor Control: 0: Disable IRQ15 Monitoring 1: Enable IRQ15 Monitoring	0
6	IRQ14 Monitor Control: 0: Disable IRQ14 Monitoring 1: Enable IRQ14 Monitoring	0
5	IRQ13 Monitor Control: 0: Disable IRQ13 Monitoring 1: Enable IRQ13 Monitoring	0
4	IRQ12 Monitor Control: 0: Disable IRQ12 Monitoring 1: Enable IRQ12 Monitoring	0
3	IRQ11 Monitor Control: 0: Disable IRQ11 Monitoring 1: Enable IRQ11 Monitoring	0
2	IRQ10 Monitor Control: 0: Disable IRQ10 Monitoring 1: Enable IRQ10 Monitoring	0
1	IRQ9 Monitor Control: 0: Disable IRQ9 Monitoring 1: Enable IRQ9 Monitoring	0
0	IRQ8 Monitor Control: 0: Disable IRQ8 Monitoring 1: Enable IRQ8 Monitoring	0

Register 4F: Throttle Timer DMA Request Detection Control #1 (Read/Write) – Index=4FH

Bit	Function	Default
7	DREQ7 Monitor Control: 0: Disable DREQ7 Monitoring 1: Enable DREQ7 Monitoring	0
6	DREQ6 Monitor Control: 0: Disable DREQ6 Monitoring 1: Enable DREQ6 Monitoring	0
5	DREQ5 Monitor Control: 0: Disable DREQ5 Monitoring 1: Enable DREQ5 Monitoring	0
4	ISA DMA/MASTER Monitor Control (Any DREQ): 0: Disable ISA DMA/MASTER Monitoring 1: Enable ISA DMA/MASTER Monitoring	0
3	DREQ3 Monitor Control: 0: Disable DREQ3 Monitoring 1: Enable DREQ3 Monitoring	0
2	DREQ2 Monitor Control: 0: Disable DREQ2 Monitoring 1: Enable DREQ2 Monitoring	0
1	DREQ1 Monitor Control: 0: Disable DREQ1 Monitoring 1: Enable DREQ1 Monitoring	0
0	DREQ0 Monitor Control: 0: Disable DREQ0 Monitoring 1: Enable DREQ0 Monitoring	0

Register 50: Non-motherboard Memory Address Range Decode for Event Detection Register x#1 (Read/Write) – Index=50H

Bit	Function	Default
7:0	Non-motherboard memory address decode AD[31:24]	00000000

Register 51: Non-motherboard Memory Address Range Decode for Event Detection Register #2 (Read/Write) – Index=51H

Bit	Function	Default
7:0	Non-motherboard memory address decode AD[23:16]	00000000

Register 52: Non-motherboard Memory Address Mask for Event Detection Register #1 (Read/Write) – Index=52H

Bit	Function	Default
7:0	Mask bits for non-motherboard memory address decode for event detection AD[31:24]	00000000

Register 53: Non-motherboard Memory Address Mask for Event Detection Register #2 (Read/Write) – Index=53H

Bit	Function	Default
7:0	Mask bits for non-motherboard memory address decode for event detection AD[23:16]	00000000

Register 54: Programmable I/O Trap 1 Address Range Register #1 (Read/Write) – Index=54H

Bit	Function	Default
7:0	I/O address AD[7:0] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 55: Programmable I/O Trap 1 Address Range Register #2 (Read/Write) – Index=55H

Bit	Function	Default
7:0	I/O address AD[15:8] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 56: Programmable I/O Trap 1 Address Range Register #3 (Read/Write) – Index=56H

Bit	Function	Default
7:0	I/O address AD[23:16] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 57: Programmable I/O Trap 1 Address Range Register #4 (Read/Write) – Index=57H

Bit	Function	Default
7:0	I/O address AD[31:24] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 58: Programmable I/O Trap 2 Address Range Register #1 (Read/Write) – Index=58H

Bit	Function	Default
7:0	Secondary I/O address AD[7:0] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 59: Programmable I/O Trap 2 Address Range Register #2 (Read/Write) – Index=59H

Bit	Function	Default
7:0	Secondary I/O address AD[15:8] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 5A: Programmable I/O Trap 2 Address Range Register #3 (Read/Write) – Index=5AH

Bit	Function	Default
7:0	Secondary I/O address AD[23:16] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 5B: Programmable I/O Trap 2 Address Range Register #4 (Read/Write) – Index=5BH

Bit	Function	Default
7:0	Secondary I/O address AD[31:24] which will automatically generate an SMI. The I/O Trap can be used in conjunction with the I/O restart feature of the Pentium.	00000000

Register 5C: Programmable I/O Trap 1 Address Detection Control (Read/Write) – Index=5CH

Bit	Function	Default
7	AD[15:0] I/O Trap 1 Address Mask Control: 0: Disable AD[15:10] I/O Trap 1 Address Masking 1: Enable AD[15:10] I/O Trap 1 Address Masking	0
6	AD[31:16] I/O Trap 1 Address Mask Control: 0: Disable AD[31:16] I/O Trap 1 Address Masking 1: Enable AD[31:16] I/O Trap 1 Address Masking	0
5	Reserved	0
4:2	Mask bit selection for Programmable I/O Trap 1 Address: 000: No Bits Masked 001: Lowest Order Bit Masked 010: 2 Lowest Order Bits Masked 011: 3 Lowest Order Bits Masked 100: 4 Lowest Order Bits Masked 101: 5 Lowest Order Bits Masked 110: 6 Lowest Order Bits Masked 111: 7 Lowest Order Bits Masked	000
1	Standby Timer Non-motherboard memory access detection control: 0: Disable Non-motherboard memory access detection 1: Enable Non-motherboard memory access detection	0
0	Suspend Timer Non-motherboard memory access detection control: 0: Disable Non-motherboard memory access detection 1: Enable Non-motherboard memory access detection	0

Register 5D: Programmable I/O Trap 2 Address Detection Control (Read/Write) – Index=5DH

Bit	Function	Default
7	AD[15:0] I/O Trap 2 Address Mask Control: 0: Disable AD[15:10] I/O Trap 1 Address Masking 1: Enable AD[15:10] I/O Trap 1 Address Masking	0
6	AD[31:16] I/O Trap 2 Address Mask Control: 0: Disable AD[31:16] I/O Trap 1 Address Masking 1: Enable AD[31:16] I/O Trap 1 Address Masking	0
5	Reserved	0
4:2	Mask bit selection for Programmable I/O Trap 2 Address: 000: No Bits Masked 001: Lowest Order Bit Masked 010: 2 Lowest Order Bits Masked 011: 3 Lowest Order Bits Masked 100: 4 Lowest Order Bits Masked 101: 5 Lowest Order Bits Masked 110: 6 Lowest Order Bits Masked 111: 7 Lowest Order Bits Masked	000
1	User Timer 1 Non-motherboard memory access detection control: 0: Disable Non-motherboard memory access detection 1: Enable Non-motherboard memory access detection	0
0	Throttle Timer Non-motherboard memory access detection control: 0: Disable Non-motherboard memory access detection 1: Enable Non-motherboard memory access detection	0

Register 5E: I/O Trap 1 and 2 Monitoring Control (Read/Write) – Index=5EH

Bit	Function	Default
7	Standby Timer I/O Trap 1 Monitor Control: 0: Disable Monitoring of I/O Trap 1 1: Enable Monitoring of I/O Trap 1	0
6	Suspend Timer I/O Trap 1 Monitor Control: 0: Disable Monitoring of I/O Trap 1 1: Enable Monitoring of I/O Trap 1	0
5	User Timer 1 I/O Trap 1 Monitor Control: 0: Disable Monitoring of I/O Trap 1 1: Enable Monitoring of I/O Trap 1	0
4	Throttle Timer I/O Trap 1 Monitor Control: 0: Disable Monitoring of I/O Trap 1 1: Enable Monitoring of I/O Trap 1	0
3	Standby Timer I/O Trap 2 Monitor Control: 0: Disable Monitoring of I/O Trap 2 1: Enable Monitoring of I/O Trap 2	0
2	Suspend Timer I/O Trap 2 Monitor Control: 0: Disable Monitoring of I/O Trap 2 1: Enable Monitoring of I/O Trap 2	0
1	User Timer 1 I/O Trap 2 Monitor Control: 0: Disable Monitoring of I/O Trap 2 1: Enable Monitoring of I/O Trap 2	0
0	Throttle Timer I/O Trap 2 Monitor Control: 0: Disable Monitoring of I/O Trap 2 1: Enable Monitoring of I/O Trap 2	0

Register 5F: Standby and Suspend Timer Terminal Count Control Register (Read/Write) – Index=5FH

Bit	Function	Default
7:4	Standby Timer Terminal Count: 0000: 0.4 seconds 1000: 2 minutes 0001: 1 second 1001: 3.8 minutes 0010: 1.8 seconds 1010: 7.5 minutes 0011: 3.5 seconds 1011: 15 minutes 0100: 7 seconds 1100: 30 minutes 0101: 14 seconds 1101: 60 minutes 0110: 28 seconds 1110: 120 minutes 0111: 56 seconds 1111: 240 minutes	0000
3:0	Suspend Timer Terminal Count: 0000: 0.2 seconds 1000: 56 seconds 0001: 0.4 seconds 1001: 2 minutes 0010: 1 second 1010: 3.8 minutes 0011: 1.8 seconds 1011: 7.5 minutes 0100: 3.5 seconds 1100: 15 minutes 0101: 7 seconds 1101: 30 minutes 0110: 14 seconds 1110: 60 minutes 0111: 28 seconds 1111: 120 minutes	0000

Register 60: User Timer 1 and User Timer 2 Terminal Count Control Register (Read/Write) – Index=60H

Bit	Function	Default
7:4	User Timer 1 Terminal Count: 0000: 0.2 seconds 1000: 56 seconds 0001: 0.4 seconds 1001: 2 minutes 0010: 1 second 1010: 3.8 minutes 0011: 1.8 seconds 1011: 7.5 minutes 0100: 3.5 seconds 1100: 15 minutes 0101: 7 seconds 1101: 30 minutes 0110: 14 seconds 1110: 60 minutes 0111: 28 seconds 1111: 120 minutes	0000
3:0	User Timer 2 Terminal Count: 0000: 0.2 seconds 1000: 56 seconds 0001: 0.4 seconds 1001: 2 minutes 0010: 1 second 1010: 3.8 minutes 0011: 1.8 seconds 1011: 7.5 minutes 0100: 3.5 seconds 1100: 15 minutes 0101: 7 seconds 1101: 30 minutes 0110: 14 seconds 1110: 60 minutes 0111: 28 seconds 1111: 120 minutes	0000

Register 61: User Timer 3 Terminal Count Control Register (Read/Write) – Index=61H

Bit	Function	Default
7:4	User Timer 3 Terminal Count: 0000: 0.2 seconds 1000: 56 seconds 0001: 0.4 seconds 1001: 2 minutes 0010: 1 second 1010: 3.8 minutes 0011: 1.8 seconds 1011: 7.5 minutes 0100: 3.5 seconds 1100: 15 minutes 0101: 7 seconds 1101: 30 minutes 0110: 14 seconds 1110: 60 minutes 0111: 28 seconds 1111: 120 minutes	0000
3:2	SLOWCLK (PMWTCNT) to $\overline{\text{STOPCLK}}$ Transition Delay: 00: 430 microseconds 01: 7 milliseconds 10: 55 milliseconds 11: 1 second	00
1:0	Reserved	00

Register 62: Throttle Timer Terminal Count Control Register (Read/Write) – Index=62H

Bit	Function	Default
7:4	Throttle Timer Low Time:: 0000: 27 microseconds 1000: 7 milliseconds 0001: 53.7 microseconds 1001: 14 milliseconds 0010: 107 microseconds 1010: 28 milliseconds 0011: 215 microseconds 1011: 55 milliseconds 0100: 430 microseconds 1100: 0.1 seconds 0101: 860 microseconds 1101: 0.2 seconds 0110: 1.7 milliseconds 1110: 0.4 seconds 0111: 3.4 milliseconds 1111: 1 second	0000
3:0	Throttle Timer High Time:: 0000: 27 microseconds 1000: 7 milliseconds 0001: 53.7 microseconds 1001: 14 milliseconds 0010: 107 microseconds 1010: 28 milliseconds 0011: 215 microseconds 1011: 55 milliseconds 0100: 430 microseconds 1100: 0.1 seconds 0101: 860 microseconds 1101: 0.2 seconds 0110: 1.7 milliseconds 1110: 0.4 seconds 0111: 3.4 milliseconds 1111: 1 second	0000

Register 63: Power Management Control Register#1 (Read/Write) – Index=63H

Bit	Function	Default
7	Power Management Enable Control: 0: Disable Power Management 1: Enable Power Management	0
6	Hardware Controlled Power Management:: 0: Disable Hardware Power Management 1: Enable Hardware Power Management	0
5	Software Controlled Power Management:: 0: Disable Software Power Management 1: Enable Software Power Management	0
4	Reserved	0
3	User Timer 1 Control: 0: Disable User Timer 1 1: Enable User Timer 1	0
2	User Timer 2 Control: 0: Disable User Timer 2 1: Enable User Timer 2	0
1	User Timer 3 Control: 0: Disable User Timer 3 1: Enable User Timer 3	0
0	Suspend Timer Event Monitor Control: 0: Disable Monitoring of Events 1: Enable Monitoring of Events	0

Register 64: Power Management Control Register#2 (Read/Write) – Index=64H

Bit	Function	Default
7	Standby Timer Control: 0: Normal Operation 1: Reset Standby Timer (Value will automatically return to Normal Operation after Standby Timer is reset.)	0
6	Suspend Timer Control: 0: Enable Suspend Timer 1: Disable Suspend Timer Disabling and reenabling the Suspend Timer allows for additional power-down states. Once the Suspend Timer has expired, it can be disabled and reenabled. When the terminal count expires, $\overline{SMI}$ will be asserted again. Software can keep track of the number of times that the Suspend Timer has expired to create additional states.	0
5	Quick Power-down Control: 0: Disable Quick Power-down 1: Enable Quick Power-down	0
4	Video Access Event Control: 0: Video Access Event consists of accesses to blocks A, B, and I/O 3b0H–3dfH. 1: Video Access Event only consists of accesses to blocks A and B	0
3	Throttle Timer NMI monitor Control: 0: Disable monitoring of the NMI signal 1: Enable monitoring of the NMI signal	0
2	User Timer 1 NMI monitor Control: 0: Disable monitoring of the NMI signal 1: Enable monitoring of the NMI signal	0
1	Suspend Timer NMI monitor Control: 0: Disable monitoring of the NMI signal 1: Enable monitoring of the NMI signal	0
0	Standby Timer NMI monitor Control: 0: Disable monitoring of the NMI signal 1: Enable monitoring of the NMI signal	0

Register 65: Power Management Clock Control Register (Read/Write) – Index=65H

Bit	Function	Default
7:6	Reserved	00
5	Software $\overline{STOPCLK}$ Assertion Control: 0: Normal Operation 1: Force Assertion of $\overline{STOPCLK}$ $\overline{STOPCLK}$ will remain active after this register is written to one until this register is written to zero.	0
4	Software $\overline{STOPCLK}$ Deassertion Control: 0: Normal Operation 1: Force Deassertion of $\overline{STOPCLK}$ $\overline{STOPCLK}$ will remain inactive after this register is written to one until this register is written to zero.	0
3:2	Hardware Controlled Power Management Control with the expiration of the Standby Timer: 00: Will not automatically assert $\overline{STOPCLK}$ 01: Reserved 10: Will automatically assert $\overline{STOPCLK}$ only 11: Will automatically assert $\overline{STOPCLK}$	00
1:0	Hardware Controlled Power Management Control with the expiration of the Suspend Timer: 00: Will not automatically assert $\overline{STOPCLK}$ 01: Reserved 10: Will automatically assert $\overline{STOPCLK}$ only 11: Will automatically assert $\overline{STOPCLK}$	00

Register 66: $\overline{\text{STOPCLK}}$ Control Register (Read/Write) – Index=66H

Bit	Function	Default
7	$\overline{\text{STOPCLK}}$ Function Control: 0: Disable $\overline{\text{STOPCLK}}$ 1: Enable $\overline{\text{STOPCLK}}$	0
6	$\overline{\text{STOPCLK}}$ Throttling Control: 0: Disable $\overline{\text{STOPCLK}}$ throttling 1: Enable $\overline{\text{STOPCLK}}$ throttling	0
5	Reserved	0
4	$\overline{\text{STOPCLK}}$ Automatic Deassertion Control: 0: $\overline{\text{STOPCLK}}$ automatically deasserted when Standby events occur 1: $\overline{\text{STOPCLK}}$ not automatically deasserted when Standby events occur	00
3	$\overline{\text{STOPCLK}}$ Assertion Delay Timer Control: 0: Disable $\overline{\text{STOPCLK}}$ Assertion Delay Timer 1: Enable $\overline{\text{STOPCLK}}$ Assertion Delay Timer	00
2:1	$\overline{\text{STOPCLK}}$ Delay Timer: 00: 430 microseconds 01: 860 microseconds 10: 1.7 milliseconds 11: 7 milliseconds	0
0	Reserved	0

Register 67: Power Management $\overline{\text{SMI}}$ Control Register (Read/Write) – Index=67H

Bit	Function	Default
7	Software $\overline{\text{SMI}}$ Assertion Control: 0: Normal Operation 1: Force Assertion of $\overline{\text{SMI}}$ $\overline{\text{SMI}}$ will remain active after this register is written to one until this register is written to zero.	0
6	Software $\overline{\text{SMI}}$ Deassertion Control: 0: Normal Operation 1: Force Deassertion of $\overline{\text{SMI}}$ $\overline{\text{SMI}}$ will remain inactive after this register is written to one until this register is written to zero.	0
5:0	Reserved	000000

Register 70: Power Management $\overline{\text{SMI}}$ Enable Register #1 (Read/Write) – Index=70H

Bit	Function	Default
7	Non-motherboard Memory Access 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
6	Video Memory Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
5	Parallel Port Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
4	COM Port 2/4 Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
3	COM Port 1/3 Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	00
2	Hard Disk Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
1	Floppy Disk Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
0	Keyboard Controller Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0

Register 71: Power Management $\overline{\text{SMI}}$ Enable Register #2 (Read/Write) – Index=71H

Bit	Function	Default
7	User Timer 2 Timeout: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
6	User Timer 1 Timeout: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
5	Suspend Timer Timeout: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
4	Standby Timer Timeout: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
3	NMI: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	00
2	PCI Master Request: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
1	Programmable I/O Trap 2 Address Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0
0	Programmable I/O Trap 1 Address Access: 0: Will Not Generate $\overline{\text{SMI}}$ 1: Will Generate $\overline{\text{SMI}}$	0

Register 72: Power Management $\overline{SMI}$ Enable Register #3 (Read/Write) – Index=72H

Bit	Function	Default
7	Reserved	0
6	Automatic $\overline{STOPCLK}$ Assertion with APM Control Port Read Control: 0: No Automatic $\overline{STOPCLK}$ Assertion with APM Control Port Read 1: Automatic $\overline{STOPCLK}$ Assertion with APM Control Port Read	0
5	DMA Request: 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$ This register bit works in conjunction with Register F5H. When this bit is set to 1, any DMA Request that is enabled will generate an $\overline{SMI}$.	0
4	Interrupt Request: 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$ This register bit works in conjunction with Registers F3H and F4H. When this bit is set to 1, any Interrupt Request that is enabled will generate an $\overline{SMI}$.	0
3	The Assertion of the $\overline{EPMI}$ Signal: 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	00
2	APMC (Advanced Power Management Code) Request: 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
1	Software $\overline{SMI}$ Request: 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
0	User Timer 3 Timeout: 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0

Register 73: Power Management $\overline{SMI}$ Enable Register #4 (Read/Write) – Index=73H

Bit	Function	Default
7	Assertion of IRQ7 (Interrupt Request 7) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
6	Assertion of IRQ6 (Interrupt Request 6) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
5	Assertion of IRQ5 (Interrupt Request 5) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
4	Assertion of IRQ4 (Interrupt Request 4) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
3	Assertion of IRQ3 (Interrupt Request 3) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	00
2	Assertion of INTR (Any Interrupt Request) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
1	Assertion of IRQ1 (Interrupt Request 1) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
0	Assertion of IRQ0 (Interrupt Request 0) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0

Register 74: Power Management $\overline{SMI}$ Enable Register #5 (Read/Write) – Index=74H

Bit	Function	Default
7	Assertion of IRQ15 (Interrupt Request 15) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
6	Assertion of IRQ14 (Interrupt Request 14) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
5	Assertion of IRQ13 (Interrupt Request 13) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
4	Assertion of IRQ12 (Interrupt Request 12) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
3	Assertion of IRQ11 (Interrupt Request 11) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	00
2	Assertion of IRQ10 (Interrupt Request 10) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
1	Assertion of IRQ9 (Interrupt Request 9) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
0	Assertion of IRQ8 (Interrupt Request 8) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0

Register 75: Power Management $\overline{SMI}$ Enable Register #6 (Read/Write) – Index=75H

Bit	Function	Default
7	Assertion of DREQ7 (DMA Request 7) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
6	Assertion of DREQ6 (DMA Request 6) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
5	Assertion of DREQ5 (DMA Request 5) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
4	Assertion of DREQ (Any DMA Request) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
3	Assertion of DREQ3 (DMA Request 3) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	00
2	Assertion of DREQ2 (DMA Request 2) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
1	Assertion of DREQ1 (DMA Request 1) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0
0	Assertion of DREQ0 (DMA Request 0) 0: Will Not Generate $\overline{SMI}$ 1: Will Generate $\overline{SMI}$	0

Register 76: Power Management $\overline{SMI}$ Status Register #1 (Read/Write) – Index=76H

Bit	Function	Default
7	Non-motherboard Memory Access READ: 0: $\overline{SMI}$ was not caused by Non-motherboard Memory Access 1: $\overline{SMI}$ was caused by Non-motherboard Memory Access WRITE: 0: No change to register. 1: Clear Register	0
6	Video Memory Access READ: 0: $\overline{SMI}$ was not caused by Video Memory Access 1: $\overline{SMI}$ was caused by Video Memory Access WRITE: 0: No change to register. 1: Clear Register	0
5	Parallel Port Access READ: 0: $\overline{SMI}$ was not caused by Parallel Port Access 1: $\overline{SMI}$ was caused by Parallel Port Access WRITE: 0: No change to register. 1: Clear Register	0
4	COM Port 2/4 Access READ: 0: $\overline{SMI}$ was not caused by COM Port 2/4 Access 1: $\overline{SMI}$ was caused by COM Port 2/4 Access WRITE: 0: No change to register. 1: Clear Register	0
3	COM Port 1/3 Access READ: 0: $\overline{SMI}$ was not caused by COM Port 1/3 Access 1: $\overline{SMI}$ was caused by COM Port 1/3 Access WRITE: 0: No change to register. 1: Clear Register	00
2	Hard Disk Access READ: 0: $\overline{SMI}$ was not caused by Hard Disk Access 1: $\overline{SMI}$ was caused by Hard Disk Access WRITE: 0: No change to register. 1: Clear Register	0
1	Floppy Disk Access READ: 0: $\overline{SMI}$ was not caused by Floppy Disk Access 1: $\overline{SMI}$ was caused by Floppy Disk Access WRITE: 0: No change to register. 1: Clear Register	0
0	Keyboard Controller Access READ: 0: $\overline{SMI}$ was not caused by Keyboard Controller Access 1: $\overline{SMI}$ was caused by Keyboard Controller Access WRITE: 0: No change to register. 1: Clear Register	0

Register 77: Power Management $\overline{SMI}$ Status Register #2 (Read/Write) – Index=77H

Bit	Function	Default
7	User Timer 2 Timeout READ: 0: $\overline{SMI}$ was not caused by a User Timer 2 Timeout 1: $\overline{SMI}$ was caused by a User Timer 2 Timeout WRITE: 0: No change to register. 1: Clear Register	0
6	User Timer 1 Timeout READ: 0: $\overline{SMI}$ was not caused by a User Timer 1 Timeout 1: $\overline{SMI}$ was caused by a User Timer 1 Timeout WRITE: 0: No change to register. 1: Clear Register	0
5	Suspend Timer Timeout READ: 0: $\overline{SMI}$ was not caused by a Suspend Timer Timeout 1: $\overline{SMI}$ was caused by a Suspend Timer Timeout WRITE: 0: No change to register. 1: Clear Register	0
4	Standby Timer Timeout READ: 0: $\overline{SMI}$ was not caused by a Standby Timer Timeout 1: $\overline{SMI}$ was caused by a Standby Timer Timeout WRITE: 0: No change to register. 1: Clear Register	0
3	PCI Master Request READ: 0: $\overline{SMI}$ was not caused by a PCI Master Request 1: $\overline{SMI}$ was caused by a PCI Master Request WRITE: 0: No change to register. 1: Clear Register	00
2	NMI (Non-maskable Interrupt) READ: 0: $\overline{SMI}$ was not caused by an NMI 1: $\overline{SMI}$ was caused by an NMI WRITE: 0: No change to register. 1: Clear Register	0
1	Programmable I/O Trap 2 READ: 0: $\overline{SMI}$ was not caused by a Programmable I/O Trap 2 1: $\overline{SMI}$ was caused by a Programmable I/O Trap 2 This Register bit is cleared automatically on a read.	0
0	Programmable I/O Trap 1 READ: 0: $\overline{SMI}$ was not caused by a Programmable I/O Trap 1 1: $\overline{SMI}$ was caused by a Programmable I/O Trap 1 This Register bit is cleared automatically on a read.	0

Register 78: Power Management $\overline{SMI}$ Status Register #3 (Read/Write) – Index=78H

Bit	Function	Default
7	Standby Timer Status when in GREENPC Mode: 0: Timer continues to count. 1: Timer expired and GREENPC mode has been entered.	0
6	Suspend Timer Status when in SUSGREENPC Mode: 0: Timer continues to count. 1: Timer expired and SUSGREENPC mode has been entered.	0
5	Interrupt Request READ: 0: $\overline{SMI}$ was not caused by an Interrupt Request 1: $\overline{SMI}$ was caused by an Interrupt Request This Register bit is cleared automatically on a read. If this bit is 0, registers F9H-FAH can be ignored. However, if this bit is 1, registers F9H–FAH contain valid status and should be cleared after being read.	0
4	DMA Request READ: 0: $\overline{SMI}$ was not caused by a DMA Request 1: $\overline{SMI}$ was caused by a DMA Request This Register bit is cleared automatically on a read. If this bit is 0, register FBH can be ignored. However, if this bit is 1, register FBH contains valid status and should be cleared after being read.	0
3	$\overline{EPMI}$ Asserted READ: 0: $\overline{SMI}$ was not caused by an $\overline{EPMI}$ Assertion 1: $\overline{SMI}$ was caused by an $\overline{EPMI}$ Assertion WRITE: 0: No change to register. 1: Clear Register	00
2	APMC Write READ: 0: $\overline{SMI}$ was not caused by an APMC Write 1: $\overline{SMI}$ was caused by an APMC Write WRITE: 0: No change to register. 1: Clear Register	0
1	Software $\overline{SMI}$ Assertion (Through Register E7H) READ: 0: $\overline{SMI}$ was not caused by a Software $\overline{SMI}$ Assertion 1: $\overline{SMI}$ was caused by a Software $\overline{SMI}$ Assertion WRITE: 0: No change to register. 1: Clear Register	0
0	User Timer 3 Timeout READ: 0: $\overline{SMI}$ was not caused by a User Timer 3 Timeout 1: $\overline{SMI}$ was caused by a User Timer 3 Timeout WRITE: 0: No change to register. 1: Clear Register	0

Register 79: Power Management Interrupt Request Status Register #1 (Read/Write) – Index=79H

Bit	Function	Default
7	Assertion of IRQ7 (Interrupt Request 7) 0: $\overline{\text{SMI}}$ was not caused by an IRQ7 1: $\overline{\text{SMI}}$ was caused by an IRQ7 WRITE: 0: No change to register 1: Clear Register	0
6	Assertion of IRQ6 (Interrupt Request 6) 0: $\overline{\text{SMI}}$ was not caused by an IRQ6 1: $\overline{\text{SMI}}$ was caused by an IRQ6 WRITE: 0: No change to register 1: Clear Register	0
5	Assertion of IRQ5 (Interrupt Request 5) 0: $\overline{\text{SMI}}$ was not caused by an IRQ5 1: $\overline{\text{SMI}}$ was caused by an IRQ5 WRITE: 0: No change to register 1: Clear Register	0
4	Assertion of IRQ4 (Interrupt Request 4) 0: $\overline{\text{SMI}}$ was not caused by an IRQ4 1: $\overline{\text{SMI}}$ was caused by an IRQ4 WRITE: 0: No change to register 1: Clear Register	0
3	Assertion of IRQ3 (Interrupt Request 3) 0: $\overline{\text{SMI}}$ was not caused by an IRQ3 1: $\overline{\text{SMI}}$ was caused by an IRQ3 WRITE: 0: No change to register 1: Clear Register	00
2	Assertion of INTR (Any Interrupt Request) 0: $\overline{\text{SMI}}$ was not caused by an INTR 1: $\overline{\text{SMI}}$ was caused by an INTR WRITE: 0: No change to register 1: Clear Register	0
1	Assertion of IRQ1 (Interrupt Request 1) 0: $\overline{\text{SMI}}$ was not caused by an IRQ1 1: $\overline{\text{SMI}}$ was caused by an IRQ1 WRITE: 0: No change to register 1: Clear Register	0
0	Assertion of IRQ0 (Interrupt Request 0) 0: $\overline{\text{SMI}}$ was not caused by an IRQ0 1: $\overline{\text{SMI}}$ was caused by an IRQ0 WRITE: 0: No change to register 1: Clear Register	0

