LS105
SATC Controller

Features

- 33 MHz 32-bit PCI bus interface, 8 interrupt lines to support up to 8 LS100s.
- 32/48 bit Interface with standard asynchronous SRAM to cache up to 64K MAC addresses.
- Supports Port based VLAN with internal port map registers. Optional Multicast type field usage bits can be implemented using one additional x16 SRAM.
- Selectable auto-learning in hardware for address learning without CPU intervention.
- Selectable auto-aging in hardware to age-out old/unused MAC address in LS105.
- Provides MDC/MDIO interface.
- Provides µAccess interface.
- JTAG compliant.
- 0.5 micron, 3.3V CMOS technology.
- 208 pin PQFP package.

Description

The LS105 is a Secondary Address Translation Cache (SATC) controller intended for the I-Cube LS Fast Ethernet SwitchSet™. By providing up to 64K cache entries it increases the cache hit ratio for backbone switch applications. The LS105 is a PCI bus master and requires no glue logic to interface to the rest of the LS chipset.

The LS105 integrates a 32 bit PCI bus master interface, hash generator, search engine, address learning and hardware aging in a 208PQFP package. It can be used in auto (hardware) address learning mode or in CPU directed address learning mode. The LS105 provides a µAccess interface to perform PCI bus bridging functions if desired.

A complete secondary ATC subsystem can be constructed using the LS105 and a minimum of two external asynchronous SRAMs. 32Kx8, 32Kx16, 64Kx16, and 256Kx16 SRAMs are supported.

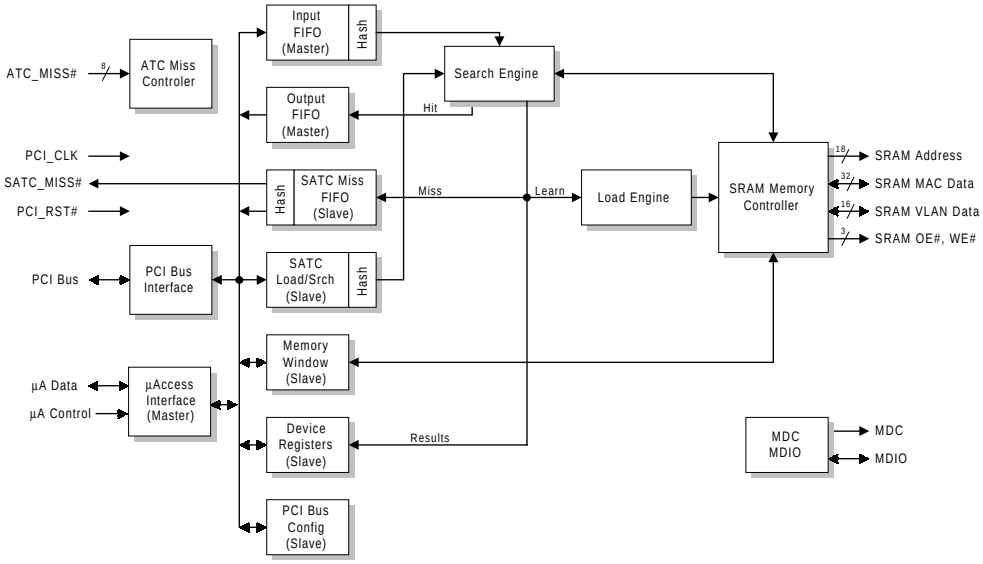


Figure 1: LS105 Functional Block Diagram

Introduction

The LS SwitchSet™ uses a hierarchical cache architecture for port mapping. The primary cache is in the LS100. It stores port mappings for the 64 most recently used MAC addresses. The system is capable of learning 64K MAC addresses; these can be stored in main memory (on a CPU card) or in the LS105's SRAM. Address maps are transferred to and from the CPU, the LS105 secondary cache, and the LS100 quad port controller via the PCI Bus. In a backbone switch application where a large number of users are expected to use the switch, the performance of address lookup and port mapping can be improved by using the second level cache in the LS105 cache system. This allows faster port mapping look up and reduces flooding.

Figure 2 shows how the LS105, the LS101 twenty five port LAN switching element, and the LS100 quad port controller can be combined to create a scaleable workgroup switching hub. Variations of this architecture can be used for backbone and stackable switch applications.

