

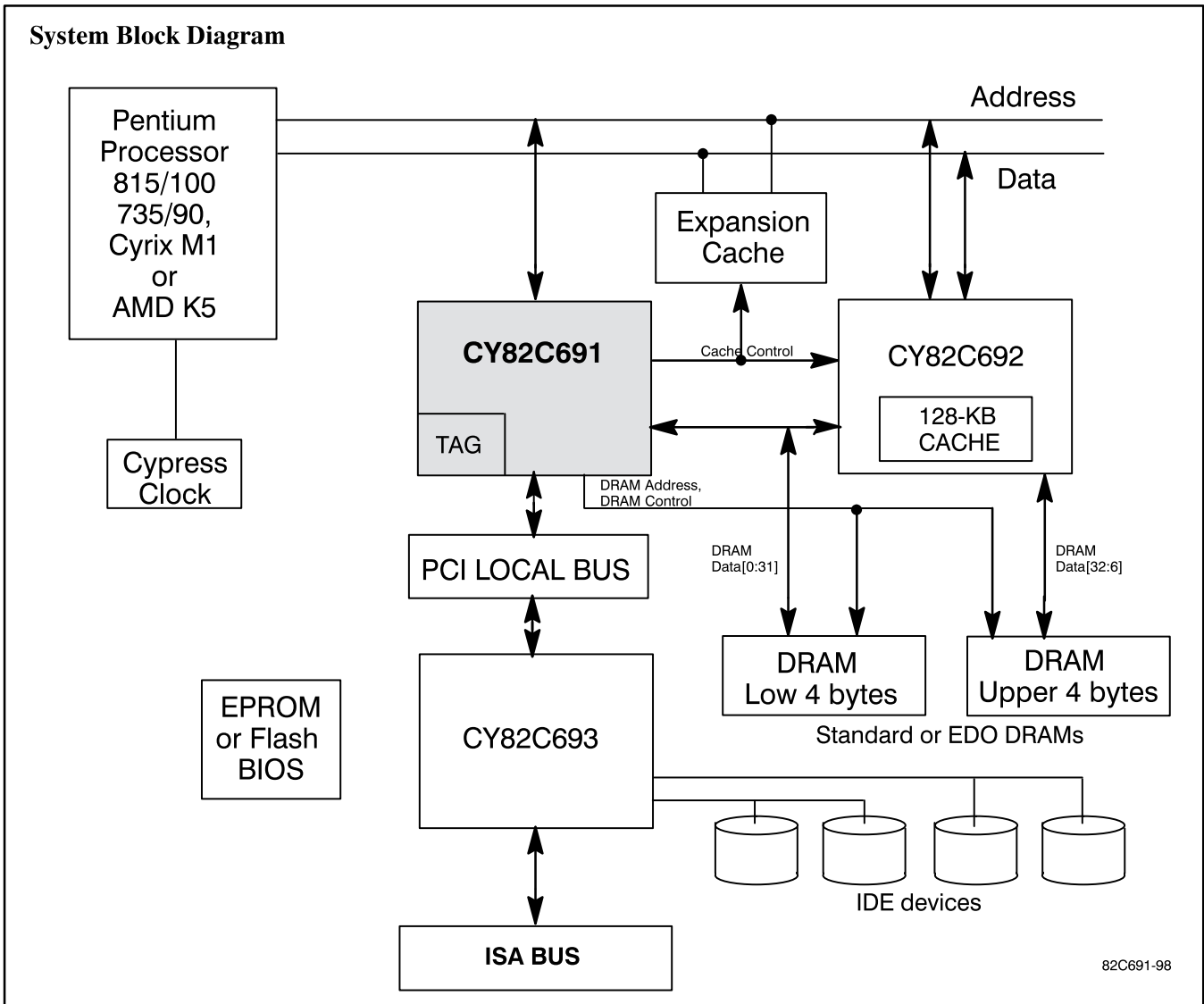


Pentium™ hyperCache™ Chipset System Controller

Features

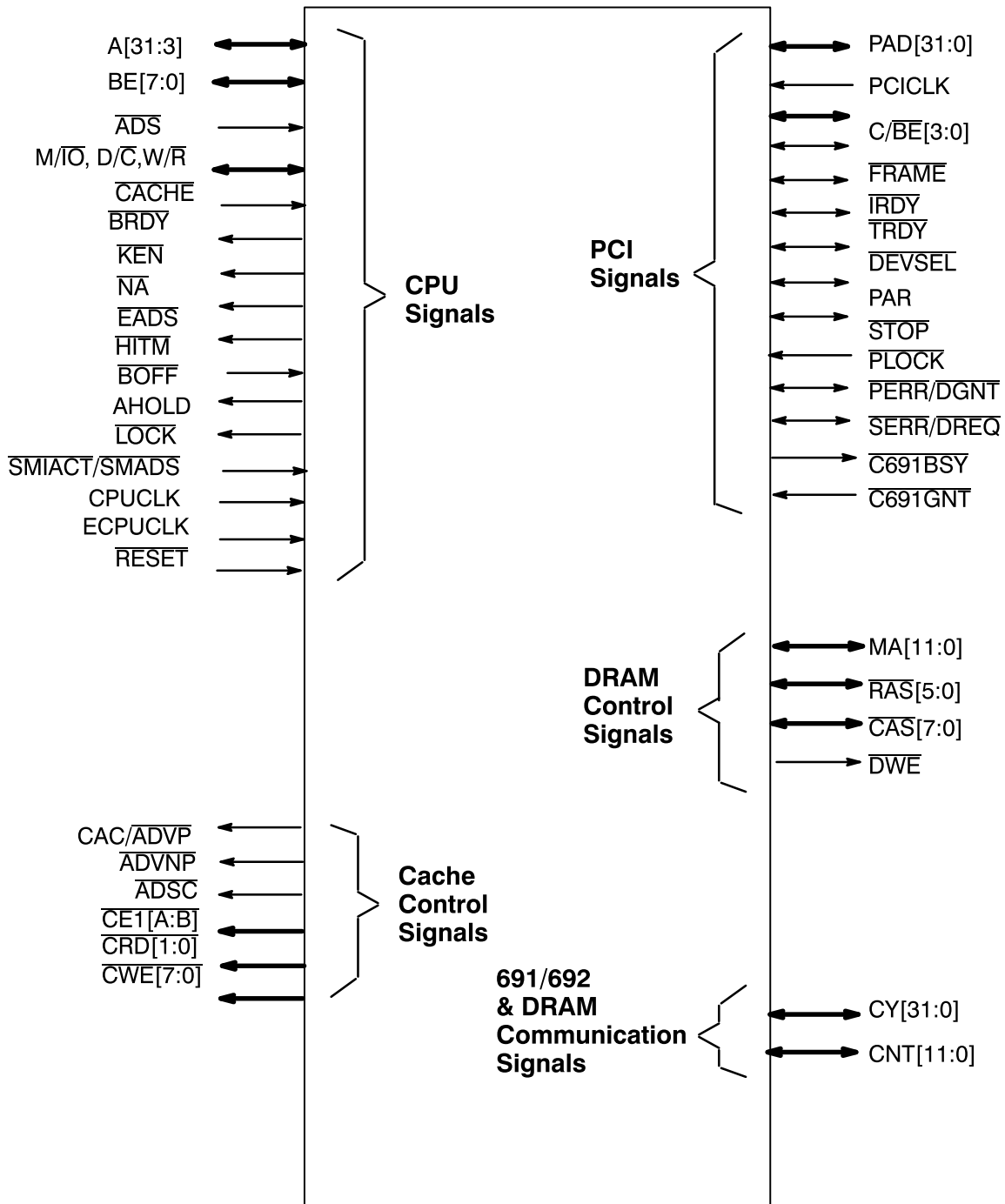
- Provides control for the cache, system memory, and the PCI bus
- PCI Bus Rev. 2.1 compliant
- Supports Pentium™ (ICOMP 735/90, 815/100, etc.), AMD K5, and Cyrix M1 CPUs
- Support for WB or WT L1 cache
- Supports 3-1-1 burst read/write operation at 66, 60, and 50 MHz bus speeds
- Supports CPU address pipelining
- Support for synchronous or pipelined BSRAMs
- Supports synchronous or asynchronous PCI operation
- Provides power management support through SMM
- Integrated 8Kx21 tag (direct mapped and two-way set associative)
- Support for cache sizes up to 1 MB
- Supports mixed standard page-mode and EDO DRAMs
- Supports shared DRAM bus
- Support for 72-bit-wide DRAM banks
- Supports non-symmetrical DRAM banks
- Supports six banks of DRAM (six RAS lines)
- Supports DRAM densities up to 16 Mb
- Up to 768 MB main memory
- 24 mA variable drive on DRAM address lines
- Provides glueless (0 TTL) system solution with CY82C692 and CY82C693
- Support for concurrent operation among CPU, cache, DRAM, and PCI
- Provides PCI post writing and pre-reading
- Packaged in a 208-pin PQFP