Register 7A: Power Management Interrupt Request Status Register #2 (Read/Write) – Index=7AH

Bit	Function	Default
7	Assertion of IRQ15 (Interrupt Request 15) 0: $\overline{\text{SMI}}$ was not caused by an IRQ15 1: $\overline{\text{SMI}}$ was caused by an IRQ15 WRITE: 0: No change to register 1: Clear Register	0
6	Assertion of IRQ14 (Interrupt Request 14) 0: $\overline{\text{SMI}}$ was not caused by an IRQ14 1: $\overline{\text{SMI}}$ was caused by an IRQ14 WRITE: 0: No change to register 1: Clear Register	0
5	Assertion of IRQ13 (Interrupt Request 13) 0: $\overline{\text{SMI}}$ was not caused by an IRQ13 1: $\overline{\text{SMI}}$ was caused by an IRQ13 WRITE: 0: No change to register 1: Clear Register	0
4	Assertion of IRQ12 (Interrupt Request 12) 0: $\overline{\text{SMI}}$ was not caused by an IRQ12 1: $\overline{\text{SMI}}$ was caused by an IRQ12 WRITE: 0: No change to register 1: Clear Register	0
3	Assertion of IRQ11 (Interrupt Request 11) 0: $\overline{\text{SMI}}$ was not caused by an IRQ11 1: $\overline{\text{SMI}}$ was caused by an IRQ11 WRITE: 0: No change to register 1: Clear Register	00
2	Assertion of IRQ10 (Interrupt Request 10) 0: $\overline{\text{SMI}}$ was not caused by an IRQ10 1: $\overline{\text{SMI}}$ was caused by an IRQ10 WRITE: 0: No change to register 1: Clear Register	0
1	Assertion of IRQ9 (Interrupt Request 9) 0: $\overline{\text{SMI}}$ was not caused by an IRQ9 1: $\overline{\text{SMI}}$ was caused by an IRQ9 WRITE: 0: No change to register 1: Clear Register	0
0	Assertion of IRQ8 (Interrupt Request 8) 0: $\overline{\text{SMI}}$ was not caused by an IRQ8 1: $\overline{\text{SMI}}$ was caused by an IRQ8 WRITE: 0: No change to register 1: Clear Register	0

Register 7B: Power Management DMA Request Status Register (Read/Write) – Index=7BH

Bit	Function	Default
7	Assertion of DREQ7 (DMA Request 7) 0: $\overline{SMI}$ was not caused by a DREQ7 1: $\overline{SMI}$ was caused by a DREQ7 WRITE: 0: No change to register 1: Clear Register	0
6	Assertion of DREQ6 (DMA Request 6) 0: $\overline{SMI}$ was not caused by a DREQ6 1: $\overline{SMI}$ was caused by a DREQ6 WRITE: 0: No change to register 1: Clear Register	0
5	Assertion of DREQ5 (DMA Request 5) 0: $\overline{SMI}$ was not caused by a DREQ5 1: $\overline{SMI}$ was caused by a DREQ5 WRITE: 0: No change to register 1: Clear Register	0
4	Assertion of DREQ (Any DMA Request) 0: $\overline{SMI}$ was not caused by a DREQ 1: $\overline{SMI}$ was caused by a DREQ WRITE: 0: No change to register 1: Clear Register	0
3	Assertion of DREQ3 (DMA Request 3) 0: $\overline{SMI}$ was not caused by a DREQ3 1: $\overline{SMI}$ was caused by a DREQ3 WRITE: 0: No change to register 1: Clear Register	00
2	Assertion of DREQ2 (DMA Request 2) 0: $\overline{SMI}$ was not caused by a DREQ2 1: $\overline{SMI}$ was caused by a DREQ2 WRITE: 0: No change to register 1: Clear Register	0
1	Assertion of DREQ1 (DMA Request 1) 0: $\overline{SMI}$ was not caused by a DREQ1 1: $\overline{SMI}$ was caused by a DREQ1 WRITE: 0: No change to register 1: Clear Register	0
0	Assertion of DREQ0 (DMA Request 0) 0: $\overline{SMI}$ was not caused by a DREQ0 1: $\overline{SMI}$ was caused by a DREQ0 WRITE: 0: No change to register 1: Clear Register	0

Register 7C: Reserved – Index=7CH

Bit	Function	Default
7:0	Reserved	00000000

Register 7D: Reserved – Index=7DH

Bit	Function	Default
7:0	Reserved	00000000

Register 7E: Reserved – Index=7EH

Bit	Function	Default
7:0	Reserved	00000000

Register 7F: Reserved – Index=7FH

Bit	Function	Default
7:0	Reserved	00000000

Special I/O Port Registers

The following ports are used for special functions. They are accessed by performing I/O transactions to the address specified.

Port 61: System Control Port B, NMI (Read/Write) - I/O Address=061H

Bit	Function	Default
7	Read Only. PCI System Error ($\overline{SERR}$) check: 0: No error 1: PCI $\overline{SERR}$ detected	0
6	Read Only. I/O channel check: 0: No error 1: I/O channel error detected (ISA $\overline{IOCHK}$ signal asserted)	0
5	Read Only. Timer 2 output: 0: Timer 2 (speaker) output is low 1: Timer 2 output is high	0
4	Read Only. Refresh detection: 0: Refresh request not detected 1: Refresh request detected	0
3	Read/Write port. I/O channel check: 0: Enable I/O channel check 1: Disable I/O channel check When enabled an I/O channel error will generate an NMI to the CPU.	0
2	Read/Write port. PCI System Error ($\overline{SERR}$) check: 0: Enable PCI $\overline{SERR}$ check 1: Disable PCI $\overline{SERR}$ check When enabled a PCI system error will generate an NMI to the CPU.	0
1	Read/Write port. Speaker data.	0
0	Read/Write. Timer 2 speaker gate: 0: Timer 2 speaker gate disabled 1: Timer 2 speaker gate enabled	0

Port 70: RTC/Configuration RAM Address Port, NMI (Write) - I/O Address=070H

Bit	Function	Default
7	NMI reporting: 0: Enable NMI reporting 1: Disable NMI reporting Sources of NMI can be either from ISA I/O channel error or PCI $\overline{SERR}$ depending on bits 3 and 2 of register port 61.	0
6:0	Configuration RAM address. Index of RTC Configuration Register for RTC access (See CY82C693U Real-Time Clock Register section).	0000000

Port 92: PS/2 Reset Control (Read/Write) - I/O Address=092H

Bit	Function	Default
7:2	Reserved	00000000
1	GTA20 Control: 0: Activate GTA20. 1: Deactivate GTA20.	0
0	Fast Reset Control: 0: INIT deactivated. Will occur automatically after reset timer expires. 1: Assert INIT.	0

Port B2: APM Control Port (Read/Write) - I/O Address=0B2H

Bit	Function	Default
7:0	This Read/Write Port can be used for APM software. A write to this register will generate an $\overline{\text{SMI}}$.	00000000

Port B3: APM Status Port (Read/Write) - I/O Address=0B3H

Bit	Function	Default
7:0	This Read/Write Port can be used for APM software.	00000000

CY82C693U DMA Controller Registers

There are two DMA controllers cascaded together inside the CY82C693U. (DMAC1 and DMAC2). Each DMA Controller contains four channels. DMAC1 controls the 8-bit DMA operations and DMAC2 controls 16-bit DMA operations. Channel 0 of DMAC2 provides the cascade between the two controllers, and therefore, may not be used for DMA data transfers. DMAC1's DMA request and acknowledge signals correspond to the DREQ0–DREQ3 and DACK0–DACK3 signals. DMAC2's DMA request and acknowledge signals correspond to the DREQ5–DREQ7 and DACK5–DACK7 signals.

The internal registers for the CY82C693U DMA controllers are defined in this section. The registers can be accessed by performing I/O reads and writes to Addresses 000H through 00FH (for DMAC1) and Addresses 0C0H through 0CFH (for DMAC2). The DMA Page (Upper Order DMA address bits are controlled using I/O addresses 080H–08FH). All DMAC registers are eight bits (1 byte) wide. Performing I/O accesses to these address ranges will place the corresponding DMA controller into its Program State.

The DMA controller register address space has been increased using an additional address flip-flop. The flip-flop toggles every time an access occurs to the DMA word count or DMA address registers. The flip-flop is cleared by the assertion of the CPURST signal (from the CY82C693U) or a MASTER CLEAR from the DMA registers. The flip-flop can also be programmed by an access to the flip-flop control register.

The DMA channels within the controllers should be masked prior to entering the Program State to insure that a DMA access is not attempted to a partially programmed channel.

After the DMA controllers are programmed, they should be unmasked. This will allow the DMA controllers to enter the ACTIVE State. The ACTIVE State will be entered when a valid DMA request is recognized by the CY82C693U. During 8-bit DMA transfers, DMAC1 places the memory address bits on Address 0 through 16. For 16-bit DMA transfers, DMAC2 places the memory address bits on Address 1 through 16 (Address 0 is zero for 16-bit transfers). The page address bits are placed on Address 17 through 23.

DMA registers 0 through 7 provide access to each channel's Current Address Register, Current Word Count Register, Base Address Register, and Base Word Count Register. The Current Address Register contains the 16-bit address used during transfers. The value in the Current Address Register is either incremented or decremented (programmable) to provide the transfer addresses. Channel 0 will hold its address (without incrementing or decrementing) by setting the Address Hold Bit in the Command Register. If Autoinitializtion is selected, the Current Address Register is reloaded with the contents of the Base Address Register when the terminal count is reached in the Current Word Count Register. The Current Word Count Register contains the number of transfers to perform. This register is decremented with each transfer. The terminal count will be reached on the transition of this register from 0000H to FFFFH. Therefore, the actual number of transfers will be one more than the value loaded in the Current Word Count Register. When the terminal count is reached, the channel will be suspended and either masked or autoinitialized. The Base Address Register is write-only and is loaded along with the Current Address Register. This register stores the initial value of the Current Address Register and will reload its contents into the Current Address Register when the terminal count is reached if autoinitialization is selected. Likewise, the Base Word Count Register is used to reload the Current Word Count Register with its initial value if autoinitialization is programmed.

DMA Register 0: DMAC1 Channel 0 Current Address Register (Read/Write) - I/O Address=000H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 1: DMAC1 Channel 0 Current Word Count Register (Read/Write) - I/O Address=001H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 2: DMAC1 Channel 1 Current Address Register (Read/Write) - I/O Address=002H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 3: DMAC1 Channel 1 Current Word Count Register (Read/Write) - I/O Address=003H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 4: DMAC1 Channel 2 Current Address Register (Read/Write) - I/O Address=004H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 5: DMAC1 Channel 2 Current Word Count Register (Read/Write) - I/O Address=005H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 6: DMAC1 Channel 3 Current Address Register (Read/Write) - I/O Address=006H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 7: DMAC1 Channel 3 Current Word Count Register (Read/Write) - I/O Address=007H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 8: DMAC1 Status/Command Register (Read/Write) - I/O Address=008H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Read Status Register
1	0	X	Write Command Register

Status Register Format (Read Only)

Bit	Function	Default
7	DMA Request for Channel 3: 0: No DMA request is pending for Channel 3 1: DMA request is pending for Channel 3 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
6	DMA Request for Channel 2: 0: No DMA request is pending for Channel 2 1: DMA request is pending for Channel 2 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
5	DMA Request for Channel 1: 0: No DMA request is pending for Channel 1 1: DMA request is pending for Channel 1 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
4	DMA Request for Channel 0: 0: No DMA request is pending for Channel 0 1: DMA request is pending for Channel 0 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
3	Terminal Count Status on Channel 3: 0: Terminal Count Not Reached on Channel 3 1: Terminal Count Reached on Channel 3 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0
2	Terminal Count Status on Channel 2: 0: Terminal Count Not Reached on Channel 2 1: Terminal Count Reached on Channel 2 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0
1	Terminal Count Status on Channel 1: 0: Terminal Count Not Reached on Channel 1 1: Terminal Count Reached on Channel 1 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0
0	Terminal Count Status on Channel 0: 0: Terminal Count Not Reached on Channel 0 1: Terminal Count Reached on Channel 0 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0

Command Register Format (Write Only)

Bit	Function	Default
7	DMA Acknowledge Signal Active Level Control: 0: DACK signals are active LOW 1: DACK signals are active HIGH	0
6	DMA Request Signal Active Level Control: 0: DREQ signals are active HIGH 1: DREQ signals are active LOW	0
5	Reserved, Must be 0	0
4	DMA Priority Control: 0: DMA Requests will be honored according to fixed priority (Channel 0 has highest priority/Channel 7 has lowest priority) 1: DMA Requests will be honored according to rotating priority (Every time a channel is acknowledged, it rotates to lowest priority)	0
3	DMA Compressed Timing Control: 0: Disable Compressed Timing 1: Enable Compressed Timing Normal DMA word transfers take 4 DMA clock cycles. Compressed timing causes the command signals and the terminal count signal to be asserted one cycle earlier. This allows the entire DMA transfer to be compressed to 3 DMA clock cycles.	0
2	DMA Controller Disable Control: 0: Normal Operation 1: Disable DMA Controllers Disabling the DMA Controllers prevents DMA cycles from occurring during channel programming.	0
1	Address Hold Control: 0: Normal Operation 1: Force the value in Channel 0's Current Address Register to remain the same (no increment or decrement). Used for memory to memory transfers.	0
0	Memory to Memory Transfer Control: 0: Normal Operation 1: Channel 0 and channel 1 will be used for memory to memory transfers.	0

DMA Register 9: DMAC1 DMA Request Register (Write Only) - I/O Address=009H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write DMA Request Register

DMA Request Register Write Format. (This Register is used to generate DMA requests through software. Multiple Software requests can be generated with separate writes to this register. Software DMA Requests cannot be masked)

Bit	Function	Default
7:3	Reserved	00000
2	DMA Request Generation Control: 0: Do not generate a DMA Request 1: Force a DMA Request on the channel specified by bits[1:0].	0
1:0	DMA Request Channel Selector: 00: Channel 0 01: Channel 1 10: Channel 2 11: Channel 3	00

DMA Register 10: DMAC1 DMA Command/Mask Register (Write Only) - I/O Address=00AH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write single-bit in DMA Request Mask Register (leaving the rest unchanged)

DMA Request Mask Register Write Single Bit Format. (This Register is used to mask DMA requests through software. Single-bit Software request masks can be generated with separate writes to this register or all channels can be masked using DMA Register 15)

Bit	Function	Default
7:3	Reserved	00000
2	DMA Request Mask Generation Control: 0: Clear the mask on the channel specified by bits[1:0] 1: Force a DMA Request on the channel specified by bits[1:0].	0
1:0	DMA Request Mask Channel Selector: 00: Channel 0 01: Channel 1 10: Channel 2 11: Channel 3	00

DMA Register 11: DMAC1 DMA Mode Register (Read/Write) - I/O Address=00BH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write Mode Register

Mode Register Format. (Should be programmed for each channel)

Bit	Function	Default
7:6	DMA Mode Selection Control: 00: Demand Transfer Mode 01: Single Transfer Mode 10: Block Transfer Mode 11: Cascade Mode See DMA Controller description for details about each mode.	00
5	Counter Direction Control: 0: Increment Address Counter After Each Transfer 1: Decrement Address Counter After Each Transfer	0
4	Autoinitialization Control: 0: Disable Autoinitialization 1: Enable Autoinitialization Autoinitialization will restore the initial values into the Current Address Register and Word Count Register when the terminal count is reached. The channel will not automatically be masked if it is autoinitialized.	0
3:2	DMA Transfer Type Selection Control: 00: Verify Transfer 01: Write Transfer 10: Read Transfer 11: Undefined (DO NOT USE) See DMA Controller description for details about each transfer type.	00
1:0	Channel Selector: 00: Channel 0 01: Channel 1 10: Channel 2 11: Channel 3 Each DMA channel has its own mode register, but they are all accessed through I/O Address 00BH. For Mode Register writes, bits[1:0] control which channel's mode register will be written. To read each channel's mode register, four sequential reads will walk through all of the mode registers. Clearing the Mode Register Counter (DMA Register 14) will start the read sequence at a known state (channel 0). During reads, the channel selector bits will be 11 regardless of the channel.	00

DMA Register 12: DMAC1 Address Space Expansion Flip-Flop Control Register (Write Only) - I/O Address=00CH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Clear Flip-Flop (Flip-Flop=0). This is a special command. The data lines are ignored.

DMA Register 13: DMAC1 Master Clear Register (Write Only) - I/O Address=00DH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Perform Master Clear. This is a special command. The data lines are ignored.

DMA Register 14: DMAC1 DMA Mask Clear Register (Write Only) - I/O Address=00EH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Clear DMA Request Mask Bits (Unmask all DMA requests). This is a special command. The data lines are ignored.

DMA Register 15: DMAC1 Request Mask Register Control (Read/Write) - I/O Address=00FH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Read All bits of DMA Request Mask Register
1	0	X	Write All bits of DMA Request Mask Register

DMA Request Mask Register Read and Write All Bits Format (This register is cleared on CPURST)

Bit	Function	Default
7:4	Reserved	0000
3	DMA Channel 3 Request Mask: 0: Channel 3 Not Masked 1: Channel 3 Masked	0
2	DMA Channel 2 Request Mask: 0: Channel 2 Not Masked 1: Channel 2 Masked	0
1	DMA Channel 1 Request Mask: 0: Channel 1 Not Masked 1: Channel 1 Masked	0
0	DMA Channel 0 Request Mask: 0: Channel 0 Not Masked 1: Channel 0 Masked	0

DMA Register 16: DMAC2 Channel 0 (Channel 4) Current Address Register (Read/Write) - I/O Address=0C0H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 17: DMAC2 Channel 0 (Channel 4) Current Word Count Register (Read/Write) - I/O Address=0C2H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 18: DMAC2 Channel 1 (Channel 5) Current Address Register (Read/Write) - I/O Address=0C4H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 19: DMAC2 Channel 1 (Channel 5) Current Word Count Register (Read/Write) - I/O Address=0C6H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 20: DMAC2 Channel 2 (Channel 6) Current Address Register (Read/Write) - I/O Address=0C8H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 21: DMAC2 Channel 2 (Channel 6) Current Word Count Register (Read/Write) - I/O Address=0CAH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 22: DMAC2 Channel 3 (Channel 7) Current Address Register (Read/Write) - I/O Address=0CCH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Address Register
0	1	1	Read High Byte of Current Address Register
1	0	0	Write Low Byte of Base Address Register and Current Address Register
1	0	1	Write High Byte of Base Address Register and Current Address Register

DMA Register 23: DMAC2 Channel 3 (Channel 7) Current Word Count Register (Read/Write) - I/O Address=0CEH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	0	Read Low Byte of Current Word Count Register
0	1	1	Read High Byte of Current Word Count Register
1	0	0	Write Low Byte of Base Word Count Register and Current Word Count Register
1	0	1	Write High Byte of Base Word Count Register and Current Word Count Register

DMA Register 24: DMAC2 Status/Command Register (Read/Write) - I/O Address=0D0H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Read Status Register
1	0	X	Write Command Register

Status Register Format (Read Only)

Bit	Function	Default
7	DMA Request for Channel 7: 0: No DMA request is pending for Channel 7 1: DMA request is pending for Channel 7 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
6	DMA Request for Channel 6: 0: No DMA request is pending for Channel 6 1: DMA request is pending for Channel 6 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
5	DMA Request for Channel 5: 0: No DMA request is pending for Channel 5 1: DMA request is pending for Channel 5 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
4	DMA Request for Channel 4: 0: No DMA request is pending for Channel 4 1: DMA request is pending for Channel 4 This bit is cleared by CPURST, Master Clear, or on the deassertion of the DMA Request. This bit will not be masked (even if the channel is masked).	0
3	Terminal Count Status on Channel 7: 0: Terminal Count Not Reached on Channel 7 1: Terminal Count Reached on Channel 7 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0
2	Terminal Count Status on Channel 6: 0: Terminal Count Not Reached on Channel 6 1: Terminal Count Reached on Channel 6 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0
1	Terminal Count Status on Channel 5: 0: Terminal Count Not Reached on Channel 5 1: Terminal Count Reached on Channel 5 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0
0	Terminal Count Status on Channel 4: 0: Terminal Count Not Reached on Channel 4 1: Terminal Count Reached on Channel 4 This bit is cleared by CPURST, Master Clear, or on a Status Register Read.	0

Command Register Format (Write Only)

Bit	Function	Default
7	DMA Acknowledge Signal Active Level Control: 0: DACK signals are active LOW 1: DACK signals are active HIGH	0
6	DMA Request Signal Active Level Control: 0: DREQ signals are active HIGH 1: DREQ signals are active LOW	0
5	Reserved, Must be 0	0
4	DMA Priority Control: 0: DMA Requests will be honored according to fixed priority (Channel 0 has highest priority/Channel 7 has lowest priority) 1: DMA Requests will be honored according to rotating priority (Every time a channel is acknowledged, it rotates to lowest priority)	0
3	DMA Compressed Timing Control: 0: Disable Compressed Timing 1: Enable Compressed Timing Normal DMA word transfers take 4 DMA clock cycles. Compressed timing causes the command signals and the terminal count signal to be asserted one cycle earlier. This allows the entire DMA transfer to be compressed to 3 DMA clock cycles.	0
2	DMA Controller Disable Control: 0: Normal Operation 1: Disable DMA Controllers Disabling the DMA Controllers prevents DMA cycles from occurring during channel programming.	0
1	Address Hold Control: 0: Normal Operation 1: Force the value in Channel 0's Current Address Register to remain the same (no increment or decrement). Used for memory to memory transfers.	0
0	Memory to Memory Transfer Control: 0: Normal Operation 1: Channel 0 and channel 1 will be used for memory to memory transfers.	0

DMA Register 25: DMAC2 DMA Request Register (Write Only) - I/O Address=0D2H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write DMA Request Register

DMA Request Register Write Format (This Register is used to generate DMA requests through software. Multiple Software requests can be generated with separate writes to this register. Software DMA Requests cannot be masked.)

Bit	Function	Default
7:3	Reserved	00000
2	DMA Request Generation Control: 0: Do not generate a DMA Request 1: Force a DMA Request on the channel specified by bits[1:0].	0
1:0	DMA Request Channel Selector: 00: Channel 4 01: Channel 5 10: Channel 6 11: Channel 7	00

DMA Register 26: DMAC2 DMA Command/Mask Register (Write Only) - I/O Address=0D4H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write single-bit in DMA Request Mask Register (leaving the rest unchanged)

DMA Request Mask Register Write Single Bit Format (This Register is used to mask DMA requests through software. Single-bit Software request masks can be generated with separate writes to this register or all channels can be masked using DMA Register 15.)

Bit	Function	Default
7:3	Reserved	00000
2	DMA Request Mask Generation Control: 0: Clear the mask on the channel specified by bits[1:0] 1: Force a DMA Request on the channel specified by bits[1:0].	0
1:0	DMA Request Mask Channel Selector: 00: Channel 4 01: Channel 5 10: Channel 6 11: Channel 7	00

DMA Register 27: DMAC2 DMA Mode Register (Write Only) - I/O Address=0D6H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write Mode Register

Mode Register Format (Should be programmed for each channel)

Bit	Function	Default
7:6	DMA Mode Selection Control: 00: Demand Transfer Mode 01: Single Transfer Mode 10: Block Transfer Mode 11: Cascade Mode See DMA Controller description for details about each mode.	00
5	Counter Direction Control: 0: Increment Address Counter After Each Transfer 1: Decrement Address Counter After Each Transfer	0
4	Autoinitialization Control: 0: Disable Autoinitialization 1: Enable Autoinitialization Autoinitialization will restore the initial values into the Current Address Register and Word Count Register when the terminal count is reached. The channel will not automatically be masked if it is autoinitialized.	0
3:2	DMA Transfer Type Selection Control: 00: Verify Transfer 01: Write Transfer 10: Read Transfer 11: Undefined (DO NOT USE) See DMA Controller description for details about each transfer type.	00
1:0	Channel Selector: 00: Channel 0 01: Channel 1 10: Channel 2 11: Channel 3 Each DMA channel has its own mode register, but they are all accessed through I/O Address 00BH. For Mode Register writes, bits[1:0] control which channel's mode register will be written. To read each channel's mode register, four sequential reads will walk through all of the mode registers. Clearing the Mode Register Counter (DMA Register 14) will start the read sequence at a known state (channel 0). During reads, the channel selector bits will be 11 regardless of the channel.	00

DMA Register 28: DMAC2 Address Space Expansion Flip-Flop Control Register (Write Only) - I/O Address=0D8H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Clear Flip-Flop (Flip-Flop=0). This is a special command. The data lines are ignored.

DMA Register 29: DMAC2 Master Clear Register (Write Only) - I/O Address=0DAH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Perform Master Clear. This is a special command. The data lines are ignored.

DMA Register 30: DMAC2 DMA Mask Clear Register (Write Only) - I/O Address=0DCH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Clear DMA Request Mask Bits (Unmask all DMA requests). This is a special command. The data lines are ignored.

DMA Register 31: DMAC2 Request Mask Register Control (Read/Write) - I/O Address=0DEH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Read DMA Request Mask Register
1	0	X	Write DMA Request Mask Register

DMA Request Mask Register Read and Write All Bits Format (This register is cleared on CPURST)

Bit	Function	Default
7:4	Reserved	0000
3	DMA Channel 7 Request Mask: 0: Channel 7 Not Masked 1: Channel 7 Masked	0
2	DMA Channel 6 Request Mask: 0: Channel 6 Not Masked 1: Channel 6 Masked	0
1	DMA Channel 5 Request Mask: 0: Channel 5 Not Masked 1: Channel 5 Masked	0
0	DMA Channel 4 Request Mask: 0: Channel 4 Not Masked 1: Channel 4 Masked	0

DMA Register 32: DMAC1 Channel 2 Page Address Register (Read/Write) – Index=081H

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC1 (8-bit DMA Controller), Channel 2	00000000

DMA Register 33: DMAC1 Channel 3 Page Address Register (Read/Write) – Index=082H

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC1 (8-bit DMA Controller), Channel 3	00000000

DMA Register 34: DMAC1 Channel 1 Page Address Register (Read/Write) – Index=083H

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC1 (8-bit DMA Controller), Channel 1	00000000

DMA Register 35: DMAC1 Channel 0 Page Address Register (Read/Write) – Index=087H

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC1 (8-bit DMA Controller), Channel 0	00000000

DMA Register 36: DMAC2 Channel 6 Page Address Register (Read/Write) – Index=089H

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC2 (16-bit DMA Controller), Channel 6	00000000

DMA Register 37: DMAC2 Channel 7 Page Address Register (Read/Write) – Index=08AH

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC2 (16-bit DMA Controller), Channel 7	00000000

DMA Register 38: DMAC2 Channel 5 Page Address Register (Read/Write) – Index=08BH

Bit	Function	Default
7:0	Page Address (Address bits 23-16) for DMAC2 (16-bit DMA Controller), Channel 5	00000000

DMA Register 39: DMAC1 Extended Mode Control (Write Only) - I/O Address=40BH

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write DMA Request Mask Register

DMAC1 Extended Mode Control Register Format (This register is cleared on CPURST)

Bit	Function	Default
7:6	Reserved	00
5:4	DMA Cycle Timing Mode: 00: Compatible Timing 01: Type “A” Timing 10: Type “B” Timing 11: Type “F” Timing	00
3	DMA Channel 3 Select: 0: Channel 3 Not Selected 1: Channel 3 Selected	0
2	DMA Channel 2 Select: 0: Channel 2 Not Selected 1: Channel 2 Selected	0
1	DMA Channel 1 Select: 0: Channel 1 Not Selected 1: Channel 1 Selected	0
0	DMA Channel 0 Select: 0: Channel 0 Not Selected 1: Channel 0 Selected	0

DMA Register 40: DMAC1 Channel 2 High Page Address Register (Read/Write) – Index=481H

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC1 (8-bit DMA Controller), Channel 2	00000000

DMA Register 41: DMAC1 Channel 3 High Page Address Register (Read/Write) – Index=482H

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC1 (8-bit DMA Controller), Channel 3	00000000

DMA Register 42: DMAC1 Channel 1 High Page Address Register (Read/Write) – Index=483H

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC1 (8-bit DMA Controller), Channel 1	00000000

DMA Register 43: DMAC1 Channel 0 High Page Address Register (Read/Write) – Index=487H

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC1 (8-bit DMA Controller), Channel 0	00000000

DMA Register 44: DMAC2 Channel 6 High Page Address Register (Read/Write) – Index=489H

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC2 (16-bit DMA Controller), Channel 6	00000000

DMA Register 45: DMAC2 Channel 7 High Page Address Register (Read/Write) – Index=48AH

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC2 (16-bit DMA Controller), Channel 7	00000000

DMA Register 46: DMAC2 Channel 5 High Page Address Register (Read/Write) – Index=48BH

Bit	Function	Default
7:0	High Page Address (Address bits 31-24) for DMAC2 (16-bit DMA Controller), Channel 5	00000000

DMA Register 47: DMAC2 Extended Mode Control (Write Only) - I/O Address=4D6H

I/O Read	I/O Write	Flip-Flop State	Function
0	1	X	Undefined
1	0	X	Write DMA Request Mask Register

DMAC2 Extended Mode Control Register Format (This register is cleared on CPURST).