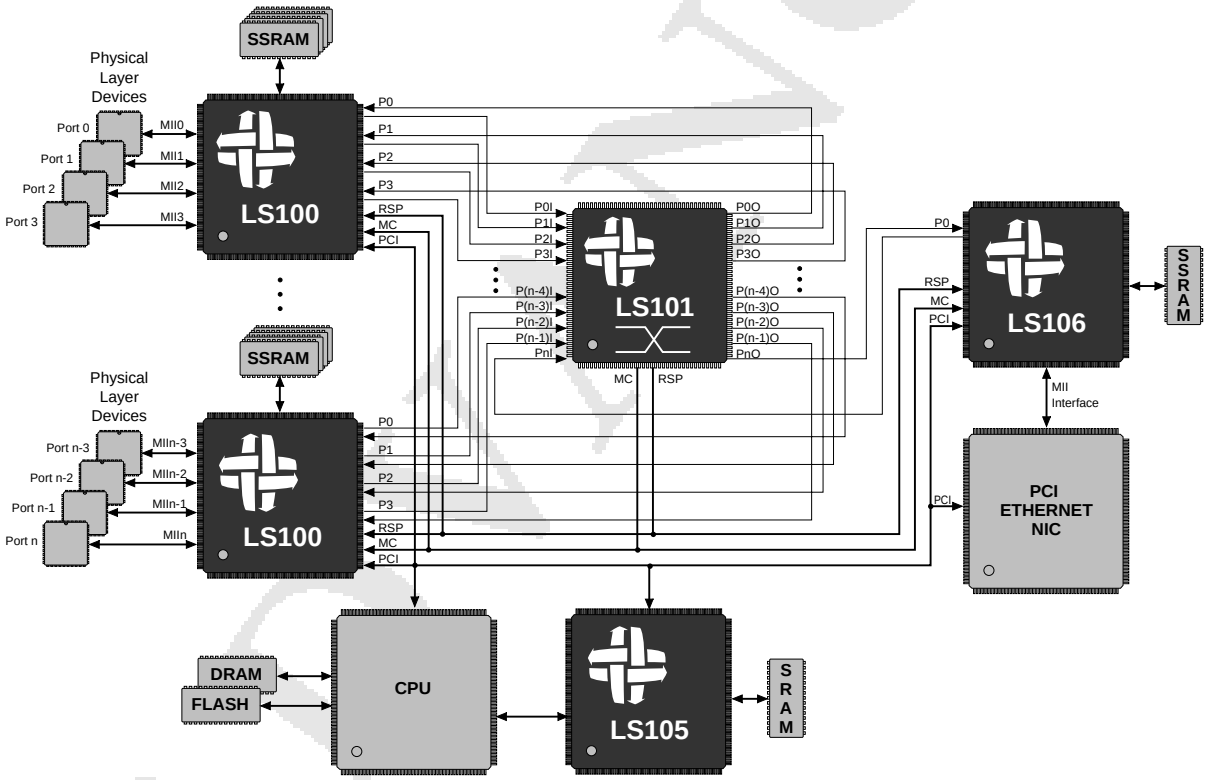


Figure 2: The LS Family System Block Diagram

Packets are transferred between input and output ports by connections established through the switch. These connections are dynamic and are constructed on a demand basis. Requests for connections through the switch are generated by the quad port ethernet switch devices (LS100s). These requests are sent to the LS101 via the same port interfaces used to transfer the packet data. This process is referred to as in-band signaling. The 4B5B coding scheme used by the LS101 lets it discriminate between packet data and signaling data.

The port datapath between the LS100s and LS101 are two bits wide. Each general purpose port consists of separate input and output pins, allowing full duplex operation. Each port is identified by a unique port code.

The secondary address translation cache improves the performance of the LS based switch by caching MAC addresses and port mappings. The LS100 quad MAC contains a sixty-four entry ATC. In a system where a large number of simultaneous users are using the switch, there is a possibility that a 64 entry ATC is not enough and additional port mappings will have to be fetched by CPU from the main memory. By incorporating a secondary cache, the system address look up performance can be significantly improved. The cache is used to map MAC destination addresses to output port numbers and to check for appearance of new MAC source addresses for learning. The LS100 compares source addresses on all incoming packets. If there is no match, it generates an ATC miss interrupt. The LS105 has 8 interrupt lines to support up to 8 LS100s. In the absence of the LS105, the CPU has to learn this new address and load the port mapping for this new address from the main memory. The LS105 secondary ATC controller off-loads the task from the CPU.