System Block Diagram



82C691-98

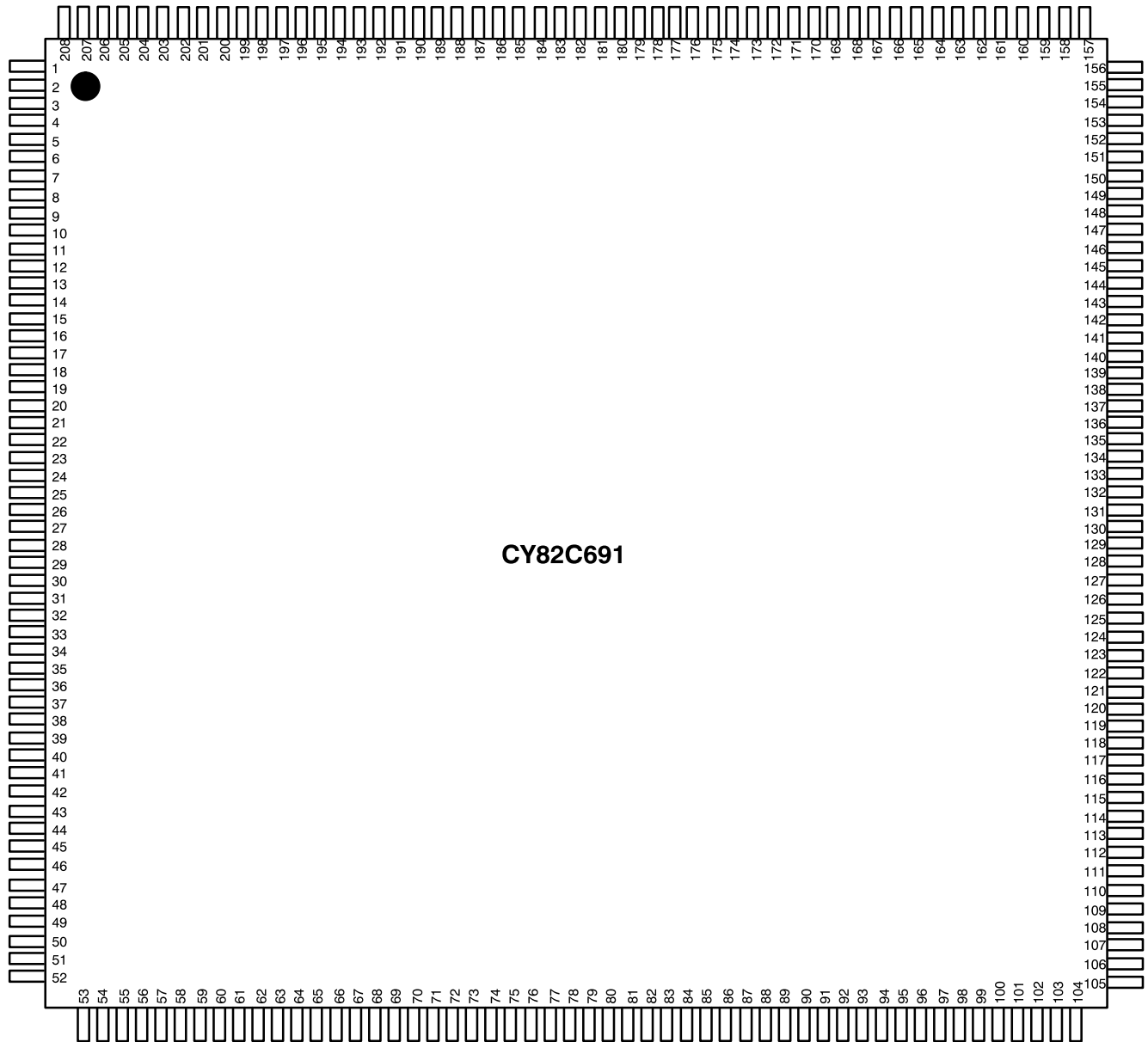
Pentium is a trademark of Intel Corporation.
hyperCache is a trademark of Cypress Semiconductor Corporation.