Bit	Function	Default
7:6	Reserved	00
5:4	DMA Cycle Timing Mode: 00: Compatible Timing 01: Type “A” Timing 10: Type “B” Timing 11: Type “F” Timing	00
3	DMA Channel 7 Select: 0: Channel 7 Not Selected 1: Channel 7 Selected	0
2	DMA Channel 6 Select: 0: Channel 6 Not Selected 1: Channel 6 Selected	0
1	DMA Channel 5 Select: 0: Channel 5 Not Selected 1: Channel 5 Selected	0
0	DMA Channel 4 Select: 0: Channel 4Not Selected 1: Channel 4Selected	0

CY82C693U IDE (Bus Mastering) DMA Controller Registers

The CY82C693U supports two channels of DMA for the IDE controller. IDE DMA is compatible with SFF-8038i (the Small Form Factor Committee specification defining bus mastering on IDE). Only 16-bit DMA operation is supported.

The IDE DMA controller supports scatter-gather. Scatter-gather operation requires setting up a linked list of DMA information in memory (Physical Region Descriptor Table) and programming the starting location of the linked-list into the DMA controller itself. The controller will issue master cycles on PCI to gather and load the DMA information (word count and starting address). After each DMA terminal count is reached, the next entry in the linked-list is loaded until the end of the list is reached.

Some DMA IDE Registers are referenced to a Base Address + an offset. The base address should be programmed in PCI configuration space (see the “Bus Master IDE I/O Base Address Register”).

SFF-8038i Registers

Offset from Base Address	Register	R/W
00H	Bus Master IDE Command Register (Primary Channel)	Readable/ Writable
01H	Reserved	Do not access
02H	Bus Master IDE Status Register (Primary Channel)	Readable/ Writable- Writing a 1 clears the corresponding bit.
03H	Reserved	Do not access
04H-07H	Bus Master IDE Descriptor Table Pointer Register (Primary Channel)	Readable/ Writable
08H	Bus Master IDE Command Register (Secondary Channel)	Readable/ Writable
09H	Reserved	Do not access
0AH	Bus Master IDE Status Register (Secondary Channel)	Readable/ Writable- Writing a 1 clears the corresponding bit.
0BH	Reserved	Do not access
0CH-0FH	Bus Master IDE Descriptor Table Pointer Register (Secondary Channel)	Readable/ Writable

Bus Master IDE Command Register Format (Offset+00H for Primary Channel; Offset +08H for Secondary Channel)

Bit	Function	Default
7:4	Reserved	0000
3	Bus Master (DMA) Transfer Direction: 0: PCI bus master read. 1: PCI bus master write. NOTE: This bit must NOT be changed when the bus master function is active.	0
2:1	Reserved	00
0	Start/Stop Bus Master Transfer: Bus Master operation begins when this bit is written from a 0 to a 1. The controller will transfer data between the IDE device and memory when this bit is set. Master operation can be halted by writing a 0 to this bit. All state information is lost when a master operation is halted. Therefore, master operations cannot be halted and then resumed.	0

Bus Master IDE Status Register Format (Offset+02H for Primary Channel; Offset +0AH for Secondary Channel)

Bit	Function	Default
7	Simplex Status (Read Only): 0: Two channels operate independently and can be used at the same time. 1: Only one channel may be used at a time.	0
6	Drive 1 DMA Capability Status: 0: Drive 1 is not capable of DMA transfers. 1: Drive 1 is capable of DMA transfers.	0
5	Drive 0 DMA Capability Status: 0: Drive 0 is not capable of DMA transfers. 1: Drive 0 is capable of DMA transfers.	0
4:3	Reserved	00
2	IDE Interrupt Pending: 0: No IDE Interrupt Pending 1: IDE Interrupt Pending. Writing a '1' to this bit will reset the status to "No IDE Interrupt Pending".	0
1	IDE DMA Error Encountered: 0: No Error Detected 1: An Error Occurred in the Transfer of Data to/from Memory. Writing a '1' to this bit will reset the status to "No IDE Interrupt Pending".	0
0	Bus Master IDE Active: 0: There is no bus master active on IDE. 1: The Start Bit has been set in the Command Register. This bit is cleared when the last transfer for a region is performed, where EOT for that region is set in the region descriptor. It is also cleared when the Start Bit is cleared in the Command Register. When this bit is read as '0', all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted.	0

Bus Master IDE Descriptor Table Pointer Register Format
(Offset+04H-07H for Primary Channel; Offset +0CH-0FH for Secondary Channel)

Bit	Function	Default
31:2	Base Address of the Descriptor Table. Corresponds to PAD[31:2] of the address that the controller will read from to obtain the first Descriptor Table Entry.	00000000H
1:0	Reserved	00

Bus Master IDE I/O Base Address Register (PCI Configuration Space, function 1, register address 20-23H)

Bit	Function	Default
31:16	Reserved	0000H
15:4	Bus Master Interface Base Address: This sets the Base Address (Corresponding to PAD[15:4]) that will be added with an offset value to access the SFF-8038i Registers.	0000H
3:2	Reserved, hardwired to 00	00
1	Reserved	0
0	Resource Type Indicator (Read Only): This bit is hardwired to '1' indicating that the base address field in this register maps to I/O space.	0

hyperCache Specific (Not Required by SFF-8038i) Registers

The following registers define options/functions that are not explicitly covered by the SFF-8038i spec. Nonetheless, these registers should be programmed for complete operation.

Bus Master IDE Channel 0 Configuration Register (I/O Address 22H with Data = 30 (Index Port); I/O Address 23H is the Data Port)

Bit	Function	Default
7:3	Reserved. Will Return '00000' on read.	00000
2	IDE DMA Transfer Mode: 0: Multiple 1: Single	0
1:0	IDE DMA Transfer Speed Mode: 00: Mode 0 01: Mode 1 10: Mode 2 11: Reserved	00

Bus Master IDE Channel 1 Configuration Register (I/O Address 22H with Data = 31 (Index Port); I/O Address 23H is the Data Port)

Bit	Function	Default
7:3	Reserved. Will Return '00000' on read.	00000
2	IDE DMA Transfer Mode: 0: Multiple 1: Single	0
1:0	IDE DMA Transfer Speed Mode: 00: Mode 0 01: Mode 1 10: Mode 2 11: Reserved	00

Bus Master IDE TimeOut Register (I/O Address 22H with Data = 32 (Index Port); I/O Address 23H is the Data Port)

Bit	Function	Default
7:0	IDE DMA time out counter value. This register provides the terminal count on a counter with a 14.318 MHz clock input. Therefore, to find the timeout period, multiply the value in this register by 69.8 ns.	00000000

Bus Master IDE Test Register (I/O Address 22H with Data = 33 (Index Port); I/O Address 23H is the Data Port)

Bit	Function	Default
7:0	Undefined on read; Must write 00000000 on writes to this register.	00000000

CY82C693U Interrupt Controller Registers

There are two Interrupt controllers cascaded together inside the CY82C693U. (INTC1 and INTC2). The controllers each accept up to eight requests (from the Interrupt Request pins), resolve interrupt priority, assert the INTR pin to the CPU, and, in response to an interrupt acknowledge cycle, will return the appropriate Interrupt Vector (an address that points to the interrupt service routine).

A subset of the interrupt request lines are dedicated for specific system operations. They include: IRQ0 - The output of the system timer inside the CY82C693U; IRQ1 - The keyboard request interrupt; IRQ2 - The cascade between INTC1 and INTC2; IRQ8 - The alarm from the Real-Time-Clock; IRQ12 - The mouse request interrupt; and IRQ13 - The coproccessor error signal. These interrupt requests are not generally available for other interrupt functions. IRQ0 and IRQ2 are not brought out to external CY82C693U pins (the connection is made internally). The rest of the dedicated interrupts are available when CY82C693U functions are disabled. IRQ1 is available on the KBCLK (keyboard clock) pin when the internal keyboard controller is disabled. IRQ8 is available on the $\overline{\text{PSRSTB}}$ (power supply to battery source signal for the RTC) when the internal RTC is disabled. IRQ12 is available on the MSCLK (mouse clock) pin when the internal keyboard controller is disabled. IRQ13 is available on the $\overline{\text{FERR}}$ pin

IRQ[15:14], IRQ[11:9], and IRQ[7:3] are available for system use. There are traditional functions for these interrupt requests. However, the system designer can configure these inputs for many uses. The traditional uses are as follows: IRQ3 - Serial Port 2 interrupt request; IRQ4 - Serial Port 1 interrupt request; IRQ5 - Parallel Port 2 interrupt request; IRQ6 - Floppy Disk Controller interrupt request; IRQ7 - Parallel Port 1 interrupt request; IRQ9 - ISA/PCI slot interrupt request; IRQ10 - ISA/PCI slot interrupt request; IRQ11 - ISA/PCI slot interrupt request; IRQ14 - Hard Disk interrupt request; and IRQ15 - ISA/PCI slot interrupt request.

When any number of interrupt requests are asserted to the CY82C693U, bits are set in the Interrupt Request Register (IRR) corresponding to the asserted interrupt requests. The In Service Register (ISR) will store bits corresponding to all interrupt channels that are currently being serviced. The Interrupt Mask Register (IMR) will allow interrupt requests to be gated (masked off) by setting bits corresponding to the channels to be masked. All of the above registers output to priority resolution logic. The resolution logic will evaluate the inputs, determine a priority for interrupt servicing, assert INTR to the CPU, and latch the highest priority (non-masked) channel value into the ISR. A vector corresponding to the highest priority (non-masked) interrupt will be provided to the PCI bus in response to an Interrupt Acknowledge cycle.

The sequence for handling a peripheral interrupt is as follows:

1. Interrupt requests (one or multiple) are issued to the CY82C693U through the IRQ pins, the $\overline{\text{PCIINT}}$ pins, or the internal requestors. This sets the corresponding bit(s) in the IRR.
2. Using the values stored in the IRR, ISR, and IMR, the next interrupt to be serviced is determined. INTR is asserted to the CPU.
3. When the CPU recognizes the assertion of INTR, it responds with an Interrupt Acknowledge Cycle. The Interrupt Acknowledge Cycle is passed on to the PCI bus.
4. When the CY82C693U sees a PCI Interrupt Acknowledge Cycle, it will respond with a vector (address corresponding to the appropriate Interrupt Service Routine).
5. The CPU uses the vector to jump to the Interrupt Service Routine and begins execution of the interrupt handler

The ISR will be reset on an End-of-Interrupt (EOI). An EOI is a special command that the interrupt handler code must issue to the CY82C693U. A specific EOI can be sent to the CY82C693U to clear a specific bit in the ISR, or the highest priority interrupt can be cleared (non-specific). Masked interrupts will not be cleared for a non-specific EOI. If Automatic End-of-Interrupt (AEOI) is programmed, the highest priority (non-masked) interrupt bit will be cleared in the ISR after the Interrupt Acknowledge cycle

The priority can be programmed to be fixed, a specific rotation, or automatic rotation. In fixed priority mode, Interrupt Request 0 (IR0) is the highest priority followed in descending order by IR1, IR8, IR9, IR10, IR11, IR12, IR13, IR14, IR15, IR3, IR4, IR5, IR6, and IR7 (the lowest priority). This priority scheme comes from the fact that INTC2 is cascaded through IR2 of INTC1. Fixed priority is the default. Rotation or interrupt polling must be programmed. If an interrupt has its bit set in the ISR, all lower priority interrupts will not cause the assertion of INTR to the CPU (until the ISR bit is cleared). Specific Rotation allows the highest priority to be changed. All Interrupt Requests wrap-around in priority (e.g., if IR3 is selected to highest priority, the priority changes to IR3–IR7 followed by IR0-IR2 for each controller in descending order.). If Automatic Rotation is programmed, the last Interrupt Request to be serviced is given lowest priority. Polling mode will inhibit the CY82C693U from generating INTR to the CPU. The CPU must poll (read the IRR register) to determine which interrupt to take.

The Interrupt Controllers are programmed using Initialization Command Words (ICWs) and Operational Command Words (OCWs).

To initialize each controller, a sequence of four bytes must be sent to the corresponding controller. An I/O write to address 020H (for INTC1) and 0A0H (for INTC2) with 1 on bit 4 of the data bus will begin the initialization sequence. The interrupt controller will automatically reset the Initialization Word Count Register, latch ICW1 into the controller, select Fixed Priority, assign IR7 the highest priority, clear the Interrupt Mask Register, set the Slave mode address to 7 (for cascading), disable Special Mask Mode, and select the IRR for Status Read operations (the contents of the IRR will be returned on a Status Read). The next three I/O writes to address 021H (for INTC1) and 0A1H (for INTC2) will load ICW2 through ICW4 for each controller. Executing an I/O write to 020H (for INTC1) and 0A0H (for INTC2) will cause the ICW writing to terminate and will begin writing OCW2 or OCW3.

The OCWs allow the controllers to be reconfigured during normal operation. OCW1 is located at I/O address 021H (for INTC1) and 0A1H (for INTC2) whenever the controller is not being initialized. OCW2 and OCW3 are at I/O location 020H (for INTC1) and 0A0H (for INTC2) with data bit 4 set to zero. Data bit 3 controls whether OCW2 (bit3=0) or OCW3 (bit3=1) is being written.

ICW1: INTC1 Interrupt Initialization Command Word 1 (Write Only) - I/O Address=020H

Bit	Function	Default
7:5	Reserved, must be 000	000
4	Controller Initialization Control: 0: This is an Operational Command Word Write 1: This is an Initialization Command Word Write (and begins the Initialization Sequence.)	0
3	Controller Interrupt Request Control: 0: Edge-Triggered 1: Level-Sensitive If this bit is set to 1, the interrupt requests must remain active until after the Interrupt Acknowledge Cycle to return the proper vector. If the request goes away early, and no other interrupts are being requested, the vector returned will correspond to IR7. The interrupt request must be removed before the End-of-Interrupt to insure that a second spurious interrupt will not occur.	0
2	Don't Care	0
1	Controller Cascade Control: 0: Cascade Mode (for multiple interrupt controllers) 1: Single Mode (There are two controllers in the CY82C693U. Therefore, this bit should NEVER be programmed to one.)	0
0	ICW4 Write Status: 0: ICW4 write not required 1: ICW4 write required.	0

ICW2: INTC1 Interrupt Initialization Command Word 2 (Write Only) - I/O Address=021H

Bit	Function	Default
7:3	Upper 5 bits of the interrupt vector. These will appear on AD[7:3] during the data phase of an interrupt acknowledge cycle. AD[2:0] are driven with the interrupt level that comes out of the priority resolution logic (0 through 7).	00000
2:0	Interrupt Request Level, Must write to 000.	000

ICW3: INTC1 Interrupt Initialization Command Word 3 (Write Only) - I/O Address=021H

Bit	Function	Default
7	Slave Control IR7: 0: There is no slave controller connected to Interrupt Request 7 of this controller. 1: There is a slave controller connected to Interrupt Request 7 of this controller. NOTE: Must be set to 0.	0
6	Slave Control IR6: 0: There is no slave controller connected to Interrupt Request 6 of this controller. 1: There is a slave controller connected to Interrupt Request 6 of this controller. NOTE: Must be set to 0.	0
5	Slave Control IR5: 0: There is no slave controller connected to Interrupt Request 5 of this controller. 1: There is a slave controller connected to Interrupt Request 5 of this controller. NOTE: Must be set to 0.	0
4	Slave Control IR4: 0: There is no slave controller connected to Interrupt Request 4 of this controller. 1: There is a slave controller connected to Interrupt Request 4 of this controller. NOTE: Must be set to 0.	0
3	Slave Control IR3: 0: There is no slave controller connected to Interrupt Request 3 of this controller. 1: There is a slave controller connected to Interrupt Request 3 of this controller. NOTE: Must be set to 0.	0
2	Slave Control IR2: 0: There is no slave controller connected to Interrupt Request 2 of this controller. 1: There is a slave controller connected to Interrupt Request 2 of this controller. NOTE: This bit should be set to 1 for INTC2 to function properly.	0
1	Slave Control IR1: 0: There is no slave controller connected to Interrupt Request 1 of this controller. 1: There is a slave controller connected to Interrupt Request 1 of this controller. NOTE: Must be set to 0.	0
0	Slave Control IR0: 0: There is no slave controller connected to Interrupt Request 0 of this controller. 1: There is a slave controller connected to Interrupt Request 0 of this controller. NOTE: Must be set to 0.	0

ICW4: INTC1 Interrupt Initialization Command Word 4 (Write Only) - I/O Address=021H

Bit	Function	Default
7:5	Reserved, Must be set to 000	000
4	Multiple Interrupt Control: 0: Disable Multiple Interrupt from the Same Channel 1: Enable Multiple Interrupt from the Same Channel This bit (when set) allows INTC2 to fully nest interrupts, when Cascaded with Fixed Priority, without being blocked by INTC1. Correct handling requires the CPU to issue non-specific EOIs to INTC2 and check its ISR when exiting an interrupt service routine. If the ISR contains zero, a non-specific EOI should be sent to INTC1. If the ISR contains anything other than zero, no command should be issued to INTC1.	0
3	Buffered Mode: 0: Not Buffered 1: Buffered	0
2	Master/Slave stored in buffer. Must be set to 0.	0
1	Automatic End-of-Interrupt (AEOI) Control: 0: Disable AEOI 1: Enable AEOI AEOI should not be enabled if the system supports fully nested interrupts unless this controller is a Cascade Master.	0
0	Microprocessor Mode: 0: Not x86 1: x86 NOTE: This bit must be set to 1.	0

ICW1: INTC2 Interrupt Initialization Command Word 1 (Write Only) - I/O Address=0A0H

Bit	Function	Default
7:5	Reserved, Must be 000	000
4	Controller Initialization Control: 0: This is an Operational Command Word Write 1: This is an Initialization Command Word Write (and begins the Initialization Sequence.)	0
3	Controller Interrupt Request Control: 0: Edge-Triggered 1: Level-Sensitive If this is set to 1, the interrupt requests must remain active until after the Interrupt Acknowledge Cycle to return the proper vector. If the request goes away early, and no other interrupts are being requested, the vector returned will correspond to IR7. The interrupt request must be removed before the End-of-Interrupt to insure that a second spurious interrupt will not occur.	0
2	Don't Care	0
1	Controller Cascade Control: 0: Cascade Mode (for multiple interrupt controllers) 1: Single Mode (There are two controllers in the CY82C693U. Therefore, this bit should NEVER be programmed to one.)	0
0	ICW4 Write Status: 0: ICW4 write not required 1: ICW4 write required.	0

ICW2: INTC2 Interrupt Initialization Command Word 2 (Write Only) - I/O Address=0A1H

Bit	Function	Default
7:3	Upper 5 bits of the interrupt vector. These will appear on AD[7:3] during the data phase of an interrupt acknowledge cycle. AD[2:0] are driven with the interrupt level that comes out of the priority resolution logic (0 through 7).	00000
2:0	Interrupt Request Level, Must write to 000.	000

ICW3: INTC2 Interrupt Initialization Command Word 3 (Write Only) - I/O Address=0A1H

Bit	Function	Default
7:3	Reserved (These bits must be set to zero)	00000
2:0	This is the slave mode address that the controller will respond to. This should be written to 02H for proper Cascade mode operation with INTC1.	000

ICW4: INTC2 Interrupt Initialization Command Word 4 (Write Only) - I/O Address=0A1H

Bit	Function	Default
7:5	Don't Care	000
4	Multiple Interrupt Control: 0: Disable Multiple Interrupts from the Same Channel 1: Enable Multiple Interrupts from the Same Channel This bit (when set) allows INTC2 to fully nest interrupts, when Cascaded with Fixed Priority, without being blocked by INTC1. Correct handling requires the CPU to issue non-specific EOIs to INTC2 and check its ISR when exiting an interrupt service routine. If the ISR contains zero, a non-specific EOI should be sent to INTC1. If the ISR contains anything other than zero, no command should be issued to INTC1.	0
3	Buffered Mode: 0: Not Buffered 1: Buffered	0
2	Master/Slave stored in buffer. Must be set to 0.	0
1	Automatic End-of-Interrupt (AEOI) Control: 0: Disable AEOI 1: Enable AEOI AEOI should not be enabled if the system supports fully nested interrupts unless this controller is a Cascade Master.	0
0	Microprocessor Mode: 0: Not x86 1: x86 NOTE: This bit must be set to 1.	0

OCW1: INTC1 Interrupt Operational Command Word 1 (Read/Write) - I/O Address=021H

All Mask Bits are cleared by writing ICW1.

Bit	Function	Default
7	IR7 Mask Control: 0: Not Masked 1: Masked	0
6	IR6 Mask Control: 0: Not Masked 1: Masked	0
5	IR5 Mask Control: 0: Not Masked 1: Masked	0
4	IR4 Mask Control: 0: Not Masked 1: Masked	0
3	IR3 Mask Control: 0: Not Masked 1: Masked	0
2	IR2 Mask Control: 0: Not Masked 1: Masked	0
1	IR1 Mask Control: 0: Not Masked 1: Masked	0
0	IR0 Mask Control: 0: Not Masked 1: Masked	0

OCW2: INTC1 Interrupt Operational Command Word 2 (Write Only) - I/O Address=020H

Bit	Function	Default
7:5	Command Control: 000: Clear Rotate in Auto EOI 001: Non-specific EOI Command 010: No Command 011: Specific EOI Command 100: Set Rotate on AEOI 101: Rotate on non-specific EOI 110: Specific Rotate Command 111: Rotate on specific EOI	000
4	Controller Initialization Control: 0: This is an Operational Command Word Write 1: This is an Initialization Command Word Write (and begins the Initialization Sequence.)	0
3	Operational Command Word Selection Control: 0: This is a write to OCW2 1: This is a write to OCW3	0
2:0	These three bits select which interrupt channel is controlled by the specific commands. 000: Channel 0 100: Channel 4 001: Channel 1 101: Channel 5 010: Channel 2 110: Channel 6 011: Channel 3 111: Channel 7	000

OCW3: INTC1 Interrupt Operational Command Word 3 (Write Only) - I/O Address=020H

Bit	Function	Default
7	Reserved (Must be set to zero)	0
6	Set/Reset Special Mask Mode Control: 0: Disable Set/Reset of Special Mask Mode Bit 1: Enable Set/Reset of Special Mask Mode Bit This bit allows the Special Mask Mode bit (bit 5) to be protected. When this bit is disabled, bit 5 will not change.	0
5	Special Mask Mode Control: 0: Disable Special Mask Mode 1: Enable Special Mask Mode In Special Mask Mode, writing a 1 to any bit position inhibits interrupts on the associated channel. Writing a 0 to any bit position enables interrupts on the associated channel. The mask is handled by causing the priority resolution logic to ignore the condition of the ISR.	0
4	Controller Initialization Control: 0: This is an Operational Command Word Write 1: This is an Initialization Command Word Write (and begins the Initialization Sequence.)	0
3	Operational Command Word Selection Control: 0: This is a write to OCW2 1: This is a write to OCW3	0
2	Interrupt Polling Control: 0: Disable Polling Cycle 1: Enable Polling Cycle In systems where interrupts are polled, writing a 1 to this bit causes an interrupt status word to be returned on the next I/O read to the controller. Bit 7 of the data will give interrupt pending status (0 for no interrupts pending, 1 for interrupts pending). The level of the highest pending interrupt is encoded on bits 2–0. The IRR will not change until the read cycle is completed. The PM bit will automatically reset.	0
1:0	Status Control: 00: Disable Status Read 01: Disable Status Read 10: Contents of the IRR will be read on a status read. 11: Contents of the ISR will be read on a status read.	00

OCW1: INTC2 Interrupt Operational Command Word 1 (Read/Write) - I/O Address=0A1H

All Mask Bits are cleared by writing ICW1.

Bit	Function	Default
7	IR7 Mask Control: 0: Not Masked 1: Masked	0
6	IR6 Mask Control: 0: Not Masked 1: Masked	0
5	IR5 Mask Control: 0: Not Masked 1: Masked	0
4	IR4 Mask Control: 0: Not Masked 1: Masked	0
3	IR3 Mask Control: 0: Not Masked 1: Masked	0
2	IR2 Mask Control: 0: Not Masked 1: Masked	0
1	IR1 Mask Control: 0: Not Masked 1: Masked	0
0	IR0 Mask Control: 0: Not Masked 1: Masked	0

OCW2: INTC2 Interrupt Operational Command Word 2 (Write Only) - I/O Address=0A0H

Bit	Function	Default
7:5	Command Control: 000: Clear Rotate in Auto EOI 001: Non-specific EOI Command 010: No Command 011: Specific EOI Command 100: Set Rotate on AEOI 101: Rotate on non-specific EOI 110: Specific Rotate Command 111: Rotate on specific EOI	000
4	Controller Initialization Control: 0: This is an Operational Command Word Write 1: This is an Initialization Command Word Write (and begins the Initialization Sequence.)	0
3	Operational Command Word Selection Control: 0: This is a write to OCW2 1: This is a write to OCW3	0
2:0	These three bits select which interrupt channel is controlled by the specific commands. 000: Channel 0 (IRQ8) 100: Channel 4 (IRQ12) 001: Channel 1 (IRQ9) 101: Channel 5 (IRQ13) 010: Channel 2 (IRQ10) 110: Channel 6 (IRQ14) 011: Channel 3 (IRQ11) 111: Channel 7 (IRQ15)	000

OCW3: INTC2 Interrupt Operational Command Word 3 (Write Only) - I/O Address=0A0H

Bit	Function	Default
7	Reserved (Must be set to zero)	0
6	Set/Reset Special Mask Mode Control: 0: Disable Set/Reset of Special Mask Mode Bit 1: Enable Set/Reset of Special Mask Mode Bit This bit allows the Special Mask Mode bit (bit 5) to be protected. When this bit is disabled, bit 5 will not change.	0
5	Special Mask Mode Control: 0: Disable Special Mask Mode 1: Enable Special Mask Mode In Special Mask Mode, writing a 1 to any bit position inhibits interrupts on the associated channel. Writing a 0 to any bit position enables interrupts on the associated channel. The mask is handled by causing the priority resolution logic to ignore the condition of the ISR.	0
4	Controller Initialization Control: 0: This is an Operational Command Word Write 1: This is an Initialization Command Word Write (and begins the Initialization Sequence.)	0
3	Operational Command Word Selection Control: 0: This is a write to OCW2 1: This is a write to OCW3	0
2	Interrupt Polling Control: 0: Disable Polling Cycle 1: Enable Polling Cycle In systems where interrupts are polled, writing a 1 to this bit causes an interrupt status word to be returned on the next I/O read to the controller. Bit 7 of the data will give interrupt pending status (0 for no interrupts pending, 1 for interrupts pending). The level of the highest pending interrupt is encoded on bits 2–0. The IRR will not change until the read cycle is completed. The PM bit will automatically reset.	0
1:0	Status Control: 00: Disable Status Read 01: Disable Status Read 10: Contents of the IRR will be read on a status read. 11: Contents of the ISR will be read on a status read.	00

CY82C693U Timer/Counter Registers

The CY82C693U contains a timer/counter which can be used to generate a speaker tone, periodic interrupts, and the ISA refresh. There are three, individually operated, 16-bit counters. The output of counter 0 is internally hardwired to the IRQ0 input of DMA controller 1. The output of counter 1 is tied to the refresh request logic for ISA refresh. The output of counter 2 is output on the SPKR (speaker output pin).

For proper operation, each counter must be programmed before use. Each counter is programmed by writing to the "Timer Control Register" with an appropriate control word and then writing an initial count to the associated counter's register. A count value may be read by reading the associated counter's register.