Overview

Address lookup

The LS105 uses a pipelined architecture to allow address lookups to occur in parallel to PCI read and write cycles. When an LS100 receives a packet which does not have an address to port mapping in the LS100 Primary ATC, the LS100 will generate an ATC_Miss# interrupt to the LS105 (if present in the system). The LS105 will read two double words from the LS100 (composed of a 48 bit MAC address and 16 bits of extra information, including the port number which the packet was received on). The LS105 will generate a 16 bit hash address from the received 48 bit MAC address. The LS105 stores address mappings within quad entry "buckets" pointed to by the hash value. Address mapping lookup operations compare the received 48 bit MAC address to up to 4 addresses stored within the quad entry bucket until a hit is found. Assuming that the mapping is present in the LS105 SATC system, the address mapping (MAC address + port number) is automatically loaded into the LS100 ATC, with the appropriate VLAN information (see VLAN section).

Address Learning

There are two learning modes supported by the LS105; Auto learning, and CPU-directed learning. In Auto learning mode, the LS105 acts as the central address mapping database. All mappings are stored in the LS105, and cached in the LS100 primary ATC. In CPU-directed learning mode, the LS105 acts as a secondary ATC; the central address mapping database is held in CPU storage, and address mappings are cached in the LS105 secondary ATC and the LS100s' primary ATCs.

Memory configuration

The LS105 SATC utilizes standard x8 or x16 asynchronous SRAMs to maintain a secondary cache of port mappings. A CRC 16 algorithm is used to generate the hash value and each hash points to a bucket of four MAC addresses. The following table shows the different possible SRAM configurations to achieve different cache sizes.

# of Hash pointers	Max no. of MAC Addresses cached	Address Cache SRAM	Multicast type field support SRAM
4K	16K	2 x 32Kx16 or 4 x 32Kx8	1 x 32Kx16 or 2 x 32Kx8
8K	32K	2 x 64Kx16	1 x 64Kx16
32K	64K	2 x 256Kx16	1 x 256Kx16

Table 1: Memory configuration table

VLAN

The LS105 uses a 32x32 bit system wide port map register to support Port based VLAN filtering. This port map stores the overall VLAN relationship between 32 possible source and destination ports. The map compares source ports to destination ports. Each column represents one source port. Each row represents one destination port. Each row is divided up into eight 4 bit segments, each of which contain the usage bits representing the four ports in a particular LS100.

The LS100 implements port based VLAN by using the usage bits to control the use of ATC mappings for Unicast traffic. If the usage bit is set for a particular port, Unicast packets arriving at that port can be forwarded using the ATC mapping. If the usage bit is not set, Unicast packets arriving at that port will be denied the use of that mapping and will be forwarded according to the set default behavior. Multicasts undergo ATC lookup operations at the LS100 Multicast output queue. Usage bits are used as vectors for Multicast traffic; if the usage bit for a particular port in a particular ATC mapping is set, multicast traffic can flow out of that port. If the bit is not set, multicast traffic will not flow out of that port.

Address mapping entries in the external LS105 SRAM are stored with source port information. When an address mapping entry is loaded from the LS105 into the LS100 ATC during an ATC miss operation, the source port information is used to select the VLAN information from the appropriate port map register row. The 4 usage bits corresponding to the LS100 into whose ATC the entry will be loaded are drawn from that row, and stored with the mapping in the LS100 ATC. If no VLAN support SRAM is present, the 4 usage bits are drawn from the port map register row. If the VLAN support SRAM is present, the 4 usage bits are drawn from the VLAN support SRAM and multicast MAC address. The usage bits will control which ports in the LS100 can utilize that mapping.

The LS105 SRAM can be optionally extended to support mulitcast type filtering. The optional VLAN support SRAM is required for multicast type field filtering because the source port ID for Multicast packets is set to FE and hence is not available for the VLAN table look up operation. When an address is learned, and the VLAN support SRAM is present, a 1x32 bit slice (row) from the system wide port map register is stored adjacent to the MAC address in the parallel VLAN support SRAM. This bit slice indicates the VLAN port numbers making up the particular VLAN which that MAC address belongs to. Each MAC address is stored as two double words (2x32 bits). Each VLAN slice is stored as two words (2x16 bits).