CY82C691 Signals


82C691-99

Pin Configuration

208-pin PQFP
Top View



82C691-100



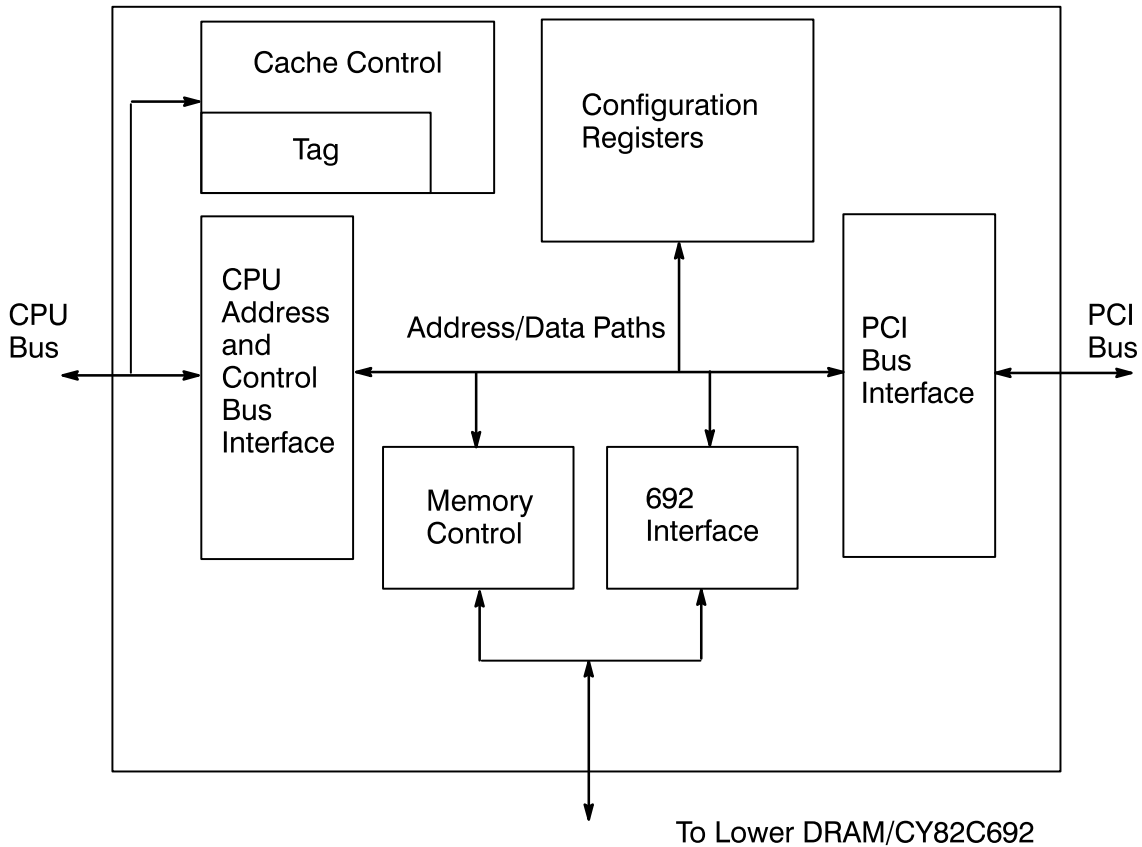
CY82C691 Pin Reference (In Numerical Order by Pin Number)

Pin No	Pin Name	Pin No	Pin Name	Pin No	Pin Name	Pin No	Pin Name	Pin No	Pin Name
1	CY13	43	A23	85	SERR/DREQ	127	PAD7	169	+3.3V
2	CY12	44	A24	86	C691GNT	128	PAD6	170	CWE5
3	CY11	45	A25	87	C691BSY	129	PAD5	171	CWE4
4	CY10	46	A26	88	STOP	130	PAD4	172	CWE3
5	CY9	47	A27	89	PAR	131	PAD3	173	CWE2
6	CY8	48	A28	90	DEVSEL	132	PAD2	174	CWE1
7	CY7	49	A29	91	+5V	133	PAD1	175	CWE0]
8	CY6	50	A30	92	TRDY	134	PAD0]	176	RESET
9	CY5	51	A31	93	GND	135	PCICLK	177	CNT11
10	CY4	52	MA11	94	IRDY	136	+5V	178	CNT10
11	CY3	53	MA10	95	FRAME	137	GND	179	CNT9
12	CY2	54	MA9	96	PC/BE3	138	CPUCLK	180	CNT8
13	CY1	55	MA8	97	PC/BE2	139	ECPUCLK	181	CNT7
14	CY0	56	MA7	98	PC/BE1	140	ADSC	182	CNT6
15	GND	57	MA6	99	PC/BE0	141	BRDY	183	CNT5
16	A3	58	MA5	100	PAD31	142	CAC/ADVP	184	CNT4
17	A4	59	MA4	101	PAD30	143	ADVNP	185	CNT3
18	A5	60	MA3	102	PAD29	144	D/C	186	CNT2
19	A6	61	MA2	103	PAD28	145	M/IO	187	CNT1
20	+5V	62	MA1	104	PAD27	146	ADS	188	CNT0
21	GND	63	MA0	105	GND	147	BE7	189	CY31
22	A7	64	RAS5	106	PAD26	148	BE6	190	CY30
23	HLDA/LOCK	65	+5V	107	PAD25	149	BE5	191	CY29
24	A8	66	RAS4	108	PAD24	150	BE4	192	CY28
25	HITM	67	GND	109	PAD23	151	GND	193	C[27
26	A9	68	RAS3	110	PAD22	152	BE3	194	CY26
27	A10	69	RAS2	111	PAD21	153	BE2	195	CY25
28	A11	70	RAS1	112	PAD20	154	BE1	196	CY24
29	A12	71	RAS0	113	PAD19	155	BE0	197	GND
30	A13	72	DWE	114	PAD18	156	HOLD/AHOLD	198	CY23
31	A14	73	CAS7	115	PAD17	157	BOFF	199	CY22
32	+3.3V	74	CAS6	116	PAD16	158	EADS	200	CY21
33	GND	75	CAS5	117	PAD15	159	NA	201	CY20
34	A15	76	+5V	118	PAD14	160	KEN	202	CY19
35	A16	77	CAS4	119	PAD13	161	W/R/INV	203	CY18
36	A17	78	CAS3	120	PAD12	162	SMIACT/SMIADS	204	CY17
37	A18	79	CAS2	121	PAD11	163	CE1B	205	CY16
38	A19	80	CAS1	122	PAD10	164	CE1A	206	CY15
39	A20	81	GND	123	PAD9	165	CRD1	207	CY14
40	A21	82	CAS0	124	+5V	166	CRD0	208	+3.3V
41	A22	83	PERR/DGNT	125	GND	167	CWE7		
42	CACHE	84	PLOCK	126	PAD8	168	CWE6		