Timer/Counter Register 0: Timer Control Word Register (Write Only) - Address=043H

Bit	Function	Default
7:0	Control Register	00000000

Timer Control Word Register Format (Not Read-Back Command or Counter Latch Command)

Bit	Function	Default
7:6	Counter Select: 00: Counter 0 01: Counter 1 10: Counter2 11: Read-Back Command (See Timer Control Word Register Format for Read-Back Command)	00
5:4	Read/Write Select: 00: Counter Latch Command (See Timer Control Word Register Format for Counter-Latch Command) 01: Read/Write Least Significant Byte of Counter 10: Read/Write Most Significant Byte of Counter 11: Read/Write Least Significant Byte followed by Most Significant Byte of Counter	00
3:1	Counter Mode Select: 000: Out signal on end of count=0 cycle. 001: Hardware re-triggerable one-shot. X10: Rate Generator (Divide by n counter). X11: Square Wave Output. 100: Software triggered strobe. 101: Hardware triggered strobe.	000
0	Binary/BCD Countdown Select: 0: Binary countdown 1: BCD countdown	0

Timer Control Word Register Format (Read-Back Command)

Bit	Function	Default
7:6	Counter Select: 11: Read-Back Command (Must be 11 for Read-Back Command)	11
5	Latch Count of Selected Counter(s) Control: 0: Do not latch current count value 1: Latch current count value	0
4	Latch Status of Selected Counter(s) Control: 0: Do not latch current status value 1: Latch current status value	0
3	Counter 2 Select: 0: Counter 2 not selected. 1: Counter 2 selected.	0
2	Counter 1 Select: 0: Counter 1 not selected. 1: Counter 1 selected.	0
1	Counter 0 Select: 0: Counter 0 not selected. 1: Counter 0 selected.	0
0	Reserved	0

Timer Control Word Register Format (Counter Latch Command)

Bit	Function	Default
7:6	Counter Select: 00: Latch Counter 0 01: Latch Counter 1 10: Latch Counter 2 11: Read-Back Command (See Timer Control Word Register Format for Read-Back Command)	00
5:4	Read/Write Select: 00: Counter Latch Command (Must be 00 for Counter Latch Command)	00
3:0	Reserved, Must be 0000	0000

Timer/Counter Register 1: Counter 0 Register (Read/Write Except for Read-Back Status Command) - Address=040H

Bit	Function	Default
7:0	Counter 0 Count Value	00000000

Counter 0 Register Format (Read-Back Status Command – Read Only)

Bit	Function	Default
7	Counter OUT State	0
6	Count available for reading Status: 0: The count value has not been loaded into the counting element. 1: The count value is available for reading.	0
5:4	Read/Write Status: 00: Counter Latch Command 01: Read/Write Least Significant Byte of Counter 10: Read/Write Most Significant Byte of Counter 11: Read/Write Least Significant Byte followed by Most Significant Byte of Counter	00
3:1	Counter Mode Status: 000: Out signal on end of count=0 cycle. 001: Hardware re-triggerable one-shot. X10: Rate Generator (Divide by n counter). X11: Square Wave Output. 100: Software triggered strobe. 101: Hardware triggered strobe.	000
0	Binary/BCD Countdown Status: 0: Binary countdown 1: BCD countdown	0

Timer/Counter Register 2: Counter 1 Register (Read/Write Except for Read-Back Status Command) - Address=041H

Bit	Function	Default
7:0	Counter 1 Count Value	00000000

Counter 1 Register Format (Read-Back Status Command – Read Only)

Bit	Function	Default
7	Counter OUT State	0
6	Count available for reading Status: 0: The count value has not been loaded into the counting element. 1: The count value is available for reading.	0
5:4	Read/Write Status: 00: Counter Latch Command 01: Read/Write Least Significant Byte of Counter 10: Read/Write Most Significant Byte of Counter 11: Read/Write Least Significant Byte followed by Most Significant Byte of Counter	00
3:1	Counter Mode Status: 000: Out signal on end of count=0 cycle. 001: Hardware re-triggerable one-shot. X10: Rate Generator (Divide by n counter). X11: Square Wave Output. 100: Software triggered strobe. 101: Hardware triggered strobe.	000
0	Binary/BCD Countdown Status: 0: Binary countdown 1: BCD countdown	0

Timer/Counter Register 3: Counter 2 Register (Read/Write Except for Read-Back Status Command) - Address=042H

Bit	Function	Default
7:0	Counter 2 Count Value	00000000

Counter 2 Register Format (Read-Back Status Command – Read Only)

Bit	Function	Default
7	Counter OUT State	0
6	Count available for reading Status: 0: The count value has not been loaded into the counting element. 1: The count value is available for reading.	0
5:4	Read/Write Status: 00: Counter Latch Command 01: Read/Write Least Significant Byte of Counter 10: Read/Write Most Significant Byte of Counter 11: Read/Write Least Significant Byte followed by Most Significant Byte of Counter	00
3:1	Counter Mode Status: 000: Out signal on end of count=0 cycle. 001: Hardware re-triggerable one-shot. X10: Rate Generator (Divide by n counter). X11: Square Wave Output. 100: Software triggered strobe. 101: Hardware triggered strobe.	000
0	Binary/BCD Countdown Status: 0: Binary countdown 1: BCD countdown	0

CY82C693U Real-Time-Clock Registers

The RTC inside the CY82C693U contains a time-of-day clock, an interrupt generating alarm, a 100 year calendar, a software controlled periodic interrupt, and 242 bytes of battery-backable static RAM (for scratch data). An external battery and a 32.768-kHz crystal are all that must be provided to enable the RTC to keep time in battery-backed mode. Leap-year and daylight savings time will be automatically adjusted.

RTC registers are accessed using I/O addresses 070H and 071H. The index address of the register must first be placed in the RTC's Index Address Register. This is loaded through an I/O write to address 070H with the register index as the write data. Following the I/O write to 070H, an I/O access (read or write) to address 071H will read or write the contents of the appropriate RTC register.

The scratch data is broken into two separate blocks. Access to the blocks is controlled through I/O accesses to different addresses. The lower 128 register bytes (this includes the Clock Data) are accessed using I/O writes to address 070H (with register index as the write data) followed by I/O reads or writes to address 071H. The data for the I/O access to 071H is the contents of the desired register. For the upper 128 scratch RAM bytes, the scratch RAM address should be written to I/O address 072H followed by a data read or write to I/O address 073H.

PLEASE NOTE: Where the registers are called out BCD (binary coded decimal), the values must be written/read as BCD values (Not Hexadecimal).

RTC Register 0: Seconds Byte (Read/Write except for bit 7 which is always 0) – Index=00H

Bit	Function	Default
7:0	BCD representation of seconds (must be in the range 00–59)	00000000

RTC Register 1: Seconds Alarm (Read/Write) – Index=01H

Bit	Function	Default
7:0	BCD representation of seconds alarm (must be in the range 00–59) If bit 7:6=11, then a 1 second periodic interrupt will occur.	00000000

RTC Register 2: Minutes Byte (Read/Write) – Index=02H

Bit	Function	Default
7:0	BCD representation of minutes (must be in the range 00–59)	00000000

RTC Register 3: Minutes Alarm (Read/Write) – Index=03H

Bit	Function	Default
7:0	BCD representation of minutes alarm (must be in the range 00–59)	00000000

RTC Register 4: Hours Byte (Read/Write) – Index=04H

Bit	Function	Default
7:0	BCD representation of hours 12-hour Mode: Must be in the range 01–12 (AM) or 81–92 (PM) 24-hour Mode: Must be in the range 00–23	00000000

RTC Register 5: Hours Alarm (Read/Write) – Index=05H

Bit	Function	Default
7:0	BCD representation of hours alarm 12-hour Mode: Must be in the range 01–12 (AM) or 81–92 (PM) 24-hour Mode: Must be in the range 00–23	00000000

RTC Register 6: Day-of-the-Week Byte (Read/Write) – Index=06H

Bit	Function	Default
7:0	BCD representation of day-of-the-week (must be in the range 01–07); 01=Sunday.	00000000

RTC Register 7: Day-of-the-Month Byte (Read/Write) – Index=07H

Bit	Function	Default
7:0	BCD representation of day-of-the-month (must be in the range 01–31)	00000000

RTC Register 8: Month Byte (Read/Write) – Index=08H

Bit	Function	Default
7:0	BCD representation of month (must be in the range 01–12)	00000000

RTC Register 9: Year Byte (Read/Write) – Index=09H

Bit	Function	Default
7:0	BCD representation of year (must be in the range 00–99)	00000000

RTC Register 10: Control/Status Register A (Read/Write except for bit 7 which is Read Only) – Index=0AH

Bit	Function	Default
7	Update In Progress (UIP): 0: There is no clock update about to occur (guaranteed for 244 µs) 1: An update cycle is about to occur or is occurring. Data read from the clock words will be invalid. This bit goes high 244 µs prior to an update and remains active for an additional 2 ms after the update begins. This bit can be cleared by writing a 1 to bit 7 of RTC register 11.	0
6:4	Divider/Prescaler on the RTC: 000: 4.194304 MHz Oscillator 001: 1.048576 MHz Oscillator 010: 32.768 kHz Oscillator (Recommended Setting) 1XX: Reset Divider	000
3:0	Periodic Interrupt Rate Control: 32.768KHz Oscillator / Other Frequencies 0000: No Periodic Interrupts / No Periodic Interrupts 0001: 3.90526 ms / 30.517 µs 0010: 7.8125 ms / 61.035 µs 0100: 122.070 µs / 122.070 µs 0101: 244.141 µs / 244.141 µs 0110: 488.281 µs / 488.281 µs 0111: 976.562 µs / 976.562 µs 1000: 1.953125 ms / 1.953125 ms 1001: 7.8125 ms / 7.8125 ms 1010: 15.625 ms / 15.625 ms 1011: 31.25 ms / 31.25 ms 1100: 62.5 ms / 62.5 ms 1101: 125 ms / 125 ms 1110: 250 ms / 250 ms 1111: 500 ms / 500 ms	0000

RTC Register 11: Control/Status Register B (Read/Write) – Index=0BH

Bit	Function	Default
7	Update Cycle Control: 0: Enable Update Cycles 1: Disable Update Cycles, and abort any Update Cycle In Progress This bit is not changed when $\overline{\text{PSRSTB}}$ is asserted.	0
6	Periodic Interrupt Control: 0: Disable Periodic Interrupt Generation 1: Enable Periodic Interrupt Generation This bit is cleared when $\overline{\text{PSRSTB}}$ is asserted.	0
5	Alarm Interrupt Control: 0: Disable Alarm Interrupt Generation 1: Enable Alarm Interrupt Generation Alarm interrupts will be generated when the time matches the values programmed into the alarm register fields. This bit is cleared when $\overline{\text{PSRSTB}}$ is asserted.	0
4	Update-Ended Interrupt Control: 0: Disable Update-Ended Interrupt Generation 1: Enable Update-Ended Interrupt Generation Update-Ended interrupts will be generated when an update cycle is completed. This can be used to tell the CPU that its data may be read from the clock data registers. This bit is cleared when $\overline{\text{PSRSTB}}$ is asserted.	0
3	Square Wave Control: 0: Disable Square Wave Generation 1: Enable Square Wave Generation	0
2	Data Mode: 0: BCD 1: Binary	0
1	Hour Format Control: 0: 12-hour clock format 1: 24-hour clock format This bit is not changed when $\overline{\text{PSRSTB}}$ is asserted.	0
0	Daylight Savings Time Control: 0: Disable Daylight Savings Time Change 1: Enable Daylight Savings Time Change Setting this bit will cause the update cycle to change the time automatically for Daylight Savings Time. The changes take place on the last Sunday in April and the last Sunday in October. This bit is not changed when $\overline{\text{PSRSTB}}$ is asserted.	0

RTC Register 12: Control/Status Register C (Read Only) – Index=0CH

Bit	Function	Default
7	Interrupt Request Flag: 0: No RTC Interrupts have occurred. 1: An RTC Interrupt has occurred. The specific interrupt flags (bits 6–4 of this register) and the interrupt enable bits (bits 6–4 of RTC Register 11) indicate which event(s) caused the interrupt to occur. This bit is cleared by the assertion of $\overline{\text{PSRSTB}}$ or a read to this register.	0
6	Periodic Interrupt Flag: 0: The period timer has not reached its terminal count. 1: The period timer has reached its terminal count Note: An interrupt will not be generated and bit 7 will not be set if Periodic Interrupts are disabled. This bit is cleared by the assertion of $\overline{\text{PSRSTB}}$ or a read to this register.	0
5	Alarm Interrupt Flag: 0: The clock did not match the alarm values. 1: The clock did match the alarm values. Note: An interrupt will not be generated and bit 7 will not be set if Alarm Interrupts are disabled. This bit is cleared by the assertion of $\overline{\text{PSRSTB}}$ or a read to this register.	0
4	Update-Ended Interrupt Flag: 0: An Update Cycle did not end 1: An Update Cycle did end Note: An interrupt will not be generated and bit 7 will not be set if Update-Ended Interrupts are disabled. This bit is cleared by the assertion of $\overline{\text{PSRSTB}}$ or a read to this register.	0
3:0	Reserved (Read as all zeroes).	0000

RTC Register 13: Control/Status Register D (Read Only) – Index=0DH

Bit	Function	Default
7	Valid RAM and Time Flag: 0: The contents of the RAM and Time are not guaranteed to be valid 1: The contents of the RAM and Time are valid	0
6:0	Reserved (Read as all zeroes).	0000000

RTC Registers 14-127: Battery-Backable Scratch Block 1 (Read/Write) – Indices=0EH-7FH

Bit	Function	Default
7:0	User Defined	00000000

RTC Registers 128-255: Battery-Backable Scratch Block 2 (Read/Write) – Indices=80H-FFH

NOTE: These Registers are accessed through I/O ports 072H and 073H (not 070H and 071H-the ports for the other RTC registers)

Bit	Function	Default
7:0	User Defined	00000000

CY82C693U Keyboard/Mouse Controller Registers

The Keyboard controller inside the CY82C693U is capable of supporting both AT and PS/2 modes of operation. In PS/2 mode, a PS/2 style mouse is supported without the need of an external controller.

The keyboard controller processes scan codes whenever a keyboard key is pressed and released. When a key is pressed, a “make” code is sent from the controller in the keyboard to the controller in the CY82C693U. If the key is not released immediately, “make” codes are continually sent to the system. When the key is finally released, a “break” is recognized by the system.

Special function keys (such as the <Shift> key or the <Cntrl> key) are used to set status flags inside the keyboard controller. The status flags can alter the function of other keystrokes.

The keyboard controller interrupts the processor through IRQ1 and the mouse controller interrupts the processor through IRQ12. These interrupts will cause the operating systems to enter their scan code interpretation routines.

The keyboard/mouse controller control registers are accessed using I/O port locations 64H and 60H. Status is read, by performing a read to I/O port 64H. The keyboard controller’s input buffer may be addressed at I/O port 60H or 64H (Data bytes are written to 60H and command bytes are written to 64H). The output buffer can contain scan codes from the keyboard or any requested data bytes from the controller. It can be accessed by reading I/O port 60H.

Keyboard/Mouse Register 0: Status Register (Read Only) - I/O Read to address 64H

Bit	Function	Default
7	Parity Error Detection: 0: No parity error was transmitted from the keyboard 1: A parity error was detected in the last byte transmitted. The keyboard must transmit odd parity.	0
6	Timeout Detection: 0: No timeout was detected 1: A timeout occurred during the transmission of data between the controller and the keyboard.	0
5	Mouse Output Buffer Full Flag: 0: No mouse data is present in the output buffer. 1: Mouse data is present in the output buffer.	0
4	Keyboard Lock Status: 0: KEYLOCK is active. The keyboard controller is inhibited from receiving scan codes. 1: KEYLOCK is inactive. Normal operation.	0
3	Command/Data Flag: 0: Data byte has been written to 60H. 1: command byte has been written to 64H. This bit can be used to indicate to the system whether the byte in the input buffer is a command byte or a data byte.	0
2	System Flag: This bit will reflect the value written to the Set System Flag bit of the command byte.	0
1	Input Buffer (Port 60H or 64H) Full Flag: 0: No data is present in the input buffer. 1: Data is present in the input buffer. A read of the input buffer will reset this bit to zero.	0
0	Output Buffer (Port 60H only) Full Flag: 0: No data is present in the output buffer. 1: Data is present in the output buffer. A read of the output buffer will reset this bit to zero.	0

Keyboard/Mouse Register 1: Command Byte Register. Stored in keyboard RAM at location 20H; A 20H command must be written to the input buffer at 64H, Data is accessed at 60H.

Bit	Function	Default
7	Reserved. Must be set to zero.	0
6	Scan Code Translation: 0: Do not translate scan codes. 1: Scan codes are translated to AT scan codes.	0
5	Mouse Interface Control: 0: Enable PS/2 mouse interface. 1: Disable PS/2 mouse interface.	0
4	Keyboard Interface Control: 0: Enable keyboard interface. 1: Disable keyboard interface. When the keyboard interface is disabled, the KBCLK signal is driven LOW. Nothing can be sent or received.	0
3	KEYLOCK Override Control: 0: Normal Operation. 1: Override the KEYLOCK signal. This will cause the keyboard to be unlocked regardless of the state of the KEYLOCK signal.	0
2	System Flag: 0: Report 0 on bit 2 of Status Register during status reads. 1: Report 1 on bit 2 of Status Register during status reads.	0
1	Output Buffer Filled with Mouse Data Interrupt Control: 0: Will not send IRQ12 whenever the output buffer is filled with mouse data. 1: Will send IRQ12 whenever the output buffer is filled with mouse data.	0
0	Output Buffer Filled with Keyboard Data Interrupt Control: 0: Will not send IRQ1 whenever the output buffer is filled with keyboard data. 1: Will send IRQ1 whenever the output buffer is filled with keyboard data.	0

System to Controller Command Set. These commands are executed by writing them to the input register through I/O port 064H. If data is accompanied with the command, it must be read/written to I/O port 060H.

Command	Description
20H	Read controller's command byte: The controller will put the contents of the current command byte into the output port. The command byte format is given above (Keyboard/Mouse Register 1).
21H-3FH	Read controller's RAM: The controller will put the contents of the controller's RAM addressed by bits 5 through 0 of the command into the output port.
60H	Write controller's command byte: The controller will write the new command byte. It uses the next data written to I/O port 60H for the new command byte data. The command byte format is given above (Keyboard/Mouse Register 1).
61H-7FH	Write controller's RAM: The controller will write the RAM location addressed by bits 5 through 0 of the command. It uses the next data written to I/O port 60H for the new RAM data.
A4H	Check for installed password: The controller will test for a password. If a password is installed, FAH will be placed in the output buffer. If no password is installed, F1H is placed in the output buffer.
A5H	Load password: The keyboard controller will load the contents of the input buffer into a password location and will not stop storing the password until NULL (00H) is detected. NULL will be stored as the last byte of the password.
A6H	Enable Password Security: The keyboard controller will check the installed password against the keyboard input for a match before allowing scan codes to be passed to the system.
A7H	Disable Mouse interface: This command will set bit 5 of the command byte to 1. This will cause the MSECLK signal to be driven low, effectively disabling mouse operation.
A8H	Enable Mouse interface: This command will reset bit 5 of the command byte to 0. Mouse operation will be allowed.
A9H	Mouse interface test: The controller will test the MSECLK and MSEDATA signals. The following status will be placed in the output buffer: 00H: No Error Detected 01H: MSECLK is stuck low. 02H: MSECLK is stuck high. 03H: MSEDATA is stuck low. 04H: MSEDATA is stuck high.
AAH	Execute Self-Test: The internal diagnostic self-test will be executed. If no errors are detected, 55H will be placed in the output buffer.
ABH	Keyboard interface test: The controller will test the KBCLK and KBDATA signals. The following status will be placed in the output buffer: 00H: No Error Detected 01H: KBCLK is stuck low. 02H: KBCLK is stuck high. 03H: KBDATA is stuck low. 04H: KBDATA is stuck high.
ADH	Disable Keyboard interface: This command will set bit 4 of the command byte to 1. This will cause the KBCLK signal to be driven low, effectively disabling keyboard operation.
AEH	Enable Keyboard interface: This command will reset bit 4 of the command byte to 0. Keyboard operation will be allowed.
C0H	Read controller's input port: This command will cause the controller to read its input port and place the contents into the controller's output buffer.

System to Controller Command Set. (continued) These commands are executed by writing them to the input register through I/O port 064H. If data is accompanied with the command, it must be read/written to I/O port 060H.

D0H	Read controller's output port: This command will cause the controller to read its output port and place the contents into the controller's input buffer.
D1H	Write controller's output port: This command will cause the controller to take the next data that is written to 60H and place it into the controller's output port.
D2H	Write keyboard output buffer: This command will cause the controller to take the next data that is written to 60H and place it into the controller's output buffer.
D3H	Write mouse output buffer: This command will cause the controller to take the next data that is written to 60H and place it into the controller's mouse output buffer.
D4H	Write to mouse: This command will cause the controller to take the next data that is written to 60H and transmit it to the mouse.
E0H	Read Test Inputs: T0 and T1 (KBCLK and KBDATA inputs) are read by the controller and placed in the output buffer. Bit 0 represents T0 and bit 1 represents T1.
EXH	Active Output Port Flag: Bits 1, 2, and 3 of the output buffer represent the active output port.
F0H–FFH	Pulse Output Port: This command will cause bits 3 through 0 in the controller's output port to pulse low for 6us based on the command. A 0 in bits 3 through zero will select the bits to be pulsed.

Controller to System Command Set. These commands are written from the controller to the output register. They must be read by the system performing an I/O read to address 60H.

Command	Description
55H	No error detected upon completion of self-test: This command will appear in the output register in response to a successful completion of the self-test initiated by system command AAH.
F1H	No password installed: This command will appear in the output register in response to no password detection initiated by system command A4H.
FAH	Password installed: This command will appear in the output register in response to a password detection initiated by system command A4H.
FEH	Resend Command: This command will appear in the output register in response to an illegal command.

System to Keyboard Command Set. These commands are written from the controller to the keyboard. The command should be written to the Input Buffer (Port 60H). If data is required along with the command, the system must perform a subsequent data write to I/O Port 60H.

Command	Description
EDH	Set/Reset Keyboard LED status indicators: Num Lock, Caps Lock, and Scroll Lock LEDs, which are found on most keyboards, can be turned on or off using this command. The system must first write the EDH command to I/O port 60H and get acknowledged. Then the system must write port 60H with the following data: Bits 7–3: 00000 Bit 2: Caps Lock (1 Turns LED on) Bit 1: Num Lock (1 Turns LED on) Bit 0: Scroll Lock (1 Turns LED on)
EEH	Echo Command: This command can be used to test the keyboard. A present and working keyboard will echo EEH after it receives this command.
F2H	Read keyboard ID bytes: The keyboard will acknowledge the command and send the two keyboard ID bytes.
F3H	Set typematic rate and delay period: This command is used to set the typematic rate (make codes/second when a key is pressed and held down) and delay period (number of milliseconds after a key is pressed before typematic starts). INT 16H Function 03H sends the Set Rat1e command F3H to the keyboard controller followed by the delay and rate values.
F4H	Begin Operation: This command causes the keyboard to clear its output buffer and begin scanning.
F5H	Reset and Disable: This command causes the keyboard to reset to its power-on default and disables scanning.
F6H	Reset and Enable: This command causes the keyboard to reset to its power-on default and enables scanning.
FEH	Resend: If the system detects an error in the previous transmission, this command will cause the keyboard to send the transmission again.
FFH	Self-test: This command causes the keyboard to initiate its internal diagnostic self-test.

Keyboard to System Command Set. These commands are written from the keyboard to the controller.

Command	Description
00H	Overrun character (For scan code sets 2 and 3): When the keyboard exceeds buffer capacity, it places 00H at the bottom of the buffer. When 00H reaches the top of the buffer, it is sent to the system.
83H	Low Keyboard ID Byte: The keyboard responds to the Read ID bytes command by first sending the low ID byte.
ABH	High Keyboard ID Byte: The keyboard responds to the Read ID bytes command by first sending the low ID byte. Then it sends the high ID byte.
AAH	Power-on Diagnostics Completed: If the power-on diagnostic tests completed successfully (no error detected), this command is sent to the system.
FCH	Power-on Diagnostics Failed: If the power-on diagnostic tests detected an error, this command is sent to the system. The keyboard controller will wait to be reset or receive some other command from the system.
EEH	Echo Response: This is the keyboard's response to the system's echo command.
FAH	Acknowledge (ACK): The keyboard will acknowledge any valid keyboard command that the system transmits with an ACK command.
FEH	Resend: If the keyboard detects an error in the previous transmission, this command will be sent, requesting the system to send the transmission again.
FFH	Overrun character (For scan code set 1): When the keyboard exceeds buffer capacity, it places FFH at the bottom of the buffer. When FFH reaches the top of the buffer, it is sent to the system.

System to Mouse Controller Command Set. These commands are specific to the mouse controller. The system must first send command D4H to I/O port 64H to tell the controller that the next command is intended for the mouse controller. If D4H is not sent, the controller will interpret the commands as Keyboard commands.

Command	Description
E6H	Reset mouse scaling: This command will cause scaling to be reset to 1:1.
E7H	Set scaling: When the mouse is in Stream Mode, this command will convert Input X,Y coordinates to outputs as follows: Input — Output 0 — 0 1 — 1 2 — 1 3 — 3 4 — 6 5 — 9 6 or greater — Input x 2
E8H	Set Resolution: This is a two byte command. The first byte is E8H (sent to port 64H) to tell the mouse controller to set its resolution. The second byte is data (sent to port 60H). The data must be in one of the following combinations: Data Byte Value — Resolution 00H — 1 count/mm 01H — 2 counts/mm 02H — 4 counts/mm 03H — 8 counts/mm
E9H	Status Request: In response to this command, the mouse will send a three byte status report. The first byte is the sampling rate. The second byte is the resolution, and the third byte is defined as follows: Bit — Description 7 — Reserved 6 — 0: Stream Mode 1: Remote Mode 5 — 0: Mouse Disabled 1: Mouse Enabled 4 — 0: Scaling 1:1 1: Scaling 2:1 (Defined by Set Scaling command) 3 — Reserved 2 — 0: Left mouse button not pressed 1: Left mouse button pressed 1 — Reserved 0 — 0: Right mouse button not pressed 1: Right mouse button pressed.
EAH	Set Stream Mode: When Stream Mode is selected, the mouse transmits data every time the mouse detects a unit of movement or a mouse button pressed. If no button is pressed or the mouse does not detect movement, no data will be transmitted.
EBH	Read Mouse Data: This command will force the mouse to transmit one packet of mouse data.
ECH	Reset Wrap Mode: This command takes the mouse out of wrap mode (the mouse will not echo data back to the system).
EEH	Set Wrap Mode: This command puts the mouse in wrap mode (the mouse will echo data back to the system).
F0H	Set Remote Mode: In Remote Mode, mouse data will only be transmitted in response to a Read Mouse Data command.
F2H	Read device type: The mouse will return 00H (type ID for mouse) in response to this command.

System to Mouse Controller Command Set. (continued) These commands are specific to the mouse controller. The system must first send command D4H to I/O port 64H to tell the controller that the next command is intended for the mouse controller. If D4H is not sent, the controller will interpret the commands as Keyboard commands.

F3H	Set Sampling Rate: This is a two byte command. The first byte is F3H (sent to port 64H) to tell the mouse controller to set its sampling rate (the number of times per second that the system checks the mouse for data). The second byte is data (sent to port 60H). The data must be in one of the following combinations: Data Byte Value — Sampling Rate 0AH — 10 samples/second 14H — 20 samples/second 28H — 40 samples/second 3CH — 60 samples/second 50H — 80 samples/second 64H — 100 samples/second C8H — 200 samples/second
F4H	Enable transmission: This command will enable data transmission if the mouse has been set to Stream Mode. This command has no effect if the controller is in remote mode.
F6H	Load default settings: This command will load the power-on default settings. They are as follows: 100 samples/second sampling rate Linear Scaling Stream Mode 4 counts/mm resolution Transmission disabled
FEH	Resend: If the system detects an error in the previous transmission, this command will cause the mouse to send the transmission again.
FFH	Self-test: This command causes the mouse to initiate its internal diagnostic self-test.

Mouse to System Controller Command Set. These commands are transferred from the mouse to the mouse controller.

Command	Description
FAH	Acknowledge (ACK) Command: The mouse will acknowledge valid commands by returning the ACK command (except for reset commands).
FEH	Resend: If the mouse detects an error in the previous transmission, this command will cause the mouse controller to send the transmission again.

CY82C693U PCI Configuration Registers

The PCI configuration registers for the CY82C693U are defined in this section. The registers are accessed by performing configuration read and write cycles (C/$\overline{BE}$[3:0]=1010b or 1011b) with the IDSEL signal asserted to the CY82C693U.