Pin Description

Pin Name	Type	Description
PCI_AD [31:0]	Bi-directional	PCI Address/Data Bus
PCI_CBE [3:0]#	Bi-directional	PCI Bus Command and Byte Enable
PCI_RESET#	Input	PCI Reset Signal
PCI_PAR	Bi-directional	PCI Parity Signal
PCI_CLK	Input	PCI Clock Input
PCI_STOP#	Bi-directional	PCI Transaction Stop Signal
PCI_FRAME#	Bi-directional	PCI Frame Signal
PCI_DEVSEL#	Bi-directional	PCI Device Select Signal
PCI_IRDY#	Bi-directional	PCI Initiator Ready
PCI_TRDY#	Bi-directional	PCI Target Ready Signal
PCI_IDSEL	Input	PCI Initialization Device Select Signal
SATC_MISS#	Open Drain Output	Secondary Address Translation Cache Miss Interrupt
PCI_INTA#	Open Drain Output	PCI Interrupt Request A
PCI_REQ#	Open Drain Output	PCI Bus Request
PCI_GNT#	Input	PCI Bus Grant

Table 2: PCI Interface Pin Description

All PCI Interface pin pads do not have internal resistors.

Pin Name	Type	Description
M_DATA [47:0]	Bi-directional	SRAM Databus Internal pull-down resistor
M_A [17:0]	Output	SRAM Address Bus Internal pull-down resistor
M_OE#	Output	SRAM Output Enable Signal Internal pull-up resistor
M_WE0#	Output	SRAM Write Signal for M_DATA [31:0] Internal pull-up resistor
M_WE1#	Output	SRAM Write Signal for M_DATA [47:32] (VLAN) Internal pull-up resistor

Table 3: SRAM Interface Pin Description

Pin Name	Type	Description
TCK	Input	JTAG Test Clock Internal pull-up resistor
TDI	Input	JTAG Test Data In Internal pull-up resistor
TDO	Output	JTAG Test Data Out Tristate Output
TMS	Input	JTAG Test Mode Select Internal pull-down resistor

Table 4: JTAG Interface Pin Description

Pin Name	Type	Description
MDC	Output	MII Serial Clock Output for LS101 Internal pull-up resistor
MDIO	Bi-directional	MII Data Input/Output for LS101 Internal pull-up resistor

Table 5: MDC/MDIO Interface Pin Description

Pin Name	Type	Description
ATC_MISS [7:0]#	Input	Address Translation Cache Miss Interrupt from LS100 Internal pull-up resistor

Table 6 : LS100 Interface Pin Description

Pin Name	Type	Description
µA_RST	Input	µAccess Reset This reset pin only resets the µAccess interface. Internal pull-up resistor
µA_RD#	Input	µAccess Read Internal pull-up resistor
µA_WR#	Input	µAccess Write Internal pull-up resistor
µA_CS#	Input	µAccess Chip Select Internal pull-up resistor
µA_ALE	Input	µAccess Address Latch Enable Internal pull-up resistor
µA_AD[7:0]	Bi-directional	µAccess Address/Data Bus Internal pull-up resistor
TEST	Input	Test pin. Reserved for future use. This pin must be tied to gnd. Internal pull-down resistor
GPIO[7]/µA_EN#	Bi-directional	User Definable General Purpose I/O/µAccess Enable This pin must be tied to gnd through a 1K ohm resistor to enable µAccess Interface. Internal pull-down resistor
GPIO[6:0]	Bi-directional	User Definable General Purpose I/O Pins Internal pull-down resistor