CY82C691 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.
A3	16	BE7	147	$\overline{\text{CWE7}}$	167	GND	15,21,33,67,81,93, 105,125,137,151,197	PAD19	113
A4	17	$\overline{\text{BOFF}}$	157	CY0	14	$\overline{\text{HITM}}$	25	PAD20	112
A5	18	$\overline{\text{BRDY}}$	141	CY1	13	HLDA/ $\overline{\text{LOCK}}$	23	PAD21	111
A6	19	$\overline{\text{C69IBSY}}$	87	CY2	12	HOLD/ AHOLD	156	PAD22	110
A7	22	$\overline{\text{C69IGNT}}$	86	CY3	11	$\overline{\text{IRDY}}$	94	PAD23	109
A8	24	CAC/ADV $\overline{\text{P}}$	142	CY4	10	$\overline{\text{KEN}}$	160	PAD24	108
A9	26	$\overline{\text{CACHE}}$	42	CY5	9	MA0	63	PAD25	107
A10	27	$\overline{\text{CAS0}}$	82	CY6	8	MA1	62	PAD26	106
A11	28	$\overline{\text{CAS1}}$	80	CY7	7	MA2	61	PAD27	104
A12	29	$\overline{\text{CAS2}}$	79	CY8	6	MA3	60	PAD28	103
A13	30	$\overline{\text{CAS3}}$	78	CY9	5	MA4	59	PAD29	102
A14	31	$\overline{\text{CAS4}}$	77	CY10	4	MA5	58	PAD30	101
A15	34	$\overline{\text{CAS5}}$	75	CY11	3	MA6	57	PAD31	100
A16	35	$\overline{\text{CAS6}}$	74	CY12	2	MA7	56	PAR	89
A17	36	$\overline{\text{CAS7}}$	73	CY13	1	MA8	55	PC/ $\overline{\text{BE0}}$	99
A18	37	$\overline{\text{CE1A}}$	164	CY14	207	MA9	54	PC/ $\overline{\text{BE1}}$	98
A19	38	$\overline{\text{CE1B}}$	163	CY15	206	MA10	53	PC/ $\overline{\text{BE2}}$	97
A20	39	CNT0	188	CY16	205	MA11	52	PC/ $\overline{\text{BE3}}$	96
A21	40	CNT1	187	CY17	204	M/ $\overline{\text{IO}}$	145	PCICLK	135
A22	41	CNT2	186	CY18	203	$\overline{\text{NA}}$	159	$\overline{\text{PERR/DGNT}}$	83
A23	43	CNT3	185	CY19	202	PAD0	134	PLOCK	84
A24	44	CNT4	184	CY20	201	PAD1	133	$\overline{\text{RAS0}}$	71
A25	45	CNT5	183	CY21	200	PAD2	132	$\overline{\text{RAS1}}$	70
A26	46	CNT6	182	CY22	199	PAD3	131	$\overline{\text{RAS2}}$	69
A27	47	CNT7	181	CY23	198	PAD4	130	$\overline{\text{RAS3}}$	68
A28	48	CNT8	180	CY24	196	PAD5	129	$\overline{\text{RAS4}}$	66
A29	49	CNT9	179	CY25	195	PAD6	128	$\overline{\text{RAS5}}$	64
A30	50	CNT10	178	CY26	194	PAD7	127	$\overline{\text{RESET}}$	176
A31	51	CNT11	177	CY27	193	PAD8	126	$\overline{\text{SERR/DREQ}}$	85
$\overline{\text{ADS}}$	146	CPUCLK	138	CY28	192	PAD9	123	$\overline{\text{SMIACT/}}$ $\overline{\text{SMIADS}}$	162
$\overline{\text{ADSC}}$	140	$\overline{\text{CRD0}}$	166	CY29	191	PAD10	122	$\overline{\text{STOP}}$	88
ADVNP	143	$\overline{\text{CRD1}}$	165	CY30	190	PAD11	121	$\overline{\text{TRDY}}$	92
BE0	155	$\overline{\text{CWE0}}$	175	CY31	189	PAD12	120	W/ $\overline{\text{R/INV}}$	161
BE1	154	$\overline{\text{CWE1}}$	174	D/ $\overline{\text{C}}$	144	PAD13	119	+3.3V	32,169,208
BE2	153	$\overline{\text{CWE2}}$	173	$\overline{\text{DEVSEL}}$	90	PAD14	118	+5V	20,65,76,91, 124,136
BE3	152	$\overline{\text{CWE3}}$	172	$\overline{\text{DWE}}$	72	PAD15	117		
BE4	150	$\overline{\text{CWE4}}$	171	$\overline{\text{EADS}}$	158	PAD16	116		
BE5	149	$\overline{\text{CWE5}}$	170	ECPUCLK	139	PAD17	115		
BE6	148	$\overline{\text{CWE6}}$	168	$\overline{\text{FRAME}}$	95	PAD18	114		



82C691-
101

Figure 1. CY82C691 Functional Block Diagram

Introduction

System Overview

The CY82C691/692/693 hyperCache™ Chipset provides all the functions necessary to implement a 3.3V Pentium-class processor based system with the PCI (Peripheral Component Interconnect) bus and the ISA (Industry Standard Architecture) bus. The chipset consists of the CY82C691 System Controller, the CY82C692 Data Path/Cache chip, and the CY82C693 Peripheral Controller. System designers can exploit the advantages of the PCI bus while maintaining access to the large base of ISA cards in the marketplace.

The Cypress hyperCache Chipset offers system designers several key advantages. With only three chips, CY82C691/692/693, a complete system with a 128-KB, two-way set associative, synchronous, pipelined L2 cache can be implemented. The cache size may be increased up to 1 MB with additional CY82C694 16Kx64 SRAMs or other synchronous or pipelined burst SRAMs. 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.

CY82C691 Introduction

The CY82C691 System Controller provides the interfaces to the CPU, PCI, and DRAM buses. It also integrates the memory controller, cache controller, cache tag, and the CPU bus controller. *Figure 1* shows a block diagram of the CY82C691.

Functional Overview

The CY82C691 System Controller is a highly integrated device. It provides control for the CPU, cache, memory, and PCI. The memory controller supports up to 768 MB of main memory with standard page-mode DRAMs or EDO DRAMs. The cacheable range can be configured to cover the entire memory space. Support is provided for up to 6 banks of 72-bit wide DRAM SIMMs (parity checking/generation is provided by the CY82C692). Asymmetrical DRAM banks are also supported. 24 mA outputs with programmable drive are provided on the DRAM lines, thus eliminating the need for external buffers.

The cache controller supports a look-aside (parallel) cache with synchronous or pipelined burst SRAMs. Asynchronous SRAMs are not supported. Burst read and write timing at 66 MHz provides for 3-1-1-1 operation. The CY82C691 integrates an 8Kx21 tag-RAM to further reduce system cost. The tag can be configured to be either direct mapped or two-way set associative. Cache sizes can range from 128 KB to 1 MB in 128-KB increments. Support is provided for asymmetrical SRAM banks. For example, a 384-KB cache can be configured with the 128-KB cache in the CY82C692 and a 256-KB external expansion cache. For cache bank sizes greater than or equal to 512 KB, the cache is sectored with two lines per sector. Cache coherency with main memory is maintained at all times.

Bus concurrency is supported between the CPU, cache, DRAM, and PCI bus with the use of post-write and pre-read FIFOs. Pentium pipelined addressing and power management features (SMM) are supported. The CY82C691 also supports the Cyrix M1 processor and the AMD K5 processor. External bus frequencies up to 66 MHz are supported. The CY82C691 also

generates all the control for the CY82C692 Data Path/Cache chip.

PCI Bus

The purpose of this section is to give an overview of PCI, the motivation behind it, and its features. Basic transfers and rules are discussed. For a detailed description of the PCI bus, all of the rules and requirements, see PCI Specification, Revision 2.1.

The PCI Bus was defined in order to satisfy the growing need for a standardized high-speed local bus that is independent of the processors, operating system, and CPU bus speed. New generations of computers incorporating I/O intensive software will require bandwidth that cannot be satisfied with the traditional I/O architectures. The PCI specification 2.1 addresses these requirements and provides an upgrade path for future requirements. Some of the PCI features include:

- Processor Independent
- Multiplexed, Burst Mode Operation
- 120 MBytes/sec usable throughput (32-bit data path)
- Three physical address spaces
 - Memory
 - I/O
 - Configuration
- Hidden Arbitration

PCI is defined as a synchronous bus that can operate from 0 to 33 MHz. All transfers take place on the rising edge of the clock (PCICLK). The basic data transfer in PCI is a burst. A burst transfer consists of an address phase, followed by one or more data phases. The address phase is defined as the first rising edge of the clock where $\overline{\text{FRAME}}$ is asserted (LOW). During the Address phase, the Master (also referred to as the initiator) asserts the appropriate address on the address/data lines (AD[31:0]) while also asserting the appropriate command on the Command/Byte Enable $\overline{\text{C/BE}}[3:0]$ lines. With the information transferred during the address phase, all PCI devices, including the slave (or Target), can determine: (1) whether the transaction falls within its designated address range, (2) the kind of transfer that will take place (e.g. a read or write to memory, I/O, or configuration space), and (3) how to respond to that particular command.

Once a device recognizes that it is the target for the transaction, it claims the transactions by asserting Device Select ($\overline{\text{DEVSEL}}$) LOW. $\overline{\text{DEVSEL}}$ must be asserted LOW in order for any information to be transferred.