PCI to ISA PCI Configuration Registers (Function 0 during Configuration Cycle)

Register 0: Vendor ID Number (Read Only) – Index=00H with a 16-bit access

Bit	Function	Default
15:0	Cypress ID Number: 0001000010000000	1080H

Register 1: Device ID Number (Read Only) – Index=02H with a 16-bit access

Bit	Function	Default
15:0	CY82C693U Device ID Number: 1100011010010011	C693H

Register 3: Command Register (Read/Write) – Index=04H with a 16-bit access

Bit	Function	Default
15:10	Reserved	000000
9	Fast Back-to-Back Enable. This bit is forced to 0 (Fast Back-to-Back Accesses are Disabled)	0
8	$\overline{SERR}$ Reporting Enable: 0: $\overline{SERR}$ Reporting Disabled 1: $\overline{SERR}$ Reporting Enabled	0
7:4	Reserved	0000
3	Enable Special Cycle Decoding: 0: Disable Special Cycle Decoding 1: Enable Special Cycle Decoding	0
2	Bus Master Enable. This bit is forced to 1 (Bus Mastering is Always Enabled)	1
1	Memory Access Enable. This bit is forced to 1 (Memory Access is Always Enabled)	1
0	I/O Access Enable. This bit is forced to 1 (I/O Access is Always Enabled)	1

Register 4: Status Register (Read/Write) – Index=06H with a 16-bit access

Bit	Function	Default
15	Not implemented: Read as 0.	0
14	PCI System Error READ: 0: No PCI System Error Occurred 1: PCI System Error Occurred WRITE: 0: No change to register. 1: Clear Register	0
13	PCI Master-Abort READ: 0: No PCI Master-Abort Occurred 1: PCI Master-Abort Occurred WRITE: 0: No change to register. 1: Clear Register	0
12	CY82C693U Detection of Target-Abort (from another PCI target) READ: 0: No PCI Target-Abort Occurred 1: PCI Target-Abort Occurred WRITE: 0: No change to register. 1: Clear Register	0
11	CY82C693U Assertion of Target-Abort (CY82C693U is the target) READ: 0: No PCI Target-Abort Occurred 1: PCI Target-Abort Occurred WRITE: 0: No change to register. 1: Clear Register	0
10:9	$\overline{\text{DEVSEL}}$ Timing Status. These register bits are Read Only and will always return 01 (medium timing)	01
8	Not implemented: Read as 0.	0
7	Fast Back-to-Back Transfer Occurred. This Read Only bit will always return 1.	1
6:0	Reserved	0000000

Register 5: Revision ID Number (Read Only) – Index=08H with an 8-bit access

Bit	Function	Default
7:0	Current Revision of the Part (00000000)	00H

Register 6: Programming Interface Revision ID Number (Read Only) – Index=09H with an 8-bit access

Bit	Function	Default
7:0	Current Revision of the Programming Interface (00000000)	00H

Register 7: Sub Class Code Register (Read Only) – Index=0AH with an 8-bit access

Bit	Function	Default
7:0	Sub Class Code (00000001)	01H

Register 8: Base Class Code Register (Read Only) – Index=0BH with an 8-bit access

Bit	Function	Default
7:0	Base Class Code (00000110)	06H

Register 9: Header Type Register (Read Only) – Index=0EH with an 8-bit access

Bit	Function	Default
7:0	Device Type - multi-function device	80H

Register 10: PCI Interrupt A Routing Control Register (Read/Write) – Index=40H with an 8-bit access

Bit	Function	Default
7	Interrupt A Routing Control: 0: Interrupt A Routing Enabled 1: Interrupt A Routing Disabled	0
6:4	Reserved	000
3:0	Interrupt Request Level that PCI $\overline{INTA}$ is Routed to: 0000: Reserved 1000: Reserved 0001: Reserved 1001: IRQ9 0010: Reserved 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 1101: Reserved 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15	0000

Register 11: PCI Interrupt B Routing Control Register (Read/Write) – Index=41H with an 8-bit access

Bit	Function	Default
7	Interrupt B Routing Control: 0: Interrupt B Routing Enabled 1: Interrupt B Routing Disabled	0
6:4	Reserved	000
3:0	Interrupt Request Level that PCI $\overline{INTB}$ is Routed to: 0000: Reserved 1000: Reserved 0001: Reserved 1001: IRQ9 0010: Reserved 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 1101: Reserved 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15	0000

Register 12: PCI Interrupt C Routing Control Register (Read/Write) – Index=42H with an 8-bit access

Bit	Function	Default
7	Interrupt C Routing Control: 0: Interrupt C Routing Enabled 1: Interrupt C Routing Disabled	0
6:4	Reserved	000
3:0	Interrupt Request Level that PCI $\overline{INTC}$ is Routed to: 0000: Reserved 1000: Reserved 0001: Reserved 1001: IRQ9 0010: Reserved 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 1101: Reserved 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15	0000

Register 13: PCI Interrupt D Routing Control Register (Read/Write) – Index=43H with an 8-bit access

Bit	Function	Default
7	Interrupt D Routing Control: 0: Interrupt D Routing Enabled 1: Interrupt D Routing Disabled	0
6:4	Reserved	000
3:0	Interrupt Request Level that PCI $\overline{INTD}$ is Routed to: 0000: Reserved 1000: Reserved 0001: Reserved 1001: IRQ9 0010: Reserved 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 1101: Reserved 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15	0000

Register 14: PCI Control Register (Read/Write) – Index=44H with an 8-bit access

Bit	Function	Default
7	Reserved	0
6:5	PCI Master Grant Arbitration Time 00: Reevaluate Grants after every PCI clock 01: Reevaluate Grants after 16 PCI clocks 10: Reevaluate Grants after 32 PCI clocks 11: Reevaluate Grants after 64 PCI clocks	0
4	Retry on excessive delay control: 0: Disable Retry if delay will be longer than 16 PCI clock cycles 1: Enable Retry if delay will be longer than 16 PCI clock cycles (Rev 2.1 Compliant)	0
3	Reserved	0
2	PCI Post-Write Buffer Control: 0: Disable PCI Post-Write Buffers 1: Enable PCI Post-Write Buffers	0
1	ISA Master Line Buffer Control: 0: ISA Master transfers will cause PCI single (8/16-bit) transfers 1: ISA Master transfers will cause PCI 8-byte transfers (pre-read 8 bytes)	0
0	DMA Line Buffer Control: 0: DMA transfers will cause PCI single (8/16-bit) transfers 1: DMA transfers will cause PCI 8-byte transfers (pre-read 8 bytes)	0

Register 15: PCI Error Control Register (Read/Write) – Index=45H with an 8-bit access

Bit	Function	Default
7	$\overline{\text{SERR}}$ Reporting on Target-Abort Control: 0: Disable $\overline{\text{SERR}}$ Assertion on Target-Abort 1: Enable $\overline{\text{SERR}}$ Assertion on Target-Abort	0
6	$\overline{\text{SERR}}$ Reporting on Special Cycle Data Parity Errors Control: 0: Disable $\overline{\text{SERR}}$ Assertion on Special Cycle Data Parity Errors 1: Enable $\overline{\text{SERR}}$ Assertion on Special Cycle Data Parity Errors	0
5	$\overline{\text{SERR}}$ Reporting on Address Parity Errors Control: 0: Disable $\overline{\text{SERR}}$ Assertion on Address Parity Errors 1: Enable $\overline{\text{SERR}}$ Assertion on Address Parity Errors	0
4:2	Reserved	000
1	ISA Master Post-Write Enable: 0: Disable ISA Master Post-Write Buffering 1: Enable ISA Master Post-Write Buffering	0
0	DMA Post-Write Enable: 0: Disable DMA Post-Write Buffering 1: Enable DMA Post-Write Buffering	0

Register 16: PCI Error Status Register (Read/Write) – Index=46H with an 8-bit access

Bit	Function	Default
7:5	Reserved	000
4	Shutdown Cycle READ: 0: No Shutdown Cycle Detected 1: Shutdown Cycle Detected WRITE: 0: No change to register. 1: Clear Register	0
3:2	Reserved	00
1	ISA/DMA 0–512K Forwarding: 0: Enable Forwarding 1: Disable Forwarding	0
0	ISA/DMA 640K–768K Forwarding: 0: Disable Forwarding 1: Enable Forwarding	0

Register 17: PCI BIOS Control Register (Read/Write) – Index=47H with an 8-bit access

Bit	Function	Default
7	Extended BIOS Control (Addresses FFF80000H–FFFDFFFFH): 0: Disable ROM Address Mapping from FFF80000H to FFFDFFFFH 1: Generate $\overline{\text{ROMCS}}$ for addresses between FFF80000H–FFFDFFFFH	0
6	Low BIOS Control (Addresses 000E0000H–000EFFFFH and FFFE0000H–FFFEFFFFH): 0: Disable Low ROM Address Mapping 1: Generate $\overline{\text{ROMCS}}$ for Low ROM addresses	0
5	ROM Write-protect Control: 0: $\overline{\text{ROMCS}}$ will only become active on ROM reads (ROM is Read Only). 1: $\overline{\text{ROMCS}}$ will become active on ROM reads and writes.	0
4	Block D BIOS Control (Addresses 000D0000H–000DFFFFH): 0: Disable ROM Address Mapping from 000D0000H–000DFFFFH 1: Generate $\overline{\text{ROMCS}}$ for addresses between 000D0000H–000DFFFFH	0
3	Block C BIOS Control (Addresses 000C0000H–000CFFFFH): 0: Disable ROM Address Mapping from 000C0000H–000CFFFFH 1: Generate $\overline{\text{ROMCS}}$ for addresses between 000C0000H–000CFFFFH	0
2	External Keyboard Controller Status (This bit is Read Only): 0: Internal Keyboard Controller is in use. 1: External Keyboard Controller is in use (Internal Keyboard Controller is Disabled)	0
1	External Real-Time-Clock Status (This bit is Read Only): 0: Internal RTC is in use. 1: External RTC is in use (Internal RTC is Disabled)	0
0	External XD Buffer Status (This bit is Read Only): 0: CY82C693U XD Bus is in use. 1: External buffer is in use to buffer the XD bus.	0

Register 18: ISA/DMA Top of Memory Control (Read/Write) – Index=48H with an 8-bit access

Bit	Function	Default
7:4	Top of ISA Memory: 0000: 1 MByte 1000: 9 MByte 0001: 2 MByte 1001: 10 MByte 0010: 3 MByte 1010: 11 MByte 0011: 4 MByte 1011: 12 MByte 0100: 5 MByte 1100: 13 MByte 0101: 6 MByte 1101: 14 MByte 0110: 7 MByte 1110: 15 MByte 0111: 8 MByte 1111: 16 MByte	0
3	ISA/DMA Low BIOS Forwarding Control 0: Disable ISA/DMA Low BIOS Forwarding 1: Enable ISA/DMA Low BIOS Forwarding	0
2	ISA/DMA 512K-640K BIOS Forwarding Control 0: Disable ISA/DMA 512K–640K BIOS Forwarding 1: Enable ISA/DMA 512K–640K BIOS Forwarding	0
1	ISA/DMA 832K-896K BIOS Forwarding Control 0: Disable ISA/DMA 832K–896K BIOS Forwarding 1: Enable ISA/DMA 832K–896K BIOS Forwarding	0
0	ISA/DMA 768K-832K BIOS Forwarding Control 0: Disable ISA/DMA 768K–832K BIOS Forwarding 1: Enable ISA/DMA 768K–832K BIOS Forwarding	0

Register 19: AT Control Register #1 (Read/Write) – Index=49H with an 8-bit access

Bit	Function	Default
7	Reserved	0
6	PIO IDE Routable Interrupt Request Control 0: IDE Interrupt Requests are available on XD bus pins to support Interrupt routing 1: IDE Interrupt Requests must be hardwired to IRQ14 and IRQ15.	0
5	IDE DMA Status (This bit is Read Only) 0: IDE DMA is disabled 1: IDE DMA is enabled	0
4	Keyboard Controller Status (This bit is Read Only) 0: PS/2 Keyboard Controller 1: Standard AT Keyboard Controller	0
3	16-bit I/O Recovery Control 0: Disable 16-bit I/O Recovery 1: Enable 16-bit I/O Recovery	0
2	8-bit I/O Recovery Control 0: Disable 8-bit I/O Recovery 1: Enable 8-bit I/O Recovery	0
1:0	ATCLK Control 00: PCICLK divided by four 01: PCICLK divided by three 10: 14.318 MHz Clock divided by two (7.16 MHz) 11: Reserved	00

Register 20: AT Control Register #2 (Read/Write) – Index=4AH with an 8-bit access

Bit	Function	Default
7	Wait for Halt Control: 0: Enable Wait for Halt 1: Disable Wait for Halt	0
6	Keyboard Controller Fast Reset Emulation Control: 0: Enable Fast Reset Emulation 1: Disable Fast Reset Emulation	0
5	Keyboard Controller Fast Gate A20 Emulation Control: 0: Enable Fast Gate A20 Emulation 1: Disable Fast Gate A20 Emulation	0
4	Port 92 Emulation Control: 0: Enable Port 92 Emulation 1: Disable Port 92 Emulation	0
3	AT Refresh Cycle Control 0: Refresh is four ATCLK cycles 1: Refresh is four ATCLK/2 cycles	0
2	AT Refresh Control 0: Disable AT Refresh 1: Enable AT Refresh	0
1:0	Keyboard Controller Clock Speed Control PCICLK Speed 25 MHz 33 MHz 00: 6.25 MHz 8.25 MHz 01: 12.5 MHz 16.5 MHz 10: 8.33 MHz 11 MHz 11: 16.67 MHz 22 MHz	00

Register 21: PCI IDE Interrupt Request 0 Routing Control Register (Read/Write) – Index=4BH with an 8-bit access

Bit	Function	Default
7	IDE Interrupt 0 Routing Control: 0: Interrupt 0 Routing Enabled 1: Interrupt 0 Routing Disabled	0
6:4	Reserved	000
3:0	Interrupt Request Level that PCI IDE IRQ0 is Routed to: 0000: Reserved 1000: Reserved 0001: Reserved 1001: IRQ9 0010: Reserved 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 1101: Reserved 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15	0000

Register 22: PCI IDE Interrupt Request 1 Routing Control Register (Read/Write) – Index=4CH with an 8-bit access

Bit	Function	Default
7	IDE Interrupt 1 Routing Control: 0: Interrupt 0 Routing Enabled 1: Interrupt 0 Routing Disabled	0
6:4	Reserved	000
3:0	Interrupt Request Level that PCI IDE IRQ0 is Routed to: 0000: Reserved 1000: Reserved 0001: Reserved 1001: IRQ9 0010: Reserved 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 1101: Reserved 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15	0000

Register 23: CY82C693U Stand-Alone Control and USB Host Controller Control Register (Read/Write) – Index=4DH with an 8-bit access

Bit	Function	Default
7	External PCI Arbiter Protocol Control: 0: All CY82C693U masters arbitrate for the PCI bus using the $\overline{\text{PREQ/PGNT}}$ signal pair. 1: This setting is used to allow USB master arbitration to use the $\overline{\text{SREQ/SGNT}}$ signal pair. All other CY82C693U masters use $\overline{\text{PREQ/PGNT}}$. Note: This register bit is only for use when the internal PCI arbiter is disabled. If the internal PCI arbiter is used (pin 194 HIGH at power-up), this register bit does not effect CY82C693U operation.	0
6:5	Flush Request/Acknowledge Handshake Control: 00: Normal Operation: Flush request/acknowledge handshake is performed using the $\overline{\text{FREQACK}}$ signal. 01: Flush request/acknowledge handshake is performed using the $\overline{\text{FREQACK}}$ signal. However, $\overline{\text{IRQ8}}$ (the RTC interrupt) is masked internally and will not be seen by the interrupt controller. 10: Flush request is automatically acknowledged internally. The $\overline{\text{FREQACK}}$ signal will remain in high impedance. 11: Flush request is automatically acknowledged internally. $\overline{\text{IRQ8}}$ (the RTC interrupt) is masked internally and will not be seen by the interrupt controller. $\overline{\text{IRQ8}}$ is driven out on the $\overline{\text{FREQACK}}$ pin (pin 190). Note: Flush request may not be automatically acknowledged internally if a coherent path to memory is not guaranteed.	00
4	IDE Controller Decode Control: 0: IDE Controller decode will only use lower 16 address bits (64KB). 1: IDE Controller decode will use all 32-bits (4GB). Note: If I/O transactions are allowed above 64KB, this bit should be set to 1. Otherwise, system conflicts may occur. See IDE Controller Configuration Registers 7, 8, &9.	0
3	ROM Space Decode: 0: 512KB ROM space decode 1: 1MB ROM space decode	0
2	Reserved.	
1	Reserved.	
0	USB Interface Control: 0: Disable USB interface 1: Enable USB interface	0

Register 24: CY82C693U USB Control Register 1 (Read/Write) - Index=4EH with an 8-bit access

Bit	Function	Default
7:6	USB Host Controller Test Mode Selection.	00
5	USB Host Controller Test Mode Enable.	0
4	USB Host Controller IRQ1 Enable: 0: Disabled 1: Enabled	0
3	USB Host Controller IRQ12 Enable: 0: Disabled 1: Enabled	0
2	USB Host Controller FA20 Enable: 0: Disabled 1: Enabled	0
1	USB Host Controller SMI Enable: 0: Disabled 1: Enabled	0
0	Reserved.	0

Register 24: CY82C693U USB Control Register 2 (Read/Write) - Index=4FH with an 8-bit access

Bit	Function	Default
7	USB Host Controller SMIACT Enable: 0: Disabled 1: Enabled	0
6	USB Host Controller PWREN Enable: 0: Disable 1: Enabled	0
5:4	USB Host Controller Interrupt Mapping: 00: PCI INTAN 01: PCI INTBN 10: PCI INTCN 11: PCI INTDN	00
3:0	Reserved.	0000

Primary Channel IDE PIO (Programmed I/O)
PCI Configuration Registers (Function 1 during Configuration Cycle)

Register 0: Vendor ID Number (Read Only) – Index=00H with a 16-bit access

Bit	Function	Default
15:0	Cypress ID Number: 0001000010000000	1080H

Register 1: Device ID Number (Read Only) – Index=02H with a 16-bit access

Bit	Function	Default
15:0	CY82C693U Device ID Number: 1100011010010011	C693H

Register 2: Command Register (Read/Write) – Index=04H with a 16-bit access

Bit	Function	Default
15:10	Reserved.	000000
9	Reserved, Must be set to 0.	0
8	$\overline{SERR}$ Reporting Enable: 0: $\overline{SERR}$ Reporting Disabled 1: $\overline{SERR}$ Reporting Enabled	0
7	Reserved, Must be set to 0.	0
6:3	Reserved.	0000
2	Bus Master Enable. 0 Bus Master Disabled 1 Bus Master Enabled	0
1	Memory Access Enable. This bit is forced to 0 (Memory Access is not allowed to IDE Controller).	0
0	I/O Access Enable.: 0: Primary IDE controller disabled. 1: Primary IDE controller enabled. The Primary IDE Controller will respond to PIO I/O accesses when this bit is set.	0

Register 3: Status Register (Read/Write) – Index=06H with a 16-bit access

Bit	Function	Default
15	Reserved. Read as 0.	0
14	PCI System Error This bit is not used by the IDE controllers.	0
13	PCI Master-Abort READ: 0: No PCI Master-Abort Occurred 1: PCI Master-Abort Occurred WRITE: 0: No change to register 1: Clear Register	0
12	CY82C693U Detection of Target-Abort (from another PCI target) READ: 0: No PCI Target-Abort Occurred 1: PCI Target-Abort Occurred WRITE: 0: No change to register 1: Clear Register	0
11	CY82C693U693U Assertion of Target-Abort (CY82C693U is the target) READ: 0: No PCI Target-Abort Occurred 1: PCI Target-Abort Occurred WRITE: 0: No change to register 1: Clear Register	0
10:9	$\overline{\text{DEVSEL}}$ Timing Status. These register bits are Read Only and will always return 01 (medium timing).	01
8	Reserved. Read as 0.	0
7	Fast Back-to-Back Transfer Capable. This Read Only bit will always return 1.	1
6:0	Reserved.	0000000

Register 4: Revision ID Number (Read Only) – Index=08H with an 8-bit access

Bit	Function	Default
7:0	Current Revision of the Part (00000000)	00H

Register 5: Class Code Register (Read Only) – Index=09H with a 32-bit access

Bit	Function	Default
7:0	Sub Class Code - IDE mass storage disk controller w/ bus mastering support.	00010180H

Register 6: Header Type Register (Read Only) – Index=0EH with an 8-bit access

Bit	Function	Default
7:0	Device Type, multi-function device	80H

Register 7: Primary IDE Command Address Register (Read/Write) – Index=10H with a 32-bit access

Bit	Function	Default
31:0	Primary IDE Command Address Register: This register specifies the amount of I/O space required for the primary IDE command registers. The IDE command block requires 8 bytes of I/O space (Bits [2:0] are hardwired to 001). Bits [15:3] are programmable. Bits [31:16] are hard-wired to 0000H.	00000000H

Register 8: Primary IDE Control Address Register (Read/Write) – Index=14H with a 32-bit access

Bit	Function	Default
31:0	Primary IDE Control Address Register: This register specifies the amount of I/O space required for the primary IDE control registers. The IDE control block requires 2 bytes of I/O space (Bits [1:0] are hardwired to 01). Bits [15:2] are programmable. Bits [31:16] are hard-wired to 0000H.	00000000H

Note: Register Indices 18H-1FH will return all zeroes when read.

Register 9: Primary Bus Master IDE Control Address Register (Read/Write) – Index=20H with a 32-bit access

Bit	Function	Default
31:0	Bus Master IDE Control Address Register: This register specifies the amount of I/O space required for the bus master IDE control registers. The bus master IDE control block requires 16 bytes of I/O space (Bits [3:0] are hardwired to 0001). Bits [15:4] are programmable. Bits [31:16] are hard-wired to 0000H.	00000000H

Register 10: Primary IDE Interrupt $\overline{\text{INTA}}$ Control Register (Read/Write) – Index=3CH with an 8-bit access

Bit	Function	Default
7:0	Primary IDE Interrupt Control Register: This register chooses which ISA IRQ that the $\overline{\text{INTA}}$ output from the PCI IDE controller is routed to. The default is IRQ14.	14H

Register 11: Primary IDE Interrupt Pin Control Register (Read/Write) – Index=3DH with an 8-bit access

Bit	Function	Default
7:0	Primary IDE Interrupt Pin Control Register: 00H: The Primary IDE Channel Interrupt is connected directly to one of the ISA IRQ Pins. 01H: The Primary IDE Channel Interrupt is connected to PCI $\overline{\text{INTA}}$ internally.	01H

Note: Register Indices 3EH-3FH will return all zeroes when read.

Register 12: Primary IDE Control Register (Read/Write) – Index=40H with a 32-bit access

Bit	Function	Default
31:18	Reserved	00000000000000
17	Reserved	0
16	Reserved	0
15:14	Reserved	00
13	Retry I/O Accesses Not Completed by 16 PCI Clocks Control: 0: I/O Accesses Not Completed by 16 PCI Clocks will not be retried. 1: I/O Accesses Not Completed by 16 PCI Clocks will be retried.	0
12:11	Reserved	00
10	Slave Drive Prefetch Control: 0: Disable Prefetch (Must be 0 for CDROM accesses). 1: Enable Prefetch	0
9	Post Write Control: 0: One level FIFO for Posted Writes 1: Four levels of FIFO for Posted Writes	0
8	Master Drive Prefetch Control: 0: Disable Prefetch (Must be 0 for CDROM accesses). 1: Enable Prefetch	0
7:6	Reserved	00
5:4	Post Write Length Control: The value programmed into this register+1 will be the length of the Post Write Bursts that the IDE write state machine will attempt to the IDE drive when the AT bus grant is received.	00
3:2	Reserved	00
1:0	Prefetch Length Control: The value programmed into this register+1 will be the length of the Prefetch Bursts that the IDE read state machine will attempt to the IDE drive when the AT bus grant is received.	00

Note: Register Indices 44H-47H will return all zeroes when read.

Register 13: Primary IDE Address Setup Control Register (Read/Write) – Index=48H with a 32-bit access

Bit	Function	Default
31:8	Reserved	000000H
7:4	Slave Drive IDE Address Setup Time: The value programmed into this register +1 will be the setup (in PCI Clock cycles) from address valid to $\overline{IOR}$ or $\overline{IOW}$ valid.	0011
3:0	Master Drive IDE Address Setup Time: The value programmed into this register +1 will be the setup (in PCI Clock cycles) from address valid to $\overline{IOR}$ or $\overline{IOW}$ valid.	0011

Register 14: Primary Master Drive IDE $\overline{IOR}$ Command Control Register (Read/Write) – Index=4CH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Master Drive IDE $\overline{IOR}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in AT Clock cycles) of the asserted $\overline{IOR}$ signal.	0011
3:0	16-Bit Master Drive IDE $\overline{IOR}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in AT Clock cycles) that $\overline{IOR}$ must be deasserted between transfers.	0011

Register 15: Primary Master Drive IDE $\overline{IOW}$ Command Control Register (Read/Write) – Index=4DH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Master Drive IDE $\overline{IOW}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ signal.	0110
3:0	16-Bit Master Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1110

Register 16: Primary Slave Drive IDE $\overline{IOR}$ Command Control Register (Read/Write) – Index=4EH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Slave Drive IDE $\overline{IOR}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOR}$ signal.	0011
3:0	16-Bit Slave Drive IDE $\overline{IOR}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOR}$ must be deasserted between transfers.	0011

Register 17: Primary Slave Drive IDE $\overline{IOW}$ Command Control Register (Read/Write) – Index=4FH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Slave Drive IDE $\overline{IOW}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ signal.	0110
3:0	16-Bit Slave Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1110

Register 18: Primary Master Drive 8-Bit IDE Command Control Register (Read/Write) – Index=50H with an 8-bit access

Bit	Function	Default
7:4	8-Bit Master Drive IDE Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ or $\overline{IOR}$ signal.	1010
3:0	8-Bit Master Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1010

Register 19: Primary Slave Drive 8-Bit IDE Command Control Register (Read/Write) – Index=51H with an 8-bit access

Bit	Function	Default
7:4	8-Bit Slave Drive IDE Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ or $\overline{IOR}$ signal.	1010
3:0	8-Bit Slave Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1010

Note: The 8-bit IDE Command Registers are used for mode 0, 1, and 2 drives. For mode 3 and 4 drives, the 8 and 16 bit timings are the same.

Register 20: Primary Master Drive IORDY Control Register (Read/Write) – Index=52H with an 8-bit access

Bit	Function	Default
7:4	Reserved	0000
3:2	Read Synchronization Control: The value in this register selects the number of PCI clocks used to synchronize the rising edge of IORDY during read transactions.	01
1:0	Write Synchronization Control: The value in this register selects the number of PCI clocks used to synchronize the rising edge of IORDY during write transactions.	01

Register 21: Primary Slave Drive IORDY Control Register (Read/Write) – Index=53H with an 8-bit access

Bit	Function	Default
7:4	Reserved	0000
3:2	Read Synchronization Control: The value in this register selects the number of PCI clocks used to synchronize the rising edge of IORDY during read transactions.	01
1:0	Write Synchronization Control: The value in this register selects the number of PCI clocks used to synchronize the rising edge of IORDY during write transactions.	01

Secondary Channel IDE PIO (Programmed I/O)
PCI Configuration Registers (Function 2 during Configuration Cycle)

Register 0: Vendor ID Number (Read Only) – Index=00H with a 16-bit access

Bit	Function	Default
15:0	Cypress ID Number: 0001000010000000	1080H

Register 1: Device ID Number (Read Only) – Index=02H with a 16-bit access

Bit	Function	Default
15:0	CY82C693U Device ID Number: 1100011010010011	C693H

Register 2: Command Register (Read/Write) – Index=04H with a 16-bit access

Bit	Function	Default
15:10	Reserved.	000000
9	Reserved, Must be set to 0.	0
8	$\overline{\text{SERR}}$ Reporting Enable: 0: $\overline{\text{SERR}}$ Reporting Disabled 1: $\overline{\text{SERR}}$ Reporting Enabled	0
7	Reserved, Must be set to 0.	0
6	Reserved	0
5:1	Reserved	00000
0	I/O Access Enable.: 0: Secondary IDE controller disabled. 1: Secondary IDE controller enabled. The Secondary IDE Controller will respond to PIO I/O accesses when this bit is set.	0

Register 3: Status Register (Read/Write) – Index=06H with a 16-bit access

Bit	Function	Default
15	Not Implemented. Read as 0.	0
14	PCI System Error This bit is not used by the IDE controllers.	0
13	PCI Master-Abort READ: 0: No PCI Master-Abort Occurred 1: PCI Master-Abort Occurred WRITE: 0: No change to register. 1: Clear Register	0
12	CY82C693U Detection of Target-Abort (from another PCI target) READ: 0: No PCI Target-Abort Occurred 1: PCI Target-Abort Occurred WRITE: 0: No change to register. 1: Clear Register	0
11	CY82C693U Assertion of Target-Abort (CY82C693U is the target) READ: 0: No PCI Target-Abort Occurred 1: PCI Target-Abort Occurred WRITE: 0: No change to register. 1: Clear Register	0
10:9	$\overline{\text{DEVSEL}}$ Timing Status. These register bits are Read Only and will always return 01 (medium timing)	01
8	Not Implemented. Read as 0.	0
7	Fast Back-to-Back Transfer Capable. This Read Only bit will always return 1.	1
6:0	Reserved	0000000

Register 4: Revision ID Number (Read Only) – Index=08H with an 8-bit access

Bit	Function	Default
7:0	Current Revision of the Part (00000000)	00H

Register 5: Class Code Register (Read Only) – Index=09H with a 32-bit access

Bit	Function	Default
7:0	Sub Class Code - IDE mass storage disk controller.	00010100H

Register 6: Header Type Register (Read Only) – Index=0EH with an 8-bit access

Bit	Function	Default
7:0	Device Type, multi-function device	80H

Register 7: Secondary IDE Command Address Register (Read/Write) – Index=10H with a 32-bit access

Bit	Function	Default
31:0	Secondary IDE Command Address Register: This register specifies the amount of I/O space required for the secondary IDE command registers. The IDE command block requires 8 bytes of I/O space (Bits [2:0] are hardwired to 001). Bits [15:3] are programmable. Bits [31:16] are hard-wired to 0000H.	00000000H

Register 8: Secondary IDE Control Address Register (Read/Write) – Index=14H with a 32-bit access

Bit	Function	Default
31:0	Secondary IDE Control Address Register: This register specifies the amount of I/O space required for the secondary IDE control registers. The IDE control block requires 2 bytes of I/O space (Bits [1:0] are hardwired to 01). Bits [15:2] are programmable. Bits [31:16] are hard-wired to 0000H.	00000000H

Note: Register Indices 18H-3BH will return all zeroes when read.

Register 9: Secondary IDE Interrupt $\overline{\text{INTB}}$ Control Register (Read/Write) – Index=3CH with an 8-bit access

Bit	Function	Default
7:0	Secondary IDE Interrupt Control Register: This register chooses which ISA IRQ that the $\overline{\text{INTB}}$ output from the PCI IDE controller is routed to. The default is IRQ15.	14H

Register 10: Secondary IDE Interrupt Pin Control Register (Read/Write) – Index=3DH with an 8-bit access

Bit	Function	Default
7:0	Secondary IDE Interrupt Pin Control Register: 00H: The secondary IDE Channel Interrupt is connected directly to one of the ISA IRQ Pins. 01H: The secondary IDE Channel Interrupt is connected to PCI $\overline{\text{INTB}}$ internally.	01H

Note: Register Indices 3EH-3FH will return all zeroes when read.