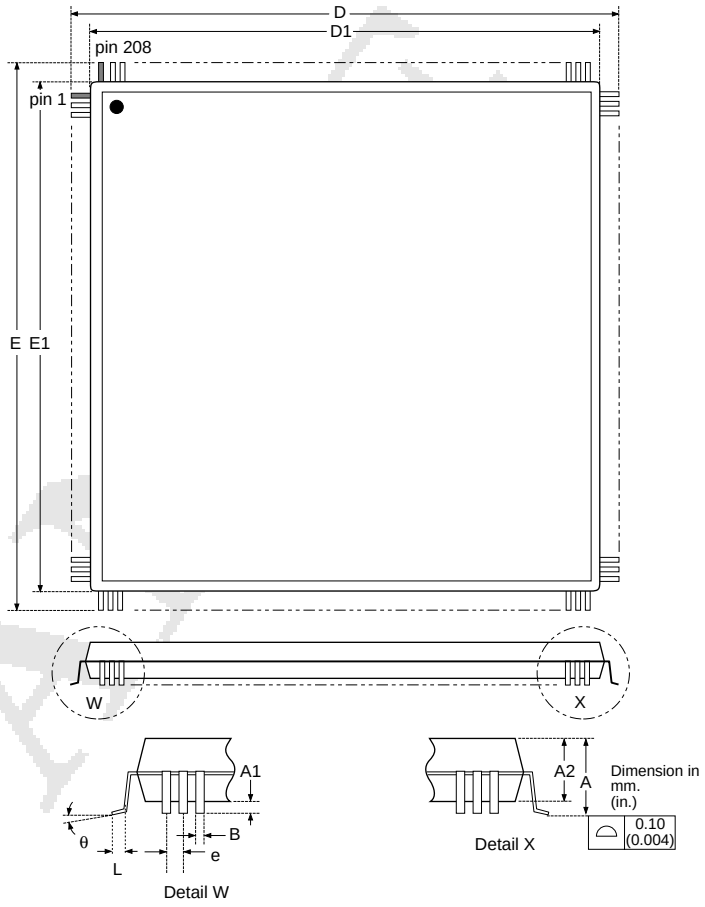
Table 7: µAccess Interface Pin Description

Pin Name	Type	Description
V_{DD}	Power	+3.3 V Power for the chip
V_{SS}	Ground	Ground for the chip

Table 8: Power Interface

Mechanical Specifications

PQFP 208 Lds Package Dimensions



Symbol	Min mm	Max mm	Min inch	Max inch
A	-	3.99	-	0.157
A1	0.25	0.43	0.010	0.017
A2	3.43	3.56	0.135	0.140
D	30.40	30.91	1.195	1.215
D1	27.93	28.14	1.098	1.106
E	30.40	30.91	1.195	1.215
E1	27.93	28.14	1.098	1.106
L	0.45	0.76	0.018	.030
B	0.15	0.28	0.006	0.011
e	0.50 BSC		0.0197 BSC	

Table 9: PQFP/208 Packeage Dimensions

Note: Use mm as the controlling dimension when used for building PCB footprint.