The address phase is followed by one or more data phases. Whether the initial data phase occurs on the subsequent clock edge is determined by the type of transaction and the ability of either agent to provide/accept the data within the appropriate time period. Since the address and data lines are multiplexed, a normal read operation requires a "turn-around" cycle to avoid bus contention. During this cycle, control of the Address/Data lines is transferred from the master to the slave, who must now use these lines to drive out the requested information. During a write operation, this "turn-around" cycle is not required since the master is providing the write data and does not have to relinquish control of the bus. PCI also allows that both the master and slave have the ability to insert wait states should either require additional cycles in order to properly participate in the transfer. This is accomplished with the Initiator Ready signal ($\overline{\text{IRDY}}$) and the Target Ready signal ($\overline{\text{TRDY}}$) for the Initiator

and Target, respectively. Either of these signals being de-asserted (HIGH) during the data phase of the transaction will insert a wait state, thereby preventing data from being transferred during that cycle. The data presented on the AD[31:0] lines is transferred during a data phase on the rising edge of the clock when ALL of the following signals are active (LOW): $\overline{\text{DEVSEL}}$, $\overline{\text{TRDY}}$, $\overline{\text{IRDY}}$, and $\overline{\text{FRAME}}$ (except during the final data phase, when $\overline{\text{FRAME}}$ is HIGH, which is explained later). The bytes containing meaningful information are controlled by the C/BE[3:0] signals. During the data phase, these signals behave as byte enable lines.

Transactions are normally terminated by the Master who deasserts $\overline{\text{FRAME}}$ HIGH on the clock prior to the last data phase. By doing so, all of the agents on the bus, including the Target and the Arbiter, recognize that the current transaction is coming to an end. This advanced notice allows the Arbiter to grant ownership of the bus to the next requesting agent. This is referred to as Hidden Arbitration since no additional clock cycles are consumed. The new Master will not start to drive the bus until the current transaction is actually completed. The Target has the ability to abort the transaction prematurely should the need arise, although this is not the typical method of termination.

Arbitration in PCI is access based instead of time slot based. This is accomplished through a simple request-grant handshaking scheme through a central arbiter. Each agent has dedicated request and grant lines to the arbiter. A bus Master must request and be granted bus ownership each time a transaction is desired.

PCI defines three different physical address spaces; memory, I/O, and configuration space. Each of these address spaces has its own characteristics. Therefore, transactions to each space are handled differently. The memory and I/O address spaces are customary, but the configuration address space has been defined by PCI in order to support hardware configuration.

The CY82C691 functions as a high-performance PCI host bridge front the processor to the PCI bus. During PCI to main memory cycles the CY82C691 acts as a target on the PCI bus, allowing masters to read and write to main memory. For CPU cycles the CY82C691 acts as a PCI master. The CY82C691 supports burst and single transfers, and will burst whenever possible. The CY82C691 is also parked on the PCI bus (default owner) so the CPU will have use of the bus without incurring arbitration overhead.

The CY82C691 also has a local arbiter which controls the ownership of the local memory bus. Arbitration is performed between the CPU, PCI, shared DRAM peripheral, and DRAM refresh. While one device owns the memory bus others may continue to operate without being held. See the section on concurrent bus operation.

A more detailed discussion is provided in the CY82C693 data sheet. See also PCI Spec Rev 2.1.

Address Space

The purpose of this section is to explain the memory and I/O address space mapping used by the CY82C691.

The CY82C691 recognizes two different physical address spaces: memory and I/O. Transactions to these two address spaces are handled differently, and therefore need to be distinguished from one another. The CY82C691 differentiates these two different type of transactions by monitoring the M/ $\overline{\text{IO}}$ signal coming from the CPU bus, or the C/BE signals coming from the PCI bus. For CPU initiated I/O cycles, the CY82C691 decodes for accesses to

its configuration registers. If the access is not to one of its registers, then the transaction is passed onto the PCI bus, including interrupt acknowledge cycles. For CPU-initiated memory cycles, the CY82C691 decodes the address based on its memory configuration registers. If the CPU address falls inside the memory range programmed in the registers, then the cycle is forwarded to memory, otherwise it is passed on to the PCI bus. As a PCI target, the CY82C691 only responds to memory cycles. CPU writes to PCI memory spaces are posted (stored in an on-chip FIFO) to improve throughput. CPU writes to PCI I/O space are not posted in the on-chip FIFO.

I/O Address Space

Although the I/O address space can extend the full 32 bits (a possible 4 GB of I/O address space), I/O transactions are limited to the lower 64 KB (0000h-FFFFh). The remaining address locations are not valid I/O address space. In addition, the lower 1 KB (0000h-03FFh) of I/O space has been assigned as AT I/O space. The CY82C691 only decodes for I/O accesses to its on-chip configuration registers, all other I/O cycles are forwarded to PCI where the decoding is handled by the CY82C693. The CY82C693 is the subtractive decode agent on the PCI bus and it directs all unclaimed transactions to the ISA bus. See the CY82C693 data sheet for more details.

Memory Address Space

The CY82C691 supports up to 768 MB of local memory space. The full 4-GB memory space can be mapped over the local memory or PCI spaces. This mapping allows the CY82C691 to determine the destination of the transaction and respond appropriately. Memory address ranges are programmed in the CY82C691 at system startup. Address ranges can be assigned to either PCI space or main memory, cacheable or non-cacheable. All transactions not destined to main memory are forwarded to the PCI bus.

PCI Configuration Space

The CY82C691 supports the preferred PCI Configuration Mechanism #1 that allows PCI configuration cycles to be generated by software. Both Type 0 and Type 1 configurations accesses are also supported. All required fields within the Configuration Header Space are also supported.

L2 Cache Controller

The purpose of this section is to describe the basic operation of the CY82C691 L2 cache controller. The CY82C691 integrates a high performance WB/WT cache controller providing an integrated tag SRAM and a full first and second level cache coherency mechanism. Cache sizes up to 1 MB are supported through the use of the 128 KB of cache in the CY82C692 and external synchronous or pipelined burst SRAMs. The tag is two-way set associative or direct mapped and can be configured to support either WB (optional dirty bit in tag) or WT write policy. Write allocation is not supported.

The tag SRAM inside the CY82C691 is an 8Kx21 RAM organized as two 8Kx10 RAMs, TAGA and TAGB, and an 8Kx1 RAM for the LRU bit. If write-back operation is selected, one of the tag bits can optionally become the modified bit. If no modified bit is used, all cache lines are assumed dirty.

The CY82C692 integrates 128 KB of BSRAM on-chip, provides data buffering, and parity generation and checking. More information on the data BSRAMs in the CY82C692 is provided in the CY82C692 datasheet. *Table 1* shows the supported cache sizes, the tag data, the tag address, and the cacheable range.

Table 1. Cache Tag Configurations (no dirty bit)

Cache Size per bank (1 or 2 banks are allowed)	Tag Address	Tag Data	Maximum Cacheable Range
128 KB	A16-A5	A26-A17	128 MB
256 KB	A17-A5	A27-A18	256 MB
512 KB	A17-A5	A28-A19	512 MB

The cache controller supports 1 or 2 banks of SRAM, with a maximum size of 512 KB per bank. The controller also supports the mixing of pipelined and non-pipelined SRAMs to provide optimum flexibility for the system designer.