Register 11: Secondary IDE Control Register (Read/Write) – Index=40H with a 32-bit access

Bit	Function	Default
31:16	Reserved	0000000000000000
15:14	Reserved	00
13	Retry I/O Accesses Not Completed by 16 PCI Clocks Control: 0: I/O Accesses Not Completed by 16 PCI Clocks will not be retried. 1: I/O Accesses Not Completed by 16 PCI Clocks will be retried.	0
12:11	Reserved	00
10	Slave Drive Prefetch Control: 0: Disable Prefetch (Must be 0 for CDROM accesses). 1: Enable Prefetch	0
9	Post Write Control: 0: One level FIFO for Posted Writes 1: Four levels of FIFO for Posted Writes	0
8	Master Drive Prefetch Control: 0: Disable Prefetch (Must be 0 for CDROM accesses). 1: Enable Prefetch	0
7:6	Reserved	00
5:4	Post Write Length Control: The value programmed into this register+1 will be the length of the Post Write Bursts that the IDE write state machine will attempt to the IDE drive when the AT bus grant is received.	00
3:2	Reserved	00
1:0	Prefetch Length Control: The value programmed into this register+1 will be the length of the Prefetch Bursts that the IDE read state machine will attempt to the IDE drive when the AT bus grant is received.	00

Note: Register Indices 44H-47H will return all zeroes when read.

Register 12: Secondary IDE Address Setup Control Register (Read/Write) – Index=48H with a 32-bit access

Bit	Function	Default
31:8	Reserved	000000H
7:4	Slave Drive IDE Address Setup Time: The value programmed into this register +1 will be the setup (in PCI Clock cycles) from address valid to $\overline{IOR}$ or $\overline{IOW}$ valid.	0011
3:0	Master Drive IDE Address Setup Time: The value programmed into this register +1 will be the setup (in PCI Clock cycles) from address valid to $\overline{IOR}$ or $\overline{IOW}$ valid.	0011

Register 13: Secondary Master Drive IDE $\overline{IOR}$ Command Control Register (Read/Write) – Index=4CH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Master Drive IDE $\overline{IOR}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOR}$ signal.	0011
3:0	16-Bit Master Drive IDE $\overline{IOR}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOR}$ must be deasserted between transfers.	0011

Register 14: Secondary Master Drive IDE $\overline{IOW}$ Command Control Register (Read/Write) – Index=4DH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Master Drive IDE $\overline{IOW}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ signal.	0110
3:0	16-Bit Master Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1110

Register 15: Secondary Slave Drive IDE $\overline{IOR}$ Command Control Register (Read/Write) – Index=4EH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Slave Drive IDE $\overline{IOR}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOR}$ signal.	0011
3:0	16-Bit Slave Drive IDE $\overline{IOR}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOR}$ must be deasserted between transfers.	0011

Register 16: Secondary Slave Drive IDE $\overline{IOW}$ Command Control Register (Read/Write) – Index=4FH with an 8-bit access

Bit	Function	Default
7:4	16-Bit Slave Drive IDE $\overline{IOW}$ Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ signal.	0110
3:0	16-Bit Slave Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1110

Register 17: Secondary Master Drive 8-Bit IDE Command Control Register (Read/Write) – Index=50H with an 8-bit access

Bit	Function	Default
7:4	8-Bit Master Drive IDE Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ or $\overline{IOR}$ signal.	1010
3:0	8-Bit Master Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1010

Register 18: Secondary Slave Drive 8-Bit IDE Command Control Register (Read/Write) – Index=51H with an 8-bit access

Bit	Function	Default
7:4	8-Bit Slave Drive IDE Command Pulse Width Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) of the asserted $\overline{IOW}$ or $\overline{IOR}$ signal.	1010
3:0	8-Bit Slave Drive IDE $\overline{IOW}$ Command Recovery Time: The value programmed into this register +1 will be the duration (in PCI Clock cycles) that $\overline{IOW}$ must be deasserted between transfers.	1010

Note: The 8-bit IDE Command Registers are used for mode 0, 1, and 2 drives. For mode 3 and 4 drives, the 8 and 16 bit timings are the same.

Register 19: Secondary Master Drive IORDY Control Register (Read/Write) – Index=52H with an 8-bit access

Bit	Function	Default
7:4	Reserved	0000
3:2	Read Synchronization Control: The value in this register selects the number of PCI clocks used to synchronize the rising edge of IORDY during read transactions.	01
1:0	Write Synchronization Control: The value in this register selects the number of PCI clocks used to synchronize the rising edge of IORDY during write transactions.	01

USB Host Controller PCI Configuration Registers (Function 3 during Configuration Cycle)

Register 0: Vendor ID Number (Read Only) - Index=00H with a 16-bit access

Bit	Function	Default
15:0	Cypress ID Number: 0001000010000000	1080H

Register 1: Device ID Number (Read Only) - Index=02H with a 16-bit access

Bit	Function	Default
15:0	CY82C693U Device ID Number: 1100011010010011	C693H

Register 2: Command Register (Read/Write) - Index=04H with a 16-bit access

Bit	Function	Default
15:10	Reserved. Always 0.	000000
9	Fast Back-to-Back enable: 0: Disabled 1 Enabled The USB Host Controller always generates fast back-to-back access. Therefore, this bit must be set to 1.	0
8	$\overline{\text{SERR}}$ Reporting Enable: 0: Disabled 1: Enabled	0
7	Wait Cycle Control. Always 0.	0
6	PERR Detection Enabled. Not supported by CY82C693U.	0
5	VGA Palette Snooping. Always 0	0
4	Memory Write and Invalidate Command Enable.	0
3	Special Cycle Enable. Always 0.	0
2	PCI Master Enable: 0: Bus Master Disabled 1: Bus Master Enabled This bit enables the USB Host Controller to act as a PCI master. If set to 0, no PCI access will be generated by the USB Host Controller. This bit must be set to 1 before USB transactions can start.	0
1	Memory Access Enable: 0: Memory Access Disabled 1: Memory Access Enabled This bit must be set to 1 for the USB Host Controller to perform memory access.	0
0	I/O Access Enable: 0: I/O Access Disabled 1: I/O Access Enabled This bit must be set to 1 for the USB Host Controller to perform I/O access	0

Register 3: Status Registers (Read/Write) - Index=06H with a 16-bit access

Bit	Function	Default
15	Detected Parity Error: READ: 0: No Parity Error Detected 1: Parity Error Detected WRITE: 0: No change to register 1: Clear register This bit is set to 1 whenever the USB Host Controller detects a parity error, even if the Parity Error Detection Enable bit (command register, bit 6) is disabled.	0
14	PCI $\overline{\text{SERR}}$ Status: READ: 0: No Parity Error Detected 1: Parity Error Detected WRITE: 0: No change to register 1: Clear register This bit is set to 1 when the USB Host Controller detects a PCI address parity error.	0
13	PCI Master-Abort READ: 0: No PCI Master-Abort occurred 1: PCI Master-Abort occurred WRITE: 0: No change to register 1: Clear register	0
12	CY82C693U Detection of Target-Abort (from another PCI target) READ: 0: No PCI Target-Abort occurred 1: PCI Target-Abort occurred WRITE: 0: No change to register 1: Clear register	0
11	CY82C693U Assertion of Target-Abort (CY82C693U is the target) READ: 0: No PCI Target-Abort occurred 1: PCI Target-Abort occurred WRITE: 0: No change to register 1: Clear register	0
10:9	$\overline{\text{DEVSEL}}$ Timing Status: These register bits are Read Only and will always read as 01 (medium timing)	01
8	Data Parity Reported. Not supported by CY82C693U ($\overline{\text{PERR}}$)	0
7	Fast Back-to-Back Transfer Capable. This Read Only bit will always return 1	1
6:0	Reserved. Always 0.	0

Register 4: Revision ID Number (Read Only) - Index=08H with an 8-bit access

Bit	Function	Default
7:0	Current Revision of the Part (00000000)	00H

Register 5: Class Code Register (Read Only) - Index=09H with a 24-bit access

Bit	Function	Default
23:0	Sub Class Code: Base Class Code: Serial Bus Controller (OCH) Sub Class Code: Universal Serial Bus (03H) Programming Interface: USB OpenHCI (10H)	0C0310H

Register 6: Cache Line Size (Read/Write) - Index=0CH with an 8-bit access

Bit	Function	Default
7:0	Cache Line Size. This register identifies the system cache line size in units of 32-bits words. The USB Host Controller will only store the value of bit 3 in this register since the cache line size of 32 bytes is the only value applicable to the design. Any value other than 08H written to this register will be read back as 00H	00H

Register 7: Latency Timer (Read/Write) - Index=0DH with an 8-bit access

Bit	Function	Default
7:0	Latency Timer. This register identifies the value of the latency timer in PCI clocks for PCI bus master cycles.	00H

Register 8: Header Type Register (Read Only) - Index=0EH with an 8-bit access

Bit	Function	Default
7:0	Header Type. This register identifies the type of the predefined header in the configuration space. Since the USB Host Controller is a single function device and not a PCI-to-PCI bridge, this byte should be read as 00H.	00H

Register 9: BIST Register (Read Only) - Index=0FH with an 8-bit access

Bit	Function	Default
7:0	Built-In Self Test. This register identifies the control and status of Built-In Self Test. The USB Host Controller does not implement BIST, so this register is read only	00H

Register 10: Base Address Register (Read/Write) - Index=10H with a 32-bit access

Bit	Function	Default
31:12	Base Address. POST writes the value of the memory base address to this register.	00000H
11:4	Always 0. Indicates a 4K byte address range is requested.	00H
3	Always 0. Indicates there is no support for prefetchable memory.	0
2:1	Always 0. Indicates that the base register is 32-bit wide and can be placed anywhere in 32-bit memory space	00
0	Always 0. Indicates that the USB Host Controller operational registers are mapped into memory space	0

Note: The Base Address Register contains the base address of the memory-mapped USB Host Controller Operational Registers, which are defined in the next section.

Register 11: Interrupt LIne Register (Read/Write) - Index=3CH with an 8-bit access

Bit	Function	Default
7:0	Interrupt Line. This register identifies which of the system interrupt controllers the device's interrupt pin is connected to. The value of this register is used by the device drivers and has no direct meaning to the USB Host Controller	00H

Register 12: Interrupt Pin Register (Read/Write) - Index=3DH with an 8-bit access

Bit	Function	Default
7:0	Interrupt Pin Register. This register identifies which interrupt pin a device uses. Since the USB Host Controller uses $\overline{INTA}$, this register is set to 01H	01H

Register 13: Min_Gnt Register (Read Only) - Index=3EH with an 8-bit access

Bit	Function	Default
7:0	PCI Min_Gnt. This register specifies how long of a burst the USB Host Controller needs assuming a clock rate of 33 MHz. The value specifies a period of time in units of 1/4 micro-second	00H

Register 14: Max_Lat Register (Read Only) - Index=3FD with an 8-bit access

Bit	Function	Default
7:0	PCI Max. Latency. This register specifies how often the USB Host Controller needs access to the PCI bus assuming a clock rate of 33 MHz. The value specifies a period of time in units of 1/4 microsecond.	00H

Register 15: ASIC Test Mode Enable Register (Read/Write) - Index=40H with a 32-bit access

Bit	Function	Default
31:0	ASIC Test Mode Enable. For normal USB Host Controller operation this register must be set to 0.	0XXXXXXXH

Register 16: ASIC Operational Mode Enable Register (Read/Write) - Index=44H with an 8-bit access

Bit	Function	Default
7:0	ASIC Operational Mode Enable. For normal USB Host Controller operation this register must remain as 0.	00H

USB Host Controller Operational Registers

The USB Host Controller Operational Registers are accessed using memory-mapped I/O. The address of each operational register is given by the Base Address (specified in the Base Address Register in the USB Host Controller PCI Configuration Registers) and the offset value of the register.

Register 0: HcRevision (Read Only) - Offset=00H with a 32-bit access

Bit	Function	Default
31:8	Reserved.	000000H
7:0	Revision Number. Indicates the OpenHCI Specification revision number implemented by the hardware.	10H

Register 1: HcControl (Read/Write) - Offset=04H with a 32-bit access

Bit	Function	Default
31:11	Reserved.	0H
10	Remote Wakeup Connected Enable: If a remote wakeup signal is supported, this bit is used to enable that operation. Since there is no remote wakeup signal supported, this bit is ignored.	0
9	Remote Wakeup Connected: This Read Only bit indicated whether the Host Controller supports a remote wakeup signal. This implementation does not support any such signal. The bit is hard-coded to 0.	0
8	Interrupt Routing: This bit is used for interrupt routing: 0: Interrupt routed to normal interrupt mechanism (INT) 1: Interrupt routed to SMI	0
7:6	Host Controller Functional State: This field is used to set the Host Controller state. The state encoding are: 00: USBReset 01: USBResume 10: USBOperational 11: USBSuspend The USB Host Controller may force a state change from USBSuspend to USBResume after detecting resume signaling from a downstream port.	00
5	Bulk List Enable: When set this bit enables processing of the Bulk list.	0
4	Control List Enable: When set this bit enables processing of the Control list.	0
3	Isochronous Enable: When cleared, this bit disables the Isochronous List when the Periodic List is enabled (so Interrupt EDs may be serviced). While processing the Periodic List, the Host Controller will check this bit when it finds an isochronous ED.	0
2	Periodic List Enable: When set, this bit enables processing of the Periodic (interrupt and isochronous) list. The Host Controller checks this bit prior to attempting any periodic transfer in a frame.	0
1:0	Control Bulk Service Ratio: Specifies the number of Control Endpoints serviced for every Bulk Endpoint. Encoding is N-1 where N is the number of Control Endpoints (i.e. ‘00’ = 1 Control Endpoint; ‘11’ = 4 Control Endpoints)	00

Register 2: HcCommandStatus (Read/Write) - Offset=08H with a 32-bit access

Bit	Function	Default
31:18	Reserved.	
17:16	Schedule Overrun Count: This field increments every time the SchedulingOverrun bit in HcInterruptStatus is set. The count wraps from '11' to '00'.	00
15:4	Reserved.	0H
3	Ownership Change Request: When set by software, this bit sets the OwnershipChange field in HcInterruptStatus. The bit is cleared by software.	0
2	Bulk List Filled: When set, this bit indicates there is an active ED on the Bulk List. the bit may be set by either software or the Host Controller. The bit is cleared by the Host Controller each time it begins processing the head of the Bulk List	0
1	Control List Filled: When set, this bit indicates there is an active ED on the Control List. The bit may be set by either software or the Host Controller. The bit is cleared by the Host Controller each time it begins processing the head of the Control List	0
0	Host Controller Reset: This bit is set to initiate a software reset. This bit is cleared by the Host Controller upon completion of the reset operation	0

Register 3: HcInterruptStatus (Read/Write) - Offset=0CH with a 32-bit access

Bit	Function	Default
31	Reserved.	0
30	Ownership Change: This bit is set when the OwnershipChangeRequest bit of HcCommandStatus is set.	0
29:7	Reserved.	0H
6	Root Hub Status Change: This bit is set when the content of HcRhStatus or the content of any HcRhPortStatus register has changed	0
5	FrameNumber Overflow: This bit is set when bit 15 of FrameNumber changes value from '0' to '1' or from '1' to '0'.	0
4	Unrecoverable Error: Read Only. This event is not implemented and is hard-coded to '0'. All writes are ignored.	0
3	Resume Dectected: This bit is set when the Host Controller detects resume signaling on a downstream port.	0
2	Start Of Frame: This bit is set when the Frame Management block signals a 'Start of Frame' event.	0
1	Writeback Done Head: This bit is set after the Host Controller has written HcDoneHead to HccaDoneHead.	0
0	Scheduling Overrun: This bit is set when the List Processor determines a Schedule Overrun has occurred.	0

Register 4: HcInterruptEnable (Read/Write) - Offset=10H with a 32-bit access

Bit	Function	Default
31	Master Interrupt Enable: This bit is a global interrupt enable. A write of '1' allows interrupts to be enabled via the specific enable bits listed below.	0
30	Ownership Change Enable: 0: Ignore 1: Enable interrupt generation due to Ownership Change	0
29:7	Reserved.	0H
6	Root Hub Status Change Enable: 0: Ignore 1: Enable interrupt generation due to Root Hub Status Change	0
5	Frame Number Overflow Enable: 0: Ignore 1: Enable interrupt generation due to Frame Number Overflow	0
4	Unrecoverable Error Enable: This event is not implemented. All writes to this bit will be ignored.	0
3	Resume Detected Enable: 0: Ignore 1: Enable interrupt generation due to Resume Detected	0
2	Start Of Frame Enable: 0: Ignore 1: Enable interrupt generation due to Start of Frame	0
1	Writeback Done Head Enable: 0: Ignore 1: Enable interrupt generation due to Writeback Done Head	0
0	Scheduling Overrun Enable: 0: Ignore 1: Enable interrupt generation due to Scheduling Overrun	0

Register 5: HcInterruptDisable (Read/Write) - Offset=14H with a 32-bit access

Bit	Function	Default
31	Master Interrupt Enable: This bit is a global interrupt disable. A write of '1' disables all interrupts.	0
30	Ownership Change Enable: 0: Ignore 1: Disable interrupt generation due to Ownership Change	0
29:7	Reserved.	0H
6	Root Hub Status Change Enable: 0: Ignore 1: Disable interrupt generation due to Root Hub Status Change	0
5	Frame Number Overflow Enable: 0: Ignore 1: Disable interrupt generation due to Frame Number Overflow	0
4	Unrecoverable Error Enable: This event is not implemented. All writes to this bit will be ignored.	0
3	Resume Detected Enable: 0: Ignore 1: Disable interrupt generation due to Resume Detected	0
2	Start Of Frame Enable: 0: Ignore 1: Disable interrupt generation due to Start of Frame	0
1	Writeback Done Head Enable: 0: Ignore 1: Disable interrupt generation due to Writeback Done Head	0
0	Scheduling Overrun Enable: 0: Ignore 1: Disable interrupt generation due to Scheduling Overrun	0

Register 6: HcHCCA (Read/Write) - Offset=18H with a 32-bit access

Bit	Function	Default
31:8	HCCA: Pointer to HCCA base address.	0H
7:0	Reserved.	0H

Register 7: HcPeriodCurrentED (Read/Write) - Offset=1CH with a 32-bit access

Bit	Function	Default
31:4	Period Current ED: Pointer to the current Periodic List ED.	0H
3:0	Reserved.	0H

Register 8: HcControlHeadED (Read/Write) - Offset=20H with a 32-bit access

Bit	Function	Default
31:4	Control Head ED: Pointer to the Control List Head ED.	0H
3:0	Reserved.	0H

Register 9: HcControlCurrentED (Read/Write) - Offset=24H with a 32-bit access

Bit	Function	Default
31:4	Control Current ED: Pointer to the current Control List ED.	0H
3:0	Reserved.	0H

Register 10: HcBulkHeadED (Read/Write) - Offset=28H with a 32-bit access

Bit	Function	Default
31:4	Bulk Head ED: Pointer to the Bulk List Head ED.	0H
3:0	Reserved.	0H

Register 11: HcBulkCurrentED (Read/Write) - Offset=2CH with a 32-bit access

Bit	Function	Default
31:4	Bulk Current ED: Pointer to the current Bulk List ED.	0H
3:0	Reserved.	0H

Register 12: HcDoneHead (Read/Write) - Offset=30H with a 32-bit access

Bit	Function	Default
31:4	Done Head: Pointer to the current Done List Head ED.	0H
3:0	Reserved.	0H

Register 13: HcFmInterval (Read/Write) - Offset=34H with a 32-bit access

Bit	Function	Default
31	Frame Interval Toggle: This bit is toggled by Host Controller Driver whenever it loads a new value into FrameInterval.	
30:16	FS Largest Data Packet: This field specifies a value which is loaded into the Largest Data Packet Counter at the beginning of each frame.	
15:14	Reserved.	0H
13:0	Frame Interval: This field specifies the length of a frame as (bit times – 1). For 12,000 bit times in a frame, a value of 11,999 is stored here.	2EDFH

Register 13: HcFrameRemaining (Read Only) - Offset=38H with a 32-bit access

Bit	Function	Default
31	Frame Remaining Toggle: This bit is loaded with FrameIntervalToggle when FrameRemaining is loaded.	0
30:14	Reserved.	0H
13:0	Frame Remaining: This field is a 14 bit decrementing counter used to time a frame. When the Host Controller is in the USBOperational state the counter decrements each 12 MHz clock period. When the count reaches 0, the end of a frame has been reached. The counter reloads with FrameInterval at that time. In addition, the counter loads when the Host Controller transitions into USBOperational state.	0H

Register 14: HcFmNumber (Read Only) - Offset=3CH with a 32-bit access

Bit	Function	Default
31:16	Reserved.	0H
15:0	Frame Number: This field is a 16 bit incrementing counter. The count is incremented coincident with the loading of FrameRemaining. The count will roll over from 'FFFFH' to '0H.'	0H

Register 15: HcPeriodicStart (Read/Write) - Offset=40H with a 32-bit access

Bit	Function	Default
31:14	Reserved.	0H
13:0	Periodic Start: This field contains a value used by the List Processor to determine where in a frame the Periodic List processing must begin.	0H

Register 16: HcLSThreshold (Read/Write) - Offset=44H with a 32-bit access

Bit	Function	Default
31:12	Reserved.	0H
11:0	LS Threshold: This field contains a value used by the Frame Management block to determine whether or not a low speed transaction can be started in the current frame.	0H

Register 17: HcRhDescriptorA (Read/Write) - Offset=48H with a 32-bit access

Bit	Function	Default
31:24	Power On To Power Good Time: Host Controller power switching is effective within 2 ms. The field value is represented as the number of 2 ms intervals. Only bits [25:24] are implemented as Read/Write. The remaining bits are Read Only as '0'. It is not expected that these bits be written to anything other than 1H, but limited adjustment is provided. This field should be written to support the system implementation. This field should always be written to a non-zero value.	01H
23:13	Reserved.	0H
12	No Over Current Protection: The Host Controller implements global over-current reporting 0: Over-current status is reported 1: Over-current status is not reported This bit should be written to support the external system port over-current implementation.	0H
11	Over Current Protection Mode: The Host Controller implements global over-current reporting 0: Global Over-Current 1: Individual Over-Current This bit is only valid when NoOverCurrentProtection is cleared. This bit should be written '0'.	0
10	Device Type: Read Only. The CY82C693U USB Host Controller is not a compound device.	0
9	No Power Switching: The Host Controller implements a global power switching mode. 0: Ports are power switched 1: Ports are always powered on This bit should be written to support the external system port power switching implementation.	0
8	Power Switching Mode: The Host Controller implements a global power switching mode. 0: Global Switching 1: Individual Switching This bit is only valid when NoPowerSwitching is cleared. this bit should be written '0'.	0

Register 17: HcRhDescriptorA (Read/Write) - Offset=48H with a 32-bit access (continued)

7:0	Number Downstream Ports: Read Only. The CY82C693U USB Host Controller supports two downstream ports.	02H

Note: This register is only reset by a power-on reset ($\overline{\text{PCIRST}}$). It is written during system initialization to configure the Root Hub. These bit should not be written during normal operation.

Register 18: HcRhDescriptorB (Read/Write) - Index=4CH with a 32-bit access

Bit	Function	Default
31:16	Port Power Control Mask: The Host Controller implements global-power switching. This field is only valid if NoPower-Switching is cleared and PowerSwitchingMode is set (individual port switching). When set, the port only responds to individual port power switching commands (Set/ClearPortPower). When cleared, the port only responds to global power switching commands (Set/ClearGlobalPower). 0: Device not removable 1: Global-power mask Port bit relationship: bit 16: Reserved bit 17: Port 1 bit 18: Port 2 ... bit 31: Port 15 Unimplemented ports are reserved, read/write ‘0’.	0000H
15:0	Device Removable: The Host Controller ports default to removable devices. 0: Device not removable 1: Device removable Port Bit relationship: 0: Reserved 1: Port 1 2: Port 2 ... 15: Port 15 Unimplemented ports are reserved, read/write ‘0’.	0000H

Note: THis register is only reset by power-on reset ($\overline{\text{PCIRST}}$). It is written during system initialization to configure the Root Hub. These bits should not be written during normal configuration.

Register 19: HcRhStatus (Read/Write) - Index=50H with a 32-bit access

Bit	Function	Default
31	Clear Remote Wakeup Enable: Write Only. Writing a '1' to this bit clears DeviceRemoteWakeupEnable. Writing a '0' has no effect.	0
30:18	Reserved.	0H
17	Over Current Indicator Change: This bit is set when OverCurrentIndicator changes. Writing a '1' clears this bit. Writing a '0' has no effect.	0
16	READ: Local Power Status Change. Not supported. Always read '0'. WRITE: Set Global Power. Writing a '1' issues a SetGlobalPower command to the ports. Writing a '0' has no effects.	0
15	READ: Device Remote Wakeup Enable. This bit enables ports' ConnectStatusChange as a remote wakeup event: 0: Disabled 1: Enabled WRITE: Set Remote Wakeup Enable. Writing a '1' sets DeviceRemoteWakeupEnable. Writing a '0' has no effect.	0
14:2	Reserved.	0H
1	Over Current Indicator: This bit reflects the state of the OVRCUR pin. This field is only valid if NoOverCurrentProtection and OverCurrentProtectionMode are cleared. 0: No over-current condition 1: Over-current condition	-
0	READ: Local Power Status. Not supported. Always read '0'. WRITE: Clear Global Power. Writing a '1' issues a ClearGlobalPower command to the ports. Writing a '0' has no effect.	0

Note: This register is reset by the USBReset state.

Register 20: HcRhPortStatus[1:2] (Read/Write) - Index=54H, 58H with a 32-bit access

Bit	Function	Default
31:21	Reserved.	0H
20	Port Reset Status Change: This bit indicates that the port reset signal has completed. 0: Port reset is not complete 1: Port reset is complete	0
19	Port Over Current Indicator Change: This bit is set when OverCurrentIndicator changes. Writing a '1' clears this bit. Writing a '0' has no effect.	0
18	Port Suspend Status Change: This bit indicates the completion of the selective resume sequence for the port. 0: Port is not resumed 1: Port resume is complete	0
17	Port Enable Status Change: This bit indicates that the port has been disabled due to a hardware event (cleared PortEnableStatus). 0: Port has not been disabled 1: PortEnableStatus has been cleared	0
16	Connect Status Change: This bit indicates a connect or disconnect event has been detected. Writing a '1' clears this bit. Writing a '0' has no effect. 0: No connect/disconnect event 1: Hardware detection of connect/disconnect event Note: If DeviceRemovable is set, this bit resets to '1'.	0
15:10	Reserved.	0H
9	READ: Low Speed Device Attached. This bit defines the speed (and bus idle) of the attached device. It is only valid when CurrentConnectStatus is set. 0: Full Speed device 1: Low Speed device WRITE: Clear Port Power. Writing a '1' clears PortPowerStatus. Writing a '0' has no effect.	0
8	READ: Port Power Status. This bit reflects the power state of the port regardless of the power switching mode. 0: Port power is off 1: Port power is on Note: if NoPowerSwitching is set, this bit is always read as '1'. WRITE: Set Port Power. Writing a '1' sets PortPowerStatus. Writing a '0' has no effect.	0
7:5	Reserved.	0H
4	READ: Port Reset Status. 0: Port reset signal is not active 1: Port reset signal is active WRITE: Set Port Reset. Writing a '1' sets PortResetStatus. Writing a '0' has no effect.	0
3	READ: Port Over Current Indicator. The Host Controller supports global over-current reporting. This bit reflects the state of the OVRCUR pin dedicated to this port. This field is only valid if NoOverCurrentProtection is cleared and OverCurrentProtectionMode is set. 0: No over-current condition 1: Over-current condition WRITE: Clear Port Suspend. Writing a '1' initiates the selective resume sequence for the port. Writing a '0' has no effect.	0

Register 20: HcRhPortStatus[1:2] (Read/Write) - Index=54H, 58H with a 32-bit access (continued)

Bit	Function	Default
2	READ: Port Suspend Status. 0: Port is not suspended 1: Port is selectively suspended WRITE: Set Port Suspend. Writing a '1' sets PortSuspendStatus. Writing a '0' has no effect	0
1	READ: Port Enable Status. 0: Port disabled 1: Port enabled WRITE: Set Port Enable. Writing a '1' sets PortEnableStatus. Writing a '0' has no effect.	0
0	READ: Current Connect Status. 0: No device connected 1 Device connected Note: If DeviceRemovable is set (not removable) this bit is always '1'. WRITE: Clear Port Enable. Writing a '1' clears PortEnableStatus. Writing a '0' has no effect.	0

Note: This register is reset by the USBReset state.

Register 21: HceControl (Read/Write) - Index=100H with a 32-bit access

Bit	Function	Default
31:9	Reserved.	0H
8	A20 State: Indicates current state of Gate A20 on keyboard controller. Used to compare against value written to 60H when GateA20Sequence is active.	0
7	IRQ12 Active: Indicates that a positive transition on IRQ12 from keyboard controller has occurred. Software may write a '1' to this bit to clear it (set it to '0'). Software write of a '0' to this bit has no effect.	0
6	IRQ1 Active: Indicates that a positive transition on IRQ1 from keyboard controller has occurred. Software may write a '1' to this bit to clear it (set it to '0'). Software write of a '0' to this bit has no effect.	0
5	Gate A20 Sequence: Set by the Host Controller when a data value of D1H is written to I/O port 64H. Cleared by the Host Controller on write to I/O port 64H of any value other than D1H.	0
4	External IRQEn: When set to '1', IRQ1 and IRQ12 from the keyboard controller will cause an emulation interrupt. The function controlled by this bit is independent of the setting of the EmulationEnable bit in this register.	0
3	IRQEn: When set the Host Controller will generate IRQ1 or IRQ12 as long as the OutputFull bit in HceStatus is set to '1'. If the AuxOutputFull bit of HceStatus is '0' then IRQ1 is generated and if it is '1', then an IRQ12 is generated.	0
2	Character Pending: When set, an emulation interrupt will be generated when the OutputFull but of the HceStatus register is set to '0'.	0
1	Emulation Interrupt: This bit is a static decode of the emulation interrupt condition.	0
0	Emulation Enable: When set to '1' the Host Controller will be enabled for legacy emulation. The Host Controller will decode accesses to I/O registers 60H and 64H and generate IRQ1 and/or IRQ12 when appropriate. Additionally, the Host Controller will generate an emulation interrupt at appropriate times to invoke the emulation software.	0

Note: This register is used to enable and control the legacy keyboard and mouse emulation hardware and report various status information.

Register 22: HceInput - Index=104H with a 32-bit access

Bit	Function	Default
31:8	Reserved.	0H
7:0	Input Data: This register holds data that is written to I/O ports 60H and 64H.	-

Note: This register is the emulation side of the legacy Input Buffer Register.

Register 23: HceOutput - Index=108H with a 32-bit access

Bit	Function	Default
31:8	Reserved.	0H
7:0	Output Data: This register hosts data that is returned when an I/O read of port 60H is performed by application software.	-

Note: This register is the emulation side of the legacy Output Buffer register where keyboard and mouse data is to be read by software.

Register 24: HceStatus (Read/Write) - Index=10CH with a 32-bit access

Bit	Function	Default
31:8	Reserved.	0H
7	Parity: Indicates parity error on keyboard/mouse data.	0
6	Timeout: Used to indicate a time-out.	0
5	Aux Output Full: IRQ12 is asserted whenever this bit is set to ‘1’ and OutputFull is set to ‘1’ and the IRQEn bit is set.	0
4	Inhibit Switch: This bit reflects the state of the keyboard inhibit switch and is set if the keyboard is NOT inhibited.	0
3	Cmd Data: The Host Controller will set this bit to ‘0’ on an I/O write to port 60H and on an I/O write to port 64H the Host Controller will set this bit to ‘1’.	0
2	Flag: Nominally used as a system flag by software to indicate a warm or cold boot.	0
1	Input Full: Except for the case of a Gate A20 sequence, this bit is set to ‘1’ on an I/O write to address 60H or 64H. While this bit is set to ‘1’ and emulation is enabled, an emulation interrupt condition exists.	0
0	Output Full: The Host Controller will set this bit to ‘0’ on a read of I/O port 60H. If IRQEn is set and AuxOutputFull is set to ‘0’ then an IRQ1 is generated as long as this bit is set to ‘1’. If IRQEn is set and AuxOutputFull is set to ‘1’ then an IRQ12 will be generated as long as this bit is set to ‘1’. While this bit is ‘0’ and CharacterPending in HceControl is set to ‘1’, an emulation interrupt condition exists.	0

Note: This register is the emulation side of the legacy Status register.

Maximum Ratings

(Above which the useful life may be impaired. For user guidelines, not tested.)

Supply Voltage (V_{CC}) .....+7 V

Ambient Operating Temperature .....–25°C to +70°C

Ambient Storage Temperature .....–40°C to 125°C

DC Voltage Applied to Outputs ..... –0.5V to V_{DD}

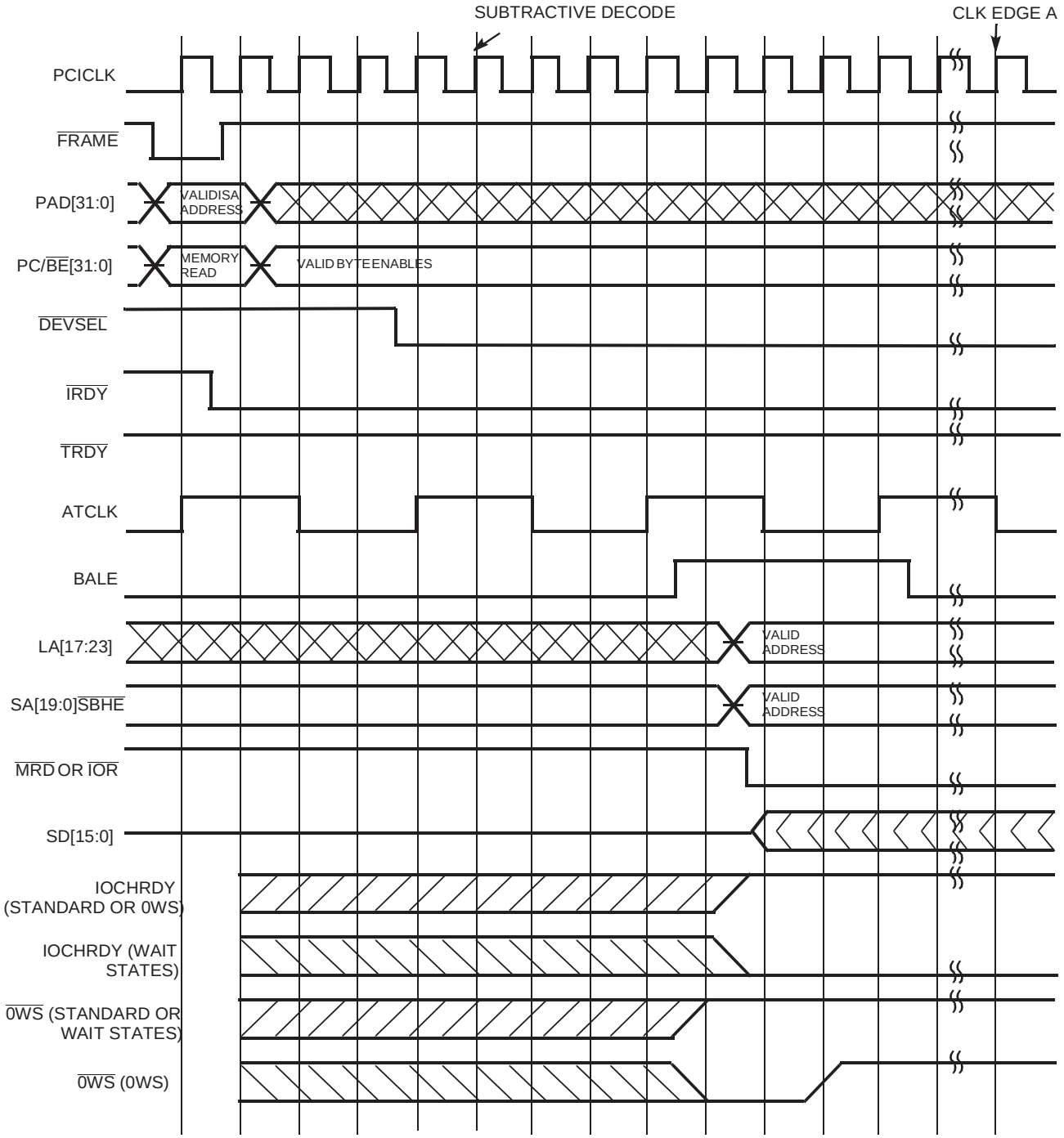
DC Input Voltage ..... –0.5V to V_{DD}

Electrical Characteristics Over the Operating Range (T_A=0°C to 70°C)

Parameter	Description		Min.	Max.	Unit
V_{CC}	Core Supply Voltage		4.5	5.5	V
V_{DD}	3.3V I/O Supply Voltage		3.0	V_{CC}	V
V_{IL}	Input LOW Voltage		–0.5	0.8	V
V_{IH}	Input HIGH Voltage		2.0	V_{DD}+0.5	V
V_{OL}	Output LOW Voltage			0.4	V
V_{OH}	Output HIGH Voltage		2.4		V
I_{IL}	Input Leakage Current			10	µA
I_{OL}	Output Leakage			10	µA
C_{IN}	Input Capacitance			10	pF
C_{OUT}	Output Capacitance			10	pF
I_{CC}	Power Supply Current	66 MHz		TBD	mA

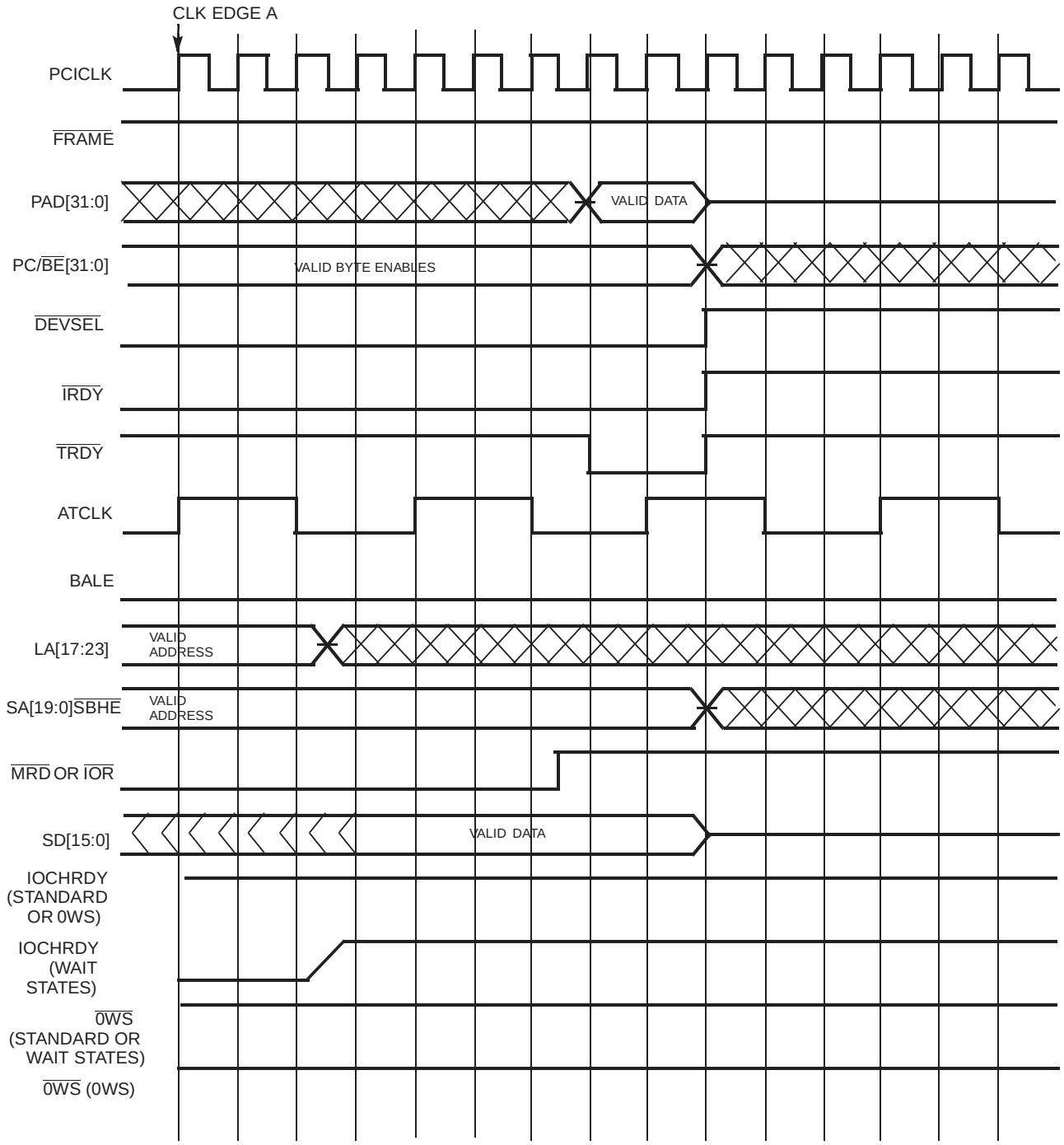
Switching Waveforms

PCI to ISA RD Cycle – No Data Packing (Part 1 of 2)



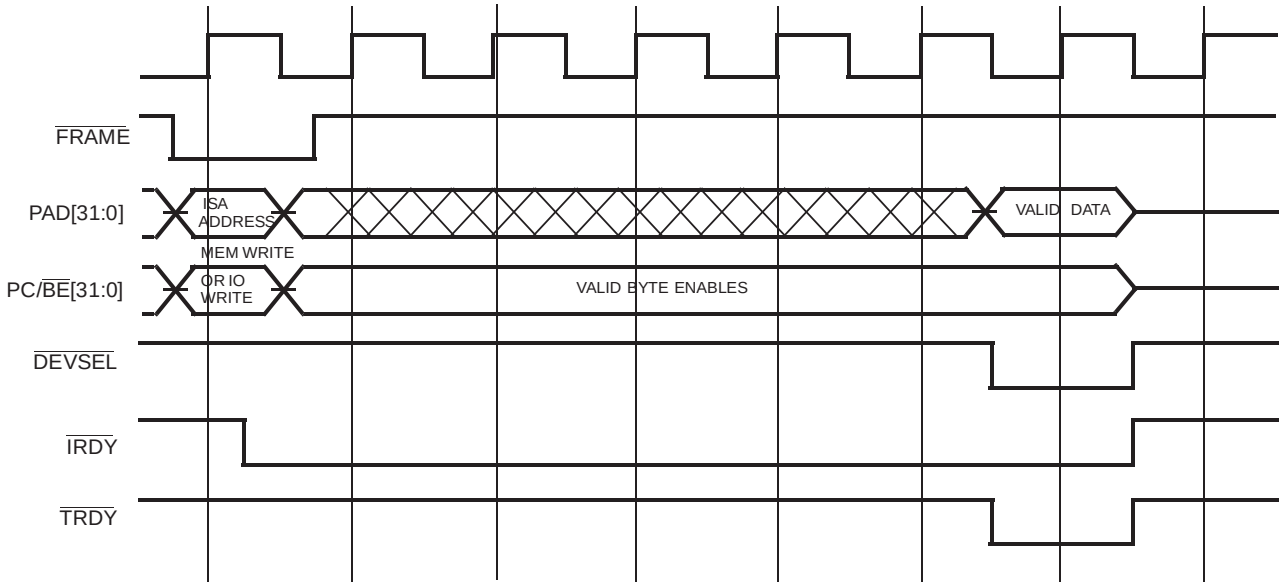
Switching Waveforms (continued)

PCI to ISA RD Cycle – No Data Packing (Part 2 of 2)



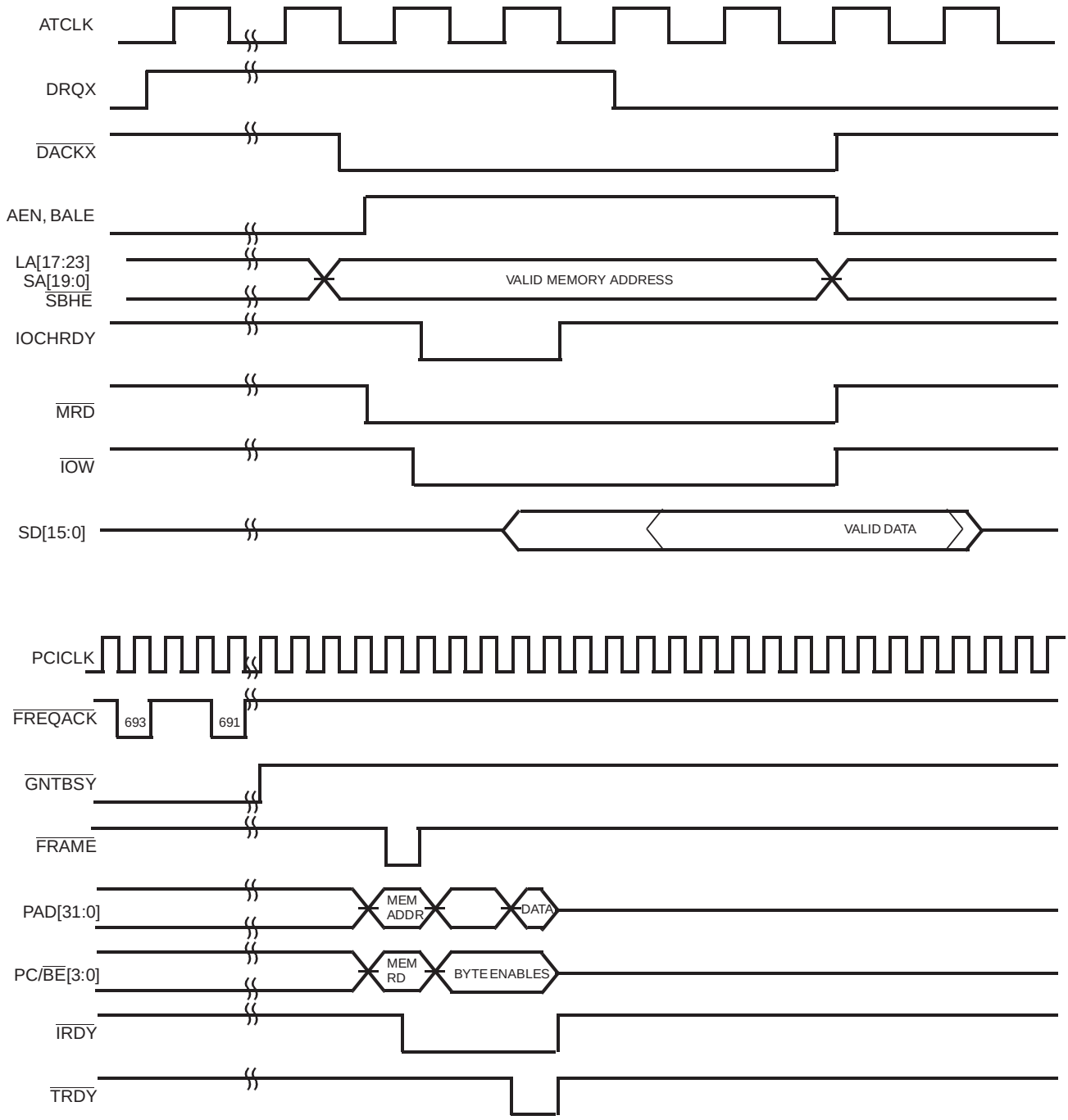
Switching Waveforms (continued)

PCI Write to ISA Post Write Buffer (Subtractive Decode Set to 6 CLKS)



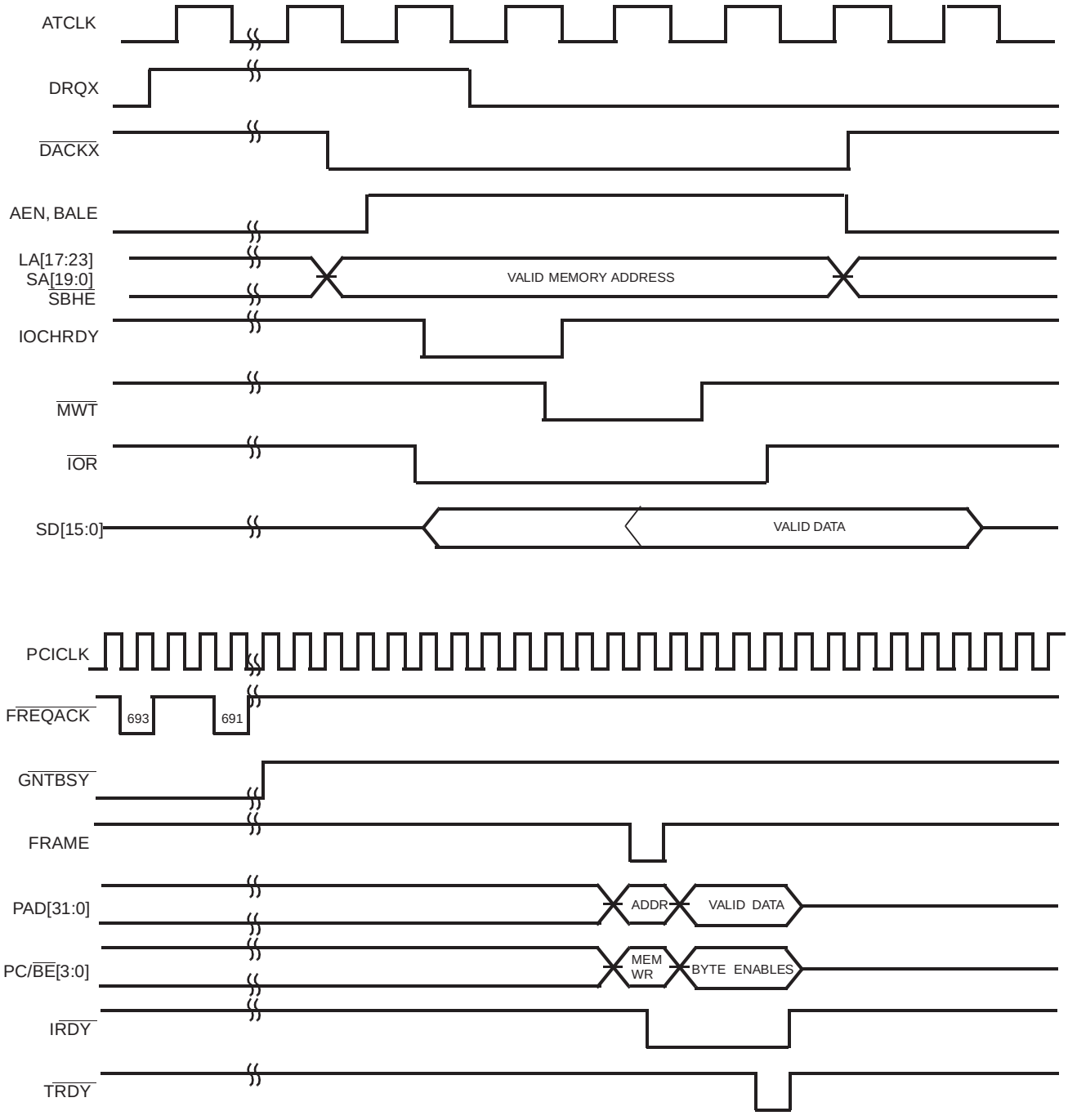
Switching Waveforms (continued)

DMA/Master Operation (Memory Read/IO Write)



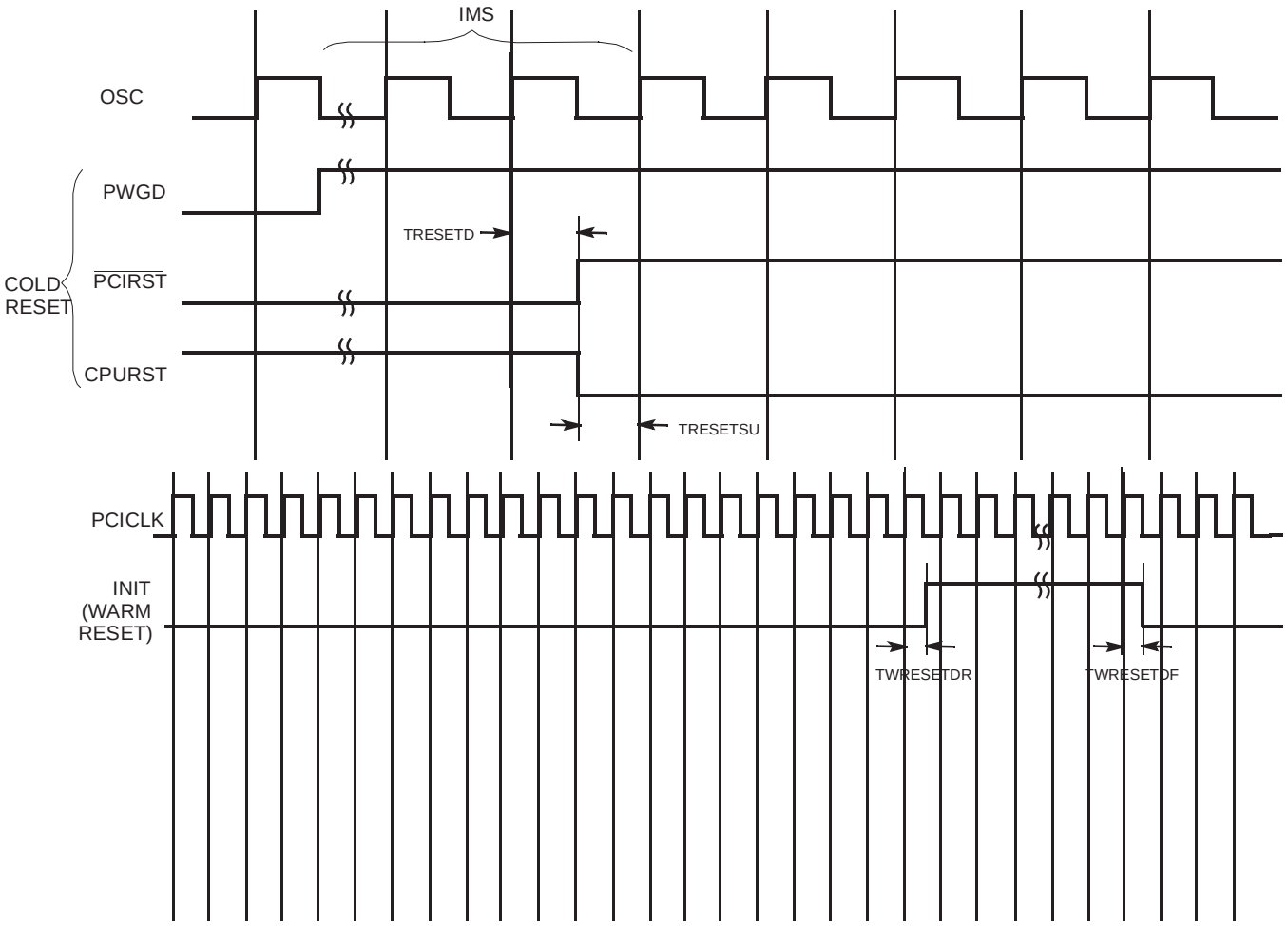
Switching Waveforms (continued)

DMA/Master Operation (IO Read/Memory Write)



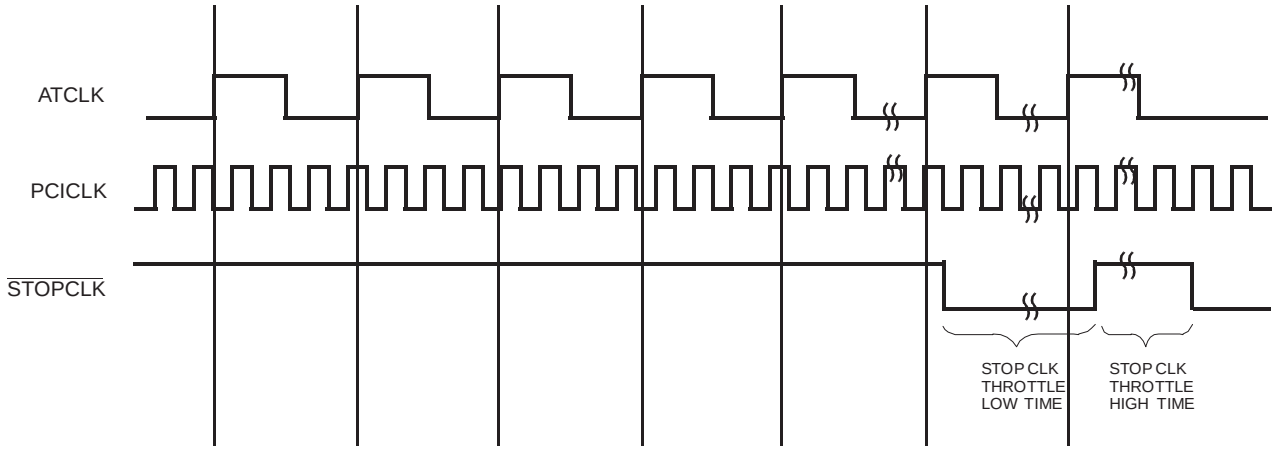
Switching Waveforms (continued)

Reset

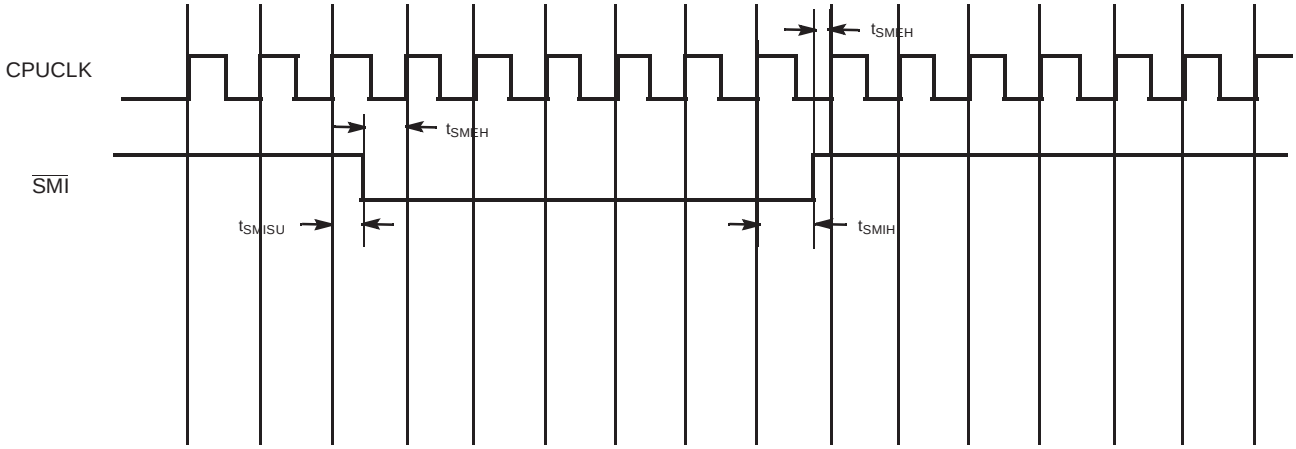


Switching Waveforms (continued)