The CY82C691 provides burst read/write performance of 3-1-1-1 for direct mapped or two-way set associative two-bank configurations. For two-way set associative one-bank configurations, the CY82C691 provides burst read/write timing of 3/4-1-1-1, where the 3 is for a bank hit and the 4 is for a bank miss. The timing of the L2 controller is programmable to allow the system designer greater flexibility in the design. Wait states can be added for all cache configurations.

Cache Configurations

Below is a list of the supported cache configurations with the associated cache sizes.

Direct mapped (non-sectored) – 128K, 256K, 512K Bytes.

Direct mapped (sectored) – 1M Bytes.

Two-way set (non-sectored, 1 bank) – 128K, 256K, 512K Bytes.

Two-way set (sectored, 1 bank) – 1M Bytes.

Two-way set (non-sectored, 2 banks) – 256K, 384K, 512K Bytes.

Two-way set (sectored, 2 banks) – 640K, 768K, 896K, 1M Bytes.

The cache configurations above can be implemented with the use of CY82C692s with CY82C694s or external BSRAMs (pipelined or synchronous).

L2 Cache Operation

Before discussing the detailed operation of the of the CY82C691 cache subsystem, it is important to define some key terms associated with caches.

Cache hit/miss cycle: These cycles occur on all CPU memory references. The CPU initiates a memory cycle by asserting the address along with \overline{ADS} in T1. The CY82C691 takes the high order address bits and compares them to the stored tag data in the location pointed to by the low order address bits. For direct-mapped configurations, one comparison is done and in 2-way set configurations, two comparisons are done simultaneously. When a match is detected and the referenced memory location is cacheable ($\overline{CACHE\ LOW}$ from the CPU and $\overline{KEN\ LOW}$ from the memory controller) a cache hit cycle takes place and data is returned from the data SRAMs. If there is no match found by the comparator, then the cycle is a cache miss and data is returned from main memory. An access to non-cacheable memory is neither a hit or a miss. The cycle is just forwarded to the memory controller and the tag is not updated.

Valid bit: A mechanism for determining if the entries in the tag contain valid data. For all cache sizes except 1-MB, there is no valid bit in the CY82C691. All lines are assumed and kept valid at all times. At power up, BIOS software is responsible for initializing the cache with valid data. For 1 MB cache sizes, there

are two valid bits in each entry in the tag because each tag entry controls two cache lines.

Dirty bit: A mechanism for monitoring coherency between the cache and system memory when using the write-back policy. This feature is programmable in the CY82C691. If not used, all lines are assumed dirty. Not using a dirty bit increases the cacheable memory range.

Linefill: This type of cycle occurs on a CPU read miss to a cacheable address. It involves reading a new line from system memory and storing it in the cache.

Castout: This type of cycle occurs on a CPU read miss to a cacheable address if the line that will be replaced is dirty. In this case the dirty line is stored back to memory prior to the linefill taking place.

CPU Write Cycle

If the CY82C691 is in write-through mode and there is a cache hit, both the cache and the main memory are updated. If there is a cache miss, only main memory is updated. A linefill is not performed (no write allocate).

CPU Read Cycle

If there is a cache hit, the data is transferred from the cache to the CPU. Main memory is not accessed.

If there is a cache miss, the line containing the requested data is transferred from main memory to the cache and the corresponding data is returned to the CPU. In the case of write-back, if the line is dirty, a castout is performed prior to the linefill.

Cache Coherency

The snoop mechanism in the CY82C691 ensures data coherency between the L1 cache, L2 cache, and main memory. For write-back caches, the term “inquiry” is often used to describe the snooping operation.

The CY82C691 monitors all master accesses to main memory and the caches. When an external master (PCI) issues an access to the main memory, the CY82C691 will generate an inquiry cycle to the CPU by driving \overline{AHOLD} , putting memory address on the CPU bus, and asserting \overline{EADS} . The CY82C691 has an inquiry filter to reduce the number of unnecessary inquiries. If a new address was snooped on the previous transaction, then no inquiry cycle will be generated. If the CPU asserts \overline{HITM} , in response to the inquiry, the CY82C691 will let the CPU issue the write-back cycle and hold off the master until the CPU cycle completes. Then the master is allowed to proceed with the transfer.

To maintain coherency in the L2 cache, the CY82C691 also snoops the L2. If a master memory read hit occurs, data will be supplied from the SRAMs. On a memory read miss, data is supplied from the DRAMs. In the case of a memory write hit, data will be written to the SRAMs and the DRAMs. A write miss cycle will only write data to the DRAMs. Master cycles are covered in detail in the next section.

PCI Master Read from Memory

A PCI master wins ownership of the PCI bus and initiates a read cycle. The CY82C691 detects \overline{FRAME} asserted and decodes the address. If the access is targeted at main memory, the CY82C691 will drive \overline{DEVSEL} active. The CY82C691 begins the memory read cycle and snoops the L1 and the L2. Several situations may occur; L1 clean hit/L2 hit, L1 dirty hit/L2 hit, L1 clean hit/L2 miss, L1 dirty hit/L2 miss, L1 miss/L2 hit, and L1 miss/L2 miss. There’s no need to differentiate between L2 dirty or clean. The

data in the L2, dirty or clean, will be returned to the master on an L2 hit. The L2 data is the most current.

L1 clean hit/L2 hit: Data is returned to the master from the L2 cache. The CPU is backed off during this procedure.

L1 dirty hit/L2 hit: The master is held, the CPU is allowed to perform the castout of the dirty line, the line is written to the L2 cache and the appropriate data is returned to the master. The CPU marks the line invalid.

L1 clean hit/L2 miss: Data is returned from the DRAMs to the master.

L1 dirty hit/L2 miss: The CPU is allowed to castout the modified line and invalidate it. The data is written to the memory and returned to the master.

L1 miss/L2 hit: Data is supplied to the master from the L2 cache. The CPU is backed off.

When both snoops are misses, the memory cycle is allowed to proceed and the data is returned to the master from main memory.

PCI Master Write to Memory

A master wins ownership of the PCI bus and initiates a write cycle. The CY82C691 detects $\overline{\text{FRAME}}$ asserted, decodes the address, and drives $\overline{\text{DEVSEL}}$ active. The CY82C691 begins the memory write cycle and snoops the L1 and L2. The same snoop results can occur as in the memory read case.

L1 clean hit/L2 hit: The line in the CPU is invalidated and the master data is written to the L2 cache and main memory.

L1 dirty hit/L2 hit: The master is held, the CPU performs the castout of the line, the line is written to the L2 where it is marked dirty, and the CPU invalidates the line in the L1. The master data is then written to the L2 cache and main memory.

L1 clean hit/L2 miss: The CPU invalidates the line and the data is written to main memory.

L1 dirty hit/L2 miss: The CPU performs the castout and invalidates the line in the L1. The CPU data is written to main memory followed by the master data.

L1 miss/L2 hit: The master data is written to both the L2 cache and to main memory. If the line in the L2 was clean, it is marked dirty. If it was dirty, it remains dirty.

If both snoops miss, the memory cycle is allowed to proceed. Data is written to main memory.

DRAM Memory Controller

The DRAM controller in the CY82C691 provides the control of the memory subsystem by driving the memory address, $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, and $\overline{\text{DWE}}$. It interfaces main memory to the CPU bus and PCI bus. The CY82C691 provides all the DRAM control signals and the lower 32-bit data path to the DRAMs (the upper 32-bit data path is supported by the CY82C692). The data path is also used to pass data from/to PCI to/from the CPU or DRAM. For more information on the data paths see the CY82C692 data-sheet. Up to 12 single-sided or 6 double-sided 72-pin SIMMs with a maximum memory size of 768 MB are supported. The DRAM controller is extremely flexible and is fully configurable through registers to provide optimum configuration for the given application.

The CY82C691 controls six banks of DRAM, each bank with a max size of 16M x 64 bits yielding 768 MB total memory. The memory controller supports standard page-mode DRAMs as well as EDO DRAMs. Different speeds are supported through programming of the DRAM controller configuration registers.

EDO DRAMs provide better performance over page-mode DRAMs. DRAM sizes up to 16 Mbit are supported by providing 12 memory address lines. In addition, page size selection from 6 to 12 address bits is available independent of DRAM size. The memory address is driven with 24 mA buffers to eliminate the need for external buffering. The drive capacity of the buffers is programmable. Parity support is provided in the CY82C691. The CY82C692 does all the parity generation and checking but is enabled or disabled via the CY82C691.

The CY82C691 memory controller supports asymmetrical DRAM banks. This means that each bank may be a different type (EDO or page-mode), a different variety (different page size DRAMs), different speed, or different density. The only restriction is that all DRAMs within a given bank are the same. Support for automatic EDO detection is also provided.

DRAM Performance

The CY82C691 DRAM performance can be controlled through the use of the configuration registers. Various DRAM timing parameters may be changed to adjust memory performance. Programmable timings allow the use of different DRAM types. There is also support for EDO DRAMs to further increase performance. The signals for EDO are the same for that of standard DRAM. EDO output disabling is controlled using the $\overline{\text{DWE}}$ signal.

Table 2 below shows the optimum DRAM timing. The numbers are based on 60-ns DRAMs with the system running at 66 MHz.

Table 2. DRAM Performance

Cycle Type	Burst Timing
DRAM Read page hit/row miss/page miss	5/9/12-3-3-3 Page-Mode 5/9/12-2-2-2 EDO
DRAM Write page hit/row miss/ page miss	4/8/11-2-2-2
Posted Write (write to CY82C692 FIFO)	3-1-1-1

Shadow RAM

The CY82C691 provides shadow RAM support to speed up accesses to system ROM. ROM code is copied to a reserved RAM space and write protected. All subsequent accesses to ROM will be routed to DRAM, thus improving performance.

Refresh

Refresh of the DRAM array is performed using the CAS before RAS refresh mechanism. The timing of refresh cycles is derived from the system clock and is totally independent of expansion bus refresh cycles. The CPU is not held during refresh cycles. Refresh cycles will be deferred until the DRAM interface is idle provided the DRAM refresh requirements are not violated.

Shared DRAM Bus Support

The CY82C691 will provide support for sharing of the DRAM bus with external peripherals such as graphics adapters or digital video chips. The chipset implications involve DRAM arbitration and DRAM protection. The CY82C691 provides a DRAM request signal and a DRAM grant signal for arbitration. When a DRAM peripheral wins control of the bus, the CY82C691 floats all of the DRAM control signals and allows the peripheral to access main memory directly. This also allows the DRAM timing for the peripheral to be independent of the CY82C691 DRAM

timing. The CY82C691 provides registers to control how the peripheral DRAM space is used. The CY82C691 can be programmed to treat the peripheral memory space as reserved.

Synchronous/Asynchronous PCI Operation

The CY82C691 allows for both synchronous and asynchronous PCI bus operation. Synchronous PCI bus operation is defined as when the PCI and CPU buses are operating from the same clock. Asynchronous PCI operation is defined as when the PCI and CPU clocks are either out of phase or running at different frequencies.

SMM Mode Support

The Cypress Pentium chipset provides for extensive power management, through both hardware and software. Most of the power management functions are handled by the CY82C693. For more information please refer to the CY82C693 datasheet.

The CY82C691 provides control over SMM memory space. When SMI is issued to the processor by the CY82C693, the CPU enters SMM mode where it runs out of a restricted memory space. The CY82C691 provides support by providing SMM address mapping and write protection.

Pentium Class CPU Support

The CY82C691 supports Pentium-class CPUs from multiple vendors. The CY82C691 supports the Intel Pentium ICOMP 815/100 and Pentium ICOMP 735/90. The CY82C691 also supports the Cyrix M1 processor and the AMD K5 processors.

Concurrent Bus Support

The CY82C691 supports full bus concurrency between the CPU and the PCI bus. This is made possible through the use of the L2 cache and the extensive buffering provided by the CY82C692.

The CPU can run out of the L2 cache while a PCI master has access to main memory. The CPU can go to main memory while a PCI master is going to another PCI device. The only times where a conflict will occur are when both a PCI master and the CPU want to go to main memory or when PCI and the CPU require access to the CPU bus (for an inquiry cycle). In this case, some level of concurrent bus operation may still be possible through the use of the FIFOs in the CY82C691 and CY82C692. The CY82C691 contains 8-level-deep FIFOs for post-writing and pre-reading data from PCI. The CY82C692 contains additional FIFOs. See the CY82C692 datasheet for more details.

When it is not possible to maintain concurrent bus operation, the CY82C691 arbitrates for the use of the memory bus between the CPU and PCI. If the PCI device wins, the CPU is held off and vice versa. The chipset can also be run in non-concurrent mode. The CPU is held off the bus while external master cycles take place and the PCI masters are held off while CPU cycles take place.

CY82C691 Signal Description

The CY82C691 signals are divided into four functional areas: PCI Interface signals, CPU Interface signals, DRAM interface signals, and Miscellaneous signals.

PCI Interface

Name	I/O	Description
PAD[31:0]	I/O	PCI Address/Data Bus: Multiplexed bidirectional address/data lines on the PCI bus. The CY82C691 either drives or samples these lines during PCI cycles.
PCICLK	I	PCI Clock: PCI Clock Input. Used to support asynchronous CPU/PCI operation. All PCI transactions are referenced to the rising edge of PCICLK.
PC/BE[3:0]	I/O	PCI Command & Byte Enables: PC/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.
FRAME	I/O	Cycle Frame: Driven by the current bus master to indicate the start and duration of a transaction.
IRDY	I/O	Initiator Ready: The assertion of IRDY indicates the current bus master's ability to complete the current data phase of the transaction. Used in conjunction with TRDY from the target.
TRDY	I/O	Target Ready: The assertion of TRDY indicates the current target's ability to complete the current data phase of the transaction. Works in conjunction with IRDY from the master.
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.
PAR	I/O	Parity: An even parity bit across PAD[31:0] and PC/BE[3:0]. As a master the CY82C691 generates even parity on PCI write cycles. On read cycles the CY82C691 checks parity by sampling PAR.
STOP	I/O	Stop: Indicates that the current target is requesting the master to stop the current transaction. STOP is used in conjunction with DEVSEL and TRDY to indicate a disconnect, target abort, and retry cycles.
PLOCK	I	PCI Lock: Used to indicate that an atomic operation is taking place and may require multiple cycles to complete without another master interfering.
PERR/DGNT	I	Parity Error/ DRAM Bus Grant: Parity Error may be asserted by any agent that detects a parity error during the data phase of a transaction. Address parity errors are reported on SERR. Also, the DRAM bus grant signal for shared DRAM bus operation.
SERR/DREQ	I/O	System Error/ DRAM Bus Request: System error may be asserted by any agent for reporting address parity errors or any other types of errors beside data parity. Also, the DRAM bus request signal for shared DRAM bus operation.
C691BSY	O	C691 Busy: The CY82C691 asserts busy to indicate to the central arbiter that the CY82C691 has ownership of the PCI bus. When not asserted the PCI bus is free to be granted to other masters.
C691GNT	I	C691Grant: When asserted, it indicates to the CY82C691 that it has been granted use of the PCI bus and is allowed to initiate a transaction.