Power Management (Hardware Controlled)

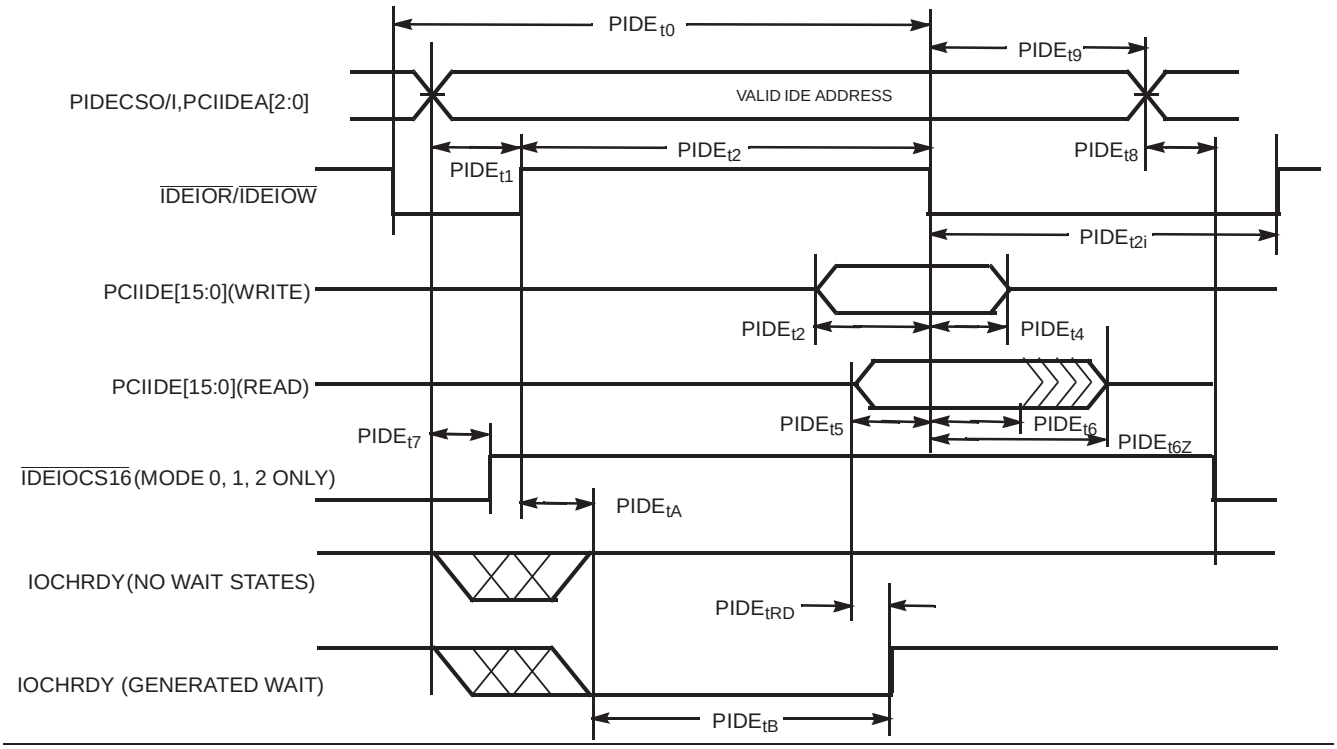


Power Management (Software Controlled)

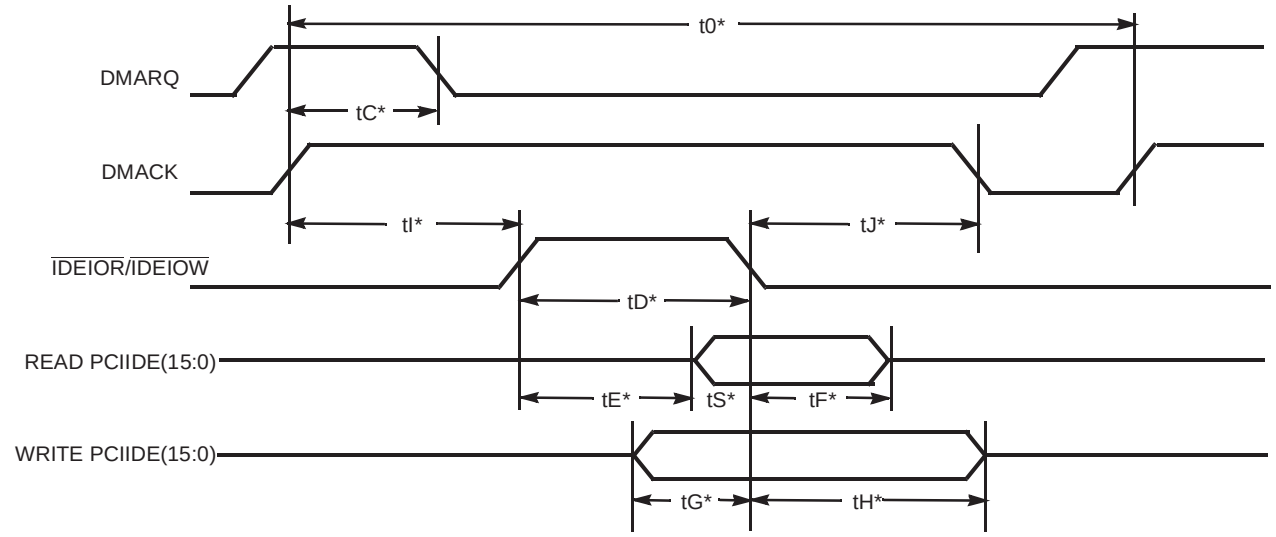


Switching Waveforms (continued)

IDE PIO Transfer



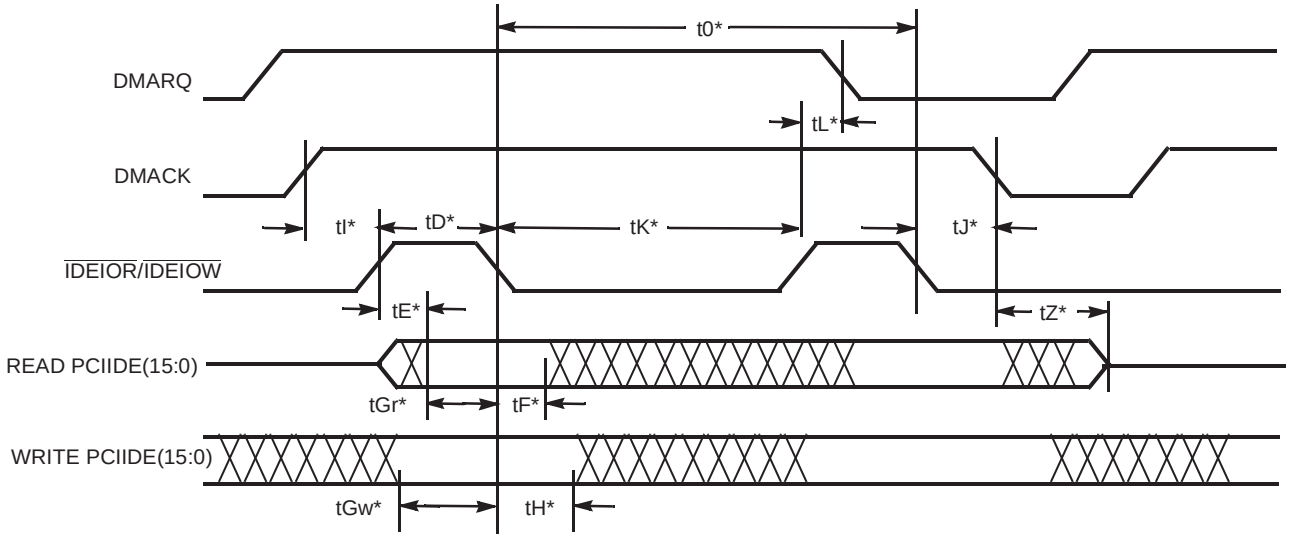
Single Word DMA Transfer



*Preface all of these indices with SWIPEDMA (e.g. TH becomes SWPIDEDMA$_{tH}$)

Switching Waveforms (continued)

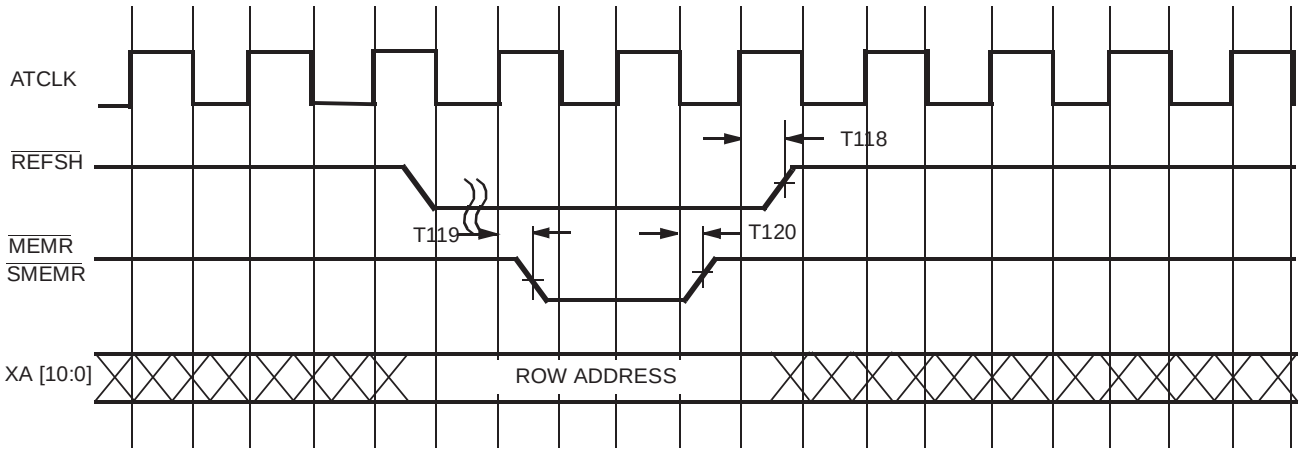
Multiword DMA Transfer



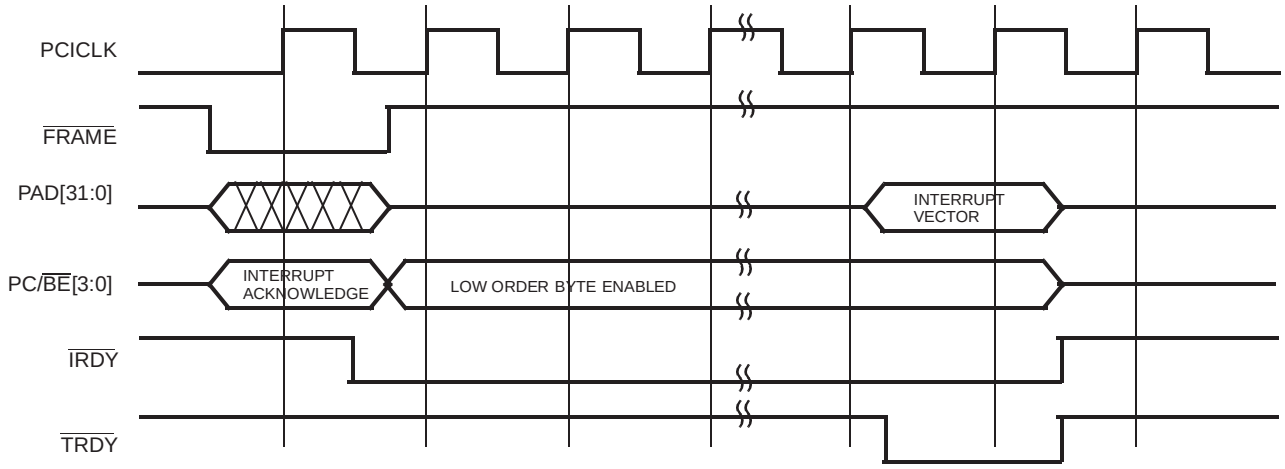
*Preface all of these indices with SWIPEDMA (e.g. TH becomes $MWPIDEDMA_{tH}$)

Switching Waveforms (continued)

AT Refresh Timing

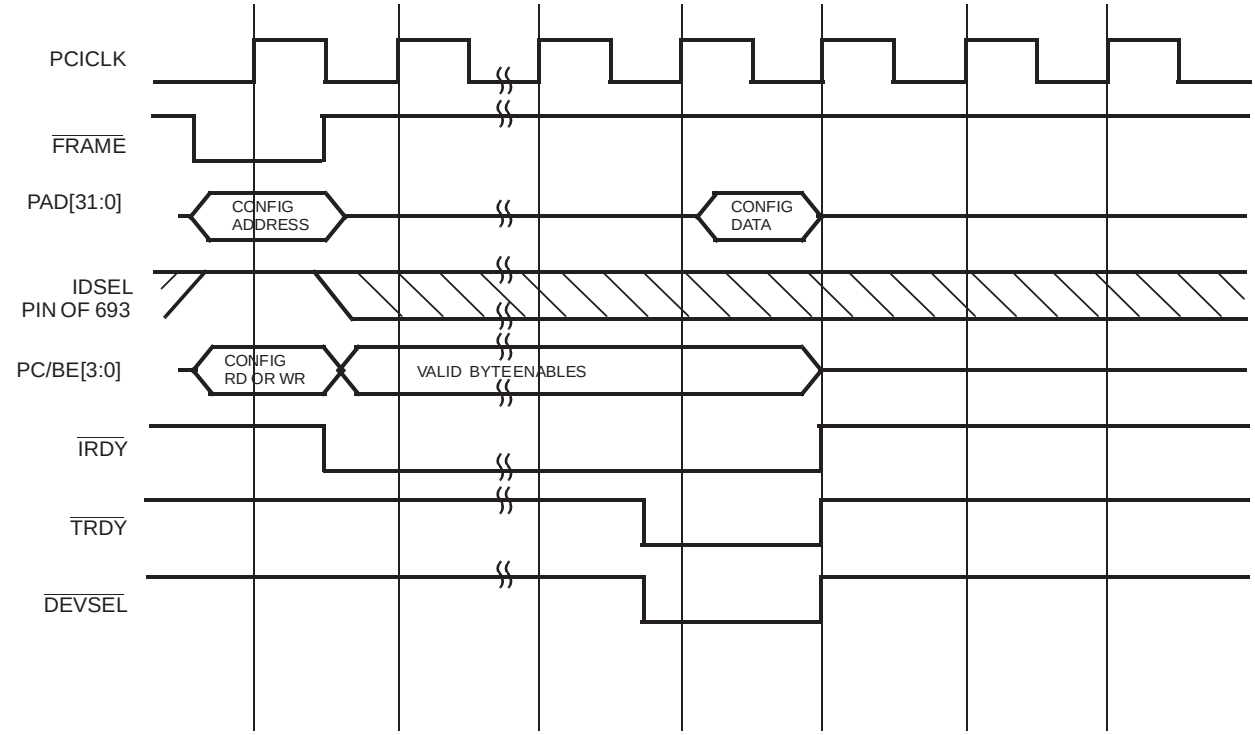


Interrupt Acknowledge Cycle



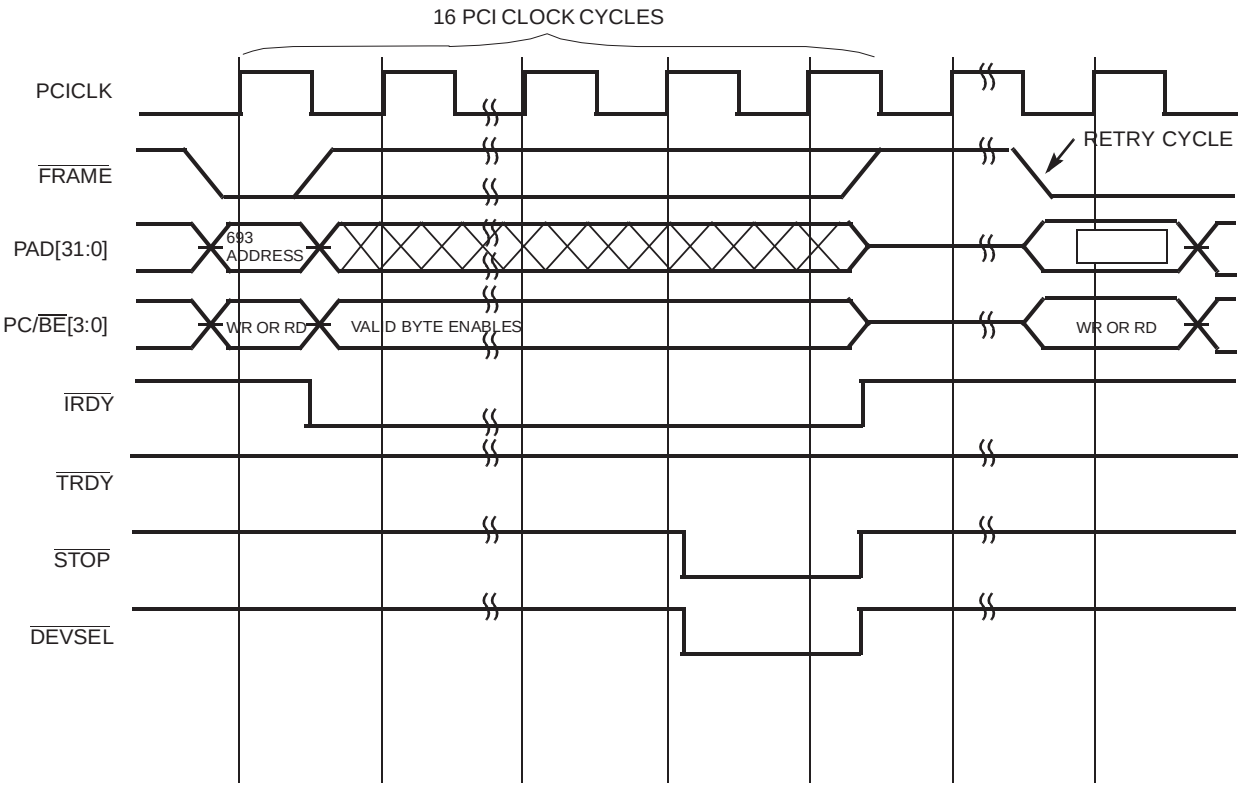
Switching Waveforms (continued)

CY82C693 Configuration Access



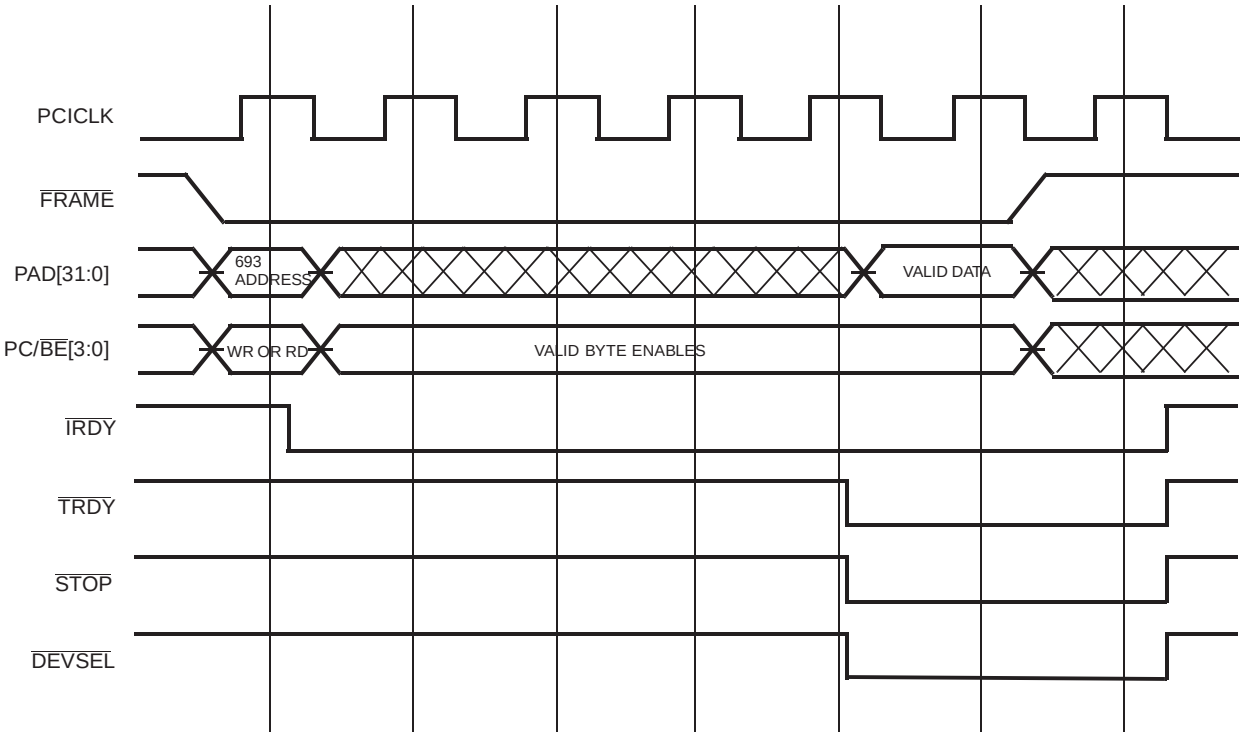
Switching Waveforms (continued)

693 Generated Retry (For Accesses Requiring More Than 16 PCI Clock Cycles to Complete)



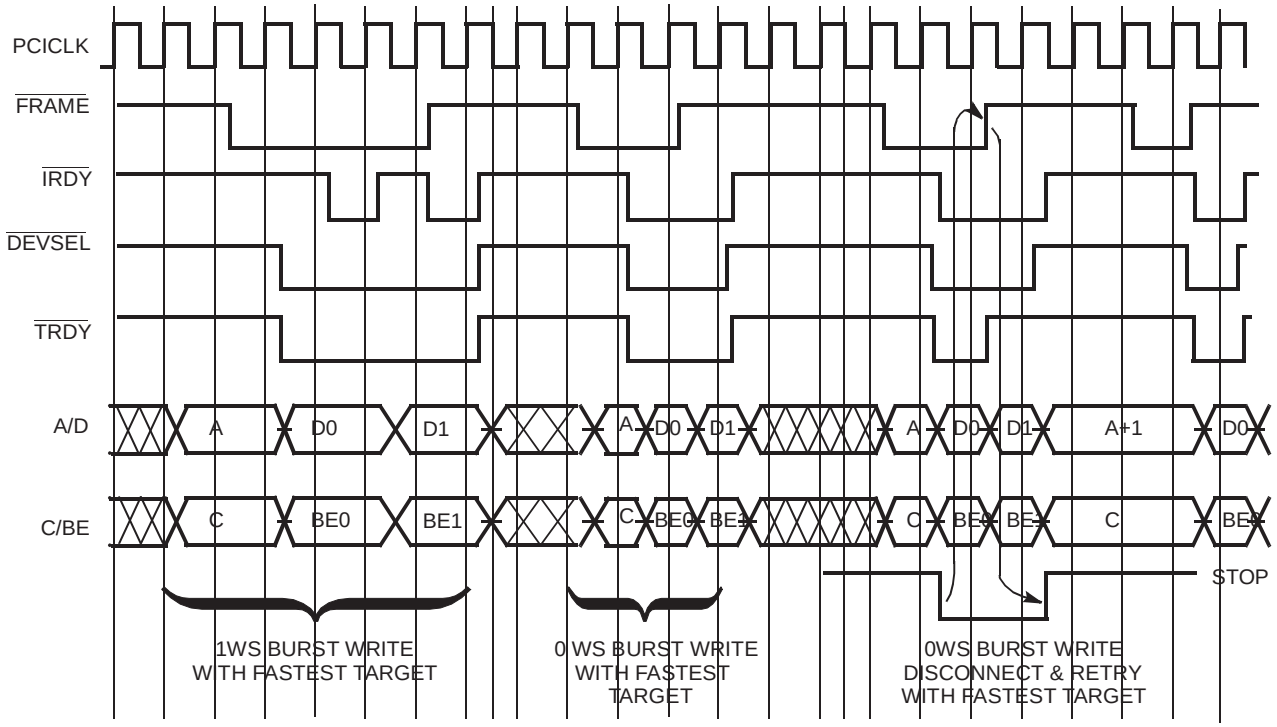
Switching Waveforms (continued)

693 PCI Disconnect on a NON–Burst Cycle (If Initiator does not Deassert Frame)



Switching Waveforms (continued)

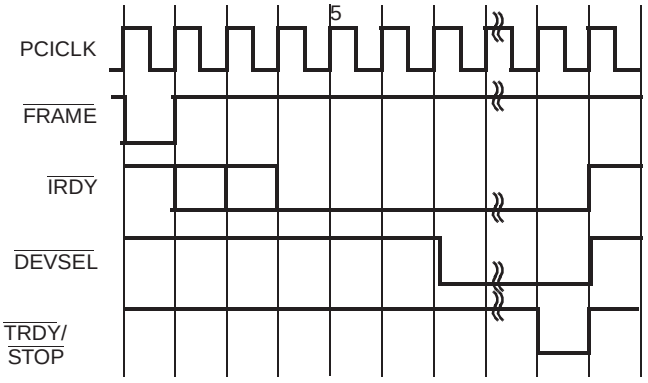
ISA Master Memory Pre-Read



82C693U–4

Switching Waveforms (continued)

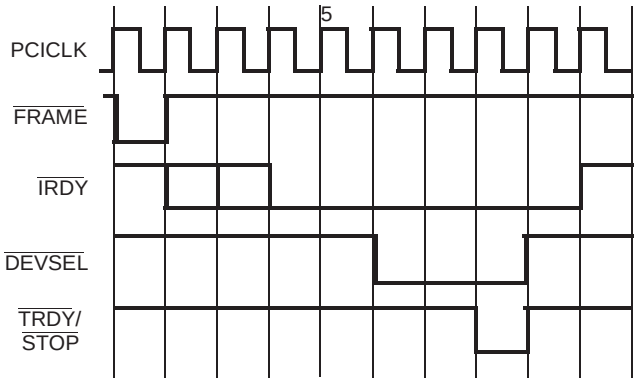
PCI Master Subtractive Decode "DEVSEL" Timing



(SUB–DECODE=6PCICLK)

82C693U–5

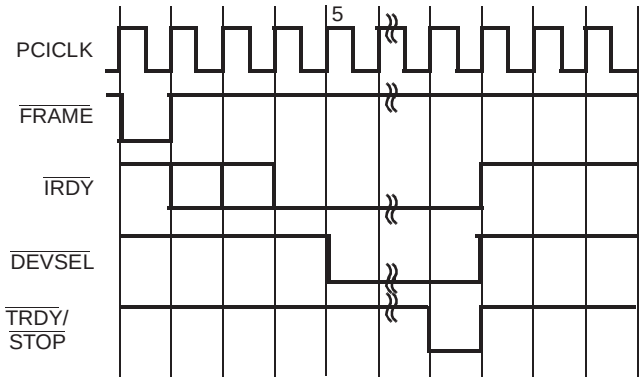
PCI Master Subtractive Decode "DEVSEL" Timing



(SUB–DECODE=5PCICLK)

82C693U–6

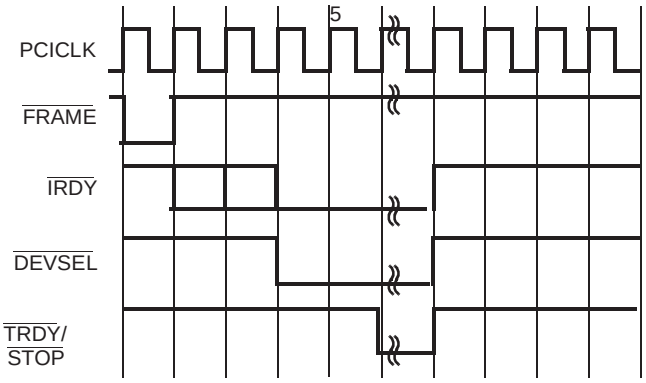
PCI Master Subtractive Decode "DEVSEL" Timing



(SUB–DECODE=4PCICLK)

82C693U–7

PCI Master Subtractive Decode "DEVSEL" Timing



(SUB–DECODE=3PCICLK)

82C693U–8

Ordering Information

Ordering Code	Package Name	Package Type	Operating Range
CY82C693U-NC	N208	208-Lead Plastic Quad Flat Pack	Commercial

Document #: 38-00598

Package Diagram

208-Lead Plastic Quad Flatpack N208