CPU Interface

Name	I/O	Description
A[31:5]	I/O	CPU Address Bus: A[31:5] are connected to the CPU A[31:5] lines. They are inputs to the CY82C691 during CPU initiated cycles and are outputs during cache inquiry cycles.
A[4:3]	I	CPU Address 4 and 3: Same as above except they are not driven during inquire cycles.
BE[7:0]	I	CPU Byte enables: The byte enables indicate which byte lanes on the CPU data bus carry valid data. They also define the type of special cycle when $M/\overline{IO} = D/\overline{C} = 0$.
\overline{ADS}	I	CPU Address Strobe: Used to indicate the start of a new bus cycle. Driven in the same clock as the address, byte enables, and cycle definition signals.
M/\overline{IO}	I	CPU Memory/IO: Driven by the CPU during the T1 to indicate a memory or I/O space access. Along with D/C and W/R make up the cycle definition signals.
D/\overline{C}	I	CPU Data/Code: Used by the CPU to differentiate accesses for data and instructions. Used as a cycle definition signal.
$W/\overline{R}/INV$	I/O	CPU Write/Read: Used by the CPU to define write and read cycles. Along with M/\overline{IO} and D/\overline{C} makes up the cycle definition signals. Also used as the invalidate signal when running in Level 1 write-back mode.
\overline{CACHE}	I	Cacheability: The Pentium asserts \overline{CACHE} to indicate the internal cacheability of a read cycle or that a write cycle is a burst write-back. It is driven along with the cycle definition signals.
\overline{BRDY}	O	Burst Ready: \overline{BRDY} indicates to the Pentium that the data is available in the current clock cycle.
\overline{KEN}	O	Cache Enable: Driven by the CY82C691 to tell the processor that the current memory cycle is cacheable. \overline{KEN} is asserted for all memory accesses that are to cacheable memory as determined by the address registers in the CY82C691. These registers are fully programmable.
\overline{NA}	O	Next Address: Used to support CPU address pipelining. The CY82C691 asserts \overline{NA} for one clock when it is ready to accept a new address from the processor, even if the current transaction hasn't completed. The CPU may drive \overline{ADS} two clocks after \overline{NA} is asserted.
\overline{EADS}	O	External \overline{ADS} : The CY82C691 drives \overline{EADS} to indicate to the processor that a valid snoop address has been placed on the address bus.
\overline{HITM}	I	Hit Modified: The CPU asserts \overline{HITM} to inform the CY82C691 that the inquiry cycle hit a modified line in the L1 cache. It is asserted two clocks after \overline{EADS} if the hit line was in the modified state.
\overline{BOFF}	O	Backoff: The CY82C691 asserts \overline{BOFF} to force the CPU to float the bus in the next clock cycle. The bus is floated until backoff is sampled deasserted. Outstanding bus cycles are restarted.
HOLD/AHOLD	O	Bus Hold Request/Address Hold: The CY82C691 asserts HOLD to tell the CPU that it wants the bus. The CPU will complete all outstanding bus cycles, float its bus and drive HOLDA. AHOLD tells the CPU to float its address bus. HOLD will be used for non-concurrent mode and AHOLD will be used for concurrent mode operation.
HLDA/ \overline{LOCK}	I/O	Hold Acknowledge/ Bus Lock: The CPU asserts HLDA in response to HOLD being asserted by the CY82C691. Prior to HLDA being asserted, the CPU completes outstanding transactions and floats the CPU bus. CPU \overline{LOCK} is used by the CPU for atomic operations. HLDA is used in non-concurrent mode and \overline{LOCK} is used for concurrent mode operation.
$\overline{SMIACT}/\overline{SMADS}$	I	System Management Interrupt Active: Output from the processor that tells the CY82C691 that the processor is operating in SMM (System Management Mode).
CPUCLK	I	CPU Clock: External Clock. All CY82C691 timing is based off CPU clock.
EPCUCLK	I	Early CPU Clock: An early version of CPU clock.
RESET	I	System Reset: The CY82C691 uses this signal to initialize the chip to a known state after system power-up.

Cache Interface Signals

Name	I/O	Description
CAC/ADVNP	O	Cache Sectored Address Control/Set Selector/Non-pipelined SRAM Advance Signal: Controls the cache address line for sectoring and z-way set, single-bank operation. Chooses whether set0 or set1 will be accessed. Also, the \overline{ADV} input for flow-through BSRAMs.
ADVP	O	Advance Address: Tells the synchronous, pipelined BSRAMs to increment their internal address counters for the next clock cycle.
\overline{ADSC}	O	Cache ADS: CPU \overline{ADS} delayed by one clock cycle for CPU initiated transactions. Also, driven by the CY82C691 when doing line replacements and castouts, or returning data to an external master.
$\overline{CEI}[A:B]$	O	Burst SRAM Cache Chip Select: Used with the CY82C692 and external BSRAMs for selecting the RAM. Also serves as the cache set selection for two-way set associative L2 organizations during writes.
$\overline{CRD}[1:0]$	O	Cache SRAM Output Enable: These signals are asserted by the CY82C691 when data is to be read from the L2 cache. Functions as the read set selection for two way set organizations.
$\overline{CWE}[7:0]$	O	Cache SRAM Write Enable: $\overline{CWE}[7:0]$ are asserted by the CY82C691 to write data to the L2 cache BSRAMs or CY82C692 on a byte-by-byte basis.

Memory Interface

Name	I/O	Description
MA[11:0]	O	DRAM Multiplexed Address: Provide the row and column address to the DRAM array. Up to 24 mA drive (programmable) to eliminate external buffering.
$\overline{RAS}[5:0]$	O	Row Address Strobe: Used by the DRAMs to latch the row address on the DRAM address lines. Each RAS line corresponds to one bank of DRAM.
$\overline{CAS}[7:0]$	O	Column Address Strobe: Used to latch the column address on the DRAM address lines. Each CAS signal corresponds to one byte of data in the DRAM array.
\overline{DWE}	O	DRAM Write Enable: The signal used to initiate a write into the DRAM array.

CY82C692 Interface Signals

Name	I/O	Description
CNT[11:0]	O	692 Control: These twelve signals are used to control the CY82C692 BSRAMs and FIFOs. The CY82C691 keeps complete track of all CY82C692 transitions.
CY[31:0]	O	CY Bus: This 32-bit bus functions as the data path between the CY82C691 and the CY82C692 for PCI transfers. It also functions as the DRAM lower (bits 31 through 0) data bus.



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 $+70^{\circ}\text{C}$

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 ($T_A=0^{\circ}\text{C}$ 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_{OH3}	Output HIGH Voltage (3.3V Outputs)	2.4	$V_{OUT}+0.3$	V
V_{OH5}	Output HIGH Voltage (5V Outputs)	2.4	$V_{CC}+0.5$	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

Document #: 38-00456