Pinout

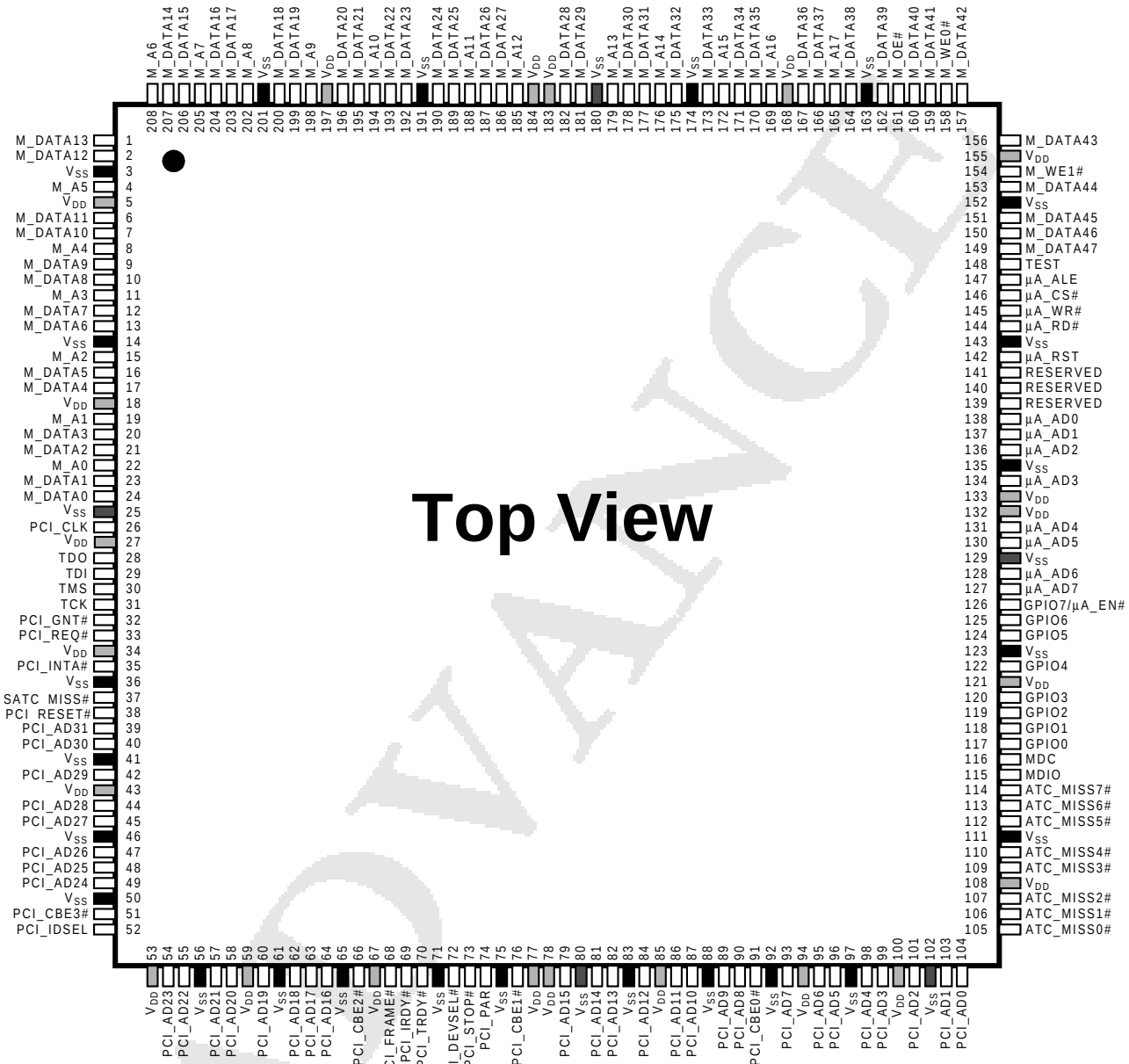
LS105 [PQFP/208L Package] Pinout: By Pin Name

Name	Pin #	Name	Pin #	Name	Pin #	Name	Pin #
µA_AD0	138	SATC_MISS#	37	M_A11	188	M_DATA42	157
µA_AD1	137	PCI_INTA#	35	M_A12	185	M_DATA43	156
µA_AD2	136	PCI_CBE0#	91	M_A13	179	M_DATA44	153
µA_AD3	134	PCI_CBE1#	76	M_A14	176	M_DATA45	151
µA_AD4	131	PCI_CBE2#	66	M_A15	172	M_DATA46	150
µA_AD5	130	PCI_CBE3#	51	M_A16	169	M_DATA47	149
µA_AD6	128	PCI_AD0	104	M_A17	165	M_OE#	161
µA_AD7	127	PCI_AD1	103	M_DATA0	24	M_WE0#	158
µA_ALE	147	PCI_AD2	101	M_DATA1	23	M_WE1#	154
µA_CS#	146	PCI_AD3	99	M_DATA2	21	V_{ss}	3
µA_WR#	145	PCI_AD4	98	M_DATA3	20	V_{ss}	14
µA_RD#	144	PCI_AD5	96	M_DATA4	17	V_{ss}	36
µA_RST	142	PCI_AD6	95	M_DATA5	16	V_{ss}	41
RESERVED	141	PCI_AD7	93	M_DATA6	13	V_{ss}	46
RESERVED	140	PCI_AD8	90	M_DATA7	12	V_{ss}	50
RESERVED	139	PCI_AD9	89	M_DATA8	10	V_{ss}	56
TEST	148	PCI_AD10	87	M_DATA9	9	V_{ss}	61
ATC_MISS0#	105	PCI_AD11	86	M_DATA10	7	V_{ss}	65
ATC_MISS1#	106	PCI_AD12	84	M_DATA11	6	V_{ss}	71
ATC_MISS2#	107	PCI_AD13	82	M_DATA12	2	V_{ss}	75
ATC_MISS3#	109	PCI_AD14	81	M_DATA13	1	V_{ss}	83
ATC_MISS4#	110	PCI_AD15	79	M_DATA14	207	V_{ss}	88
ATC_MISS5#	112	PCI_AD16	64	M_DATA15	206	V_{ss}	92
ATC_MISS6#	113	PCI_AD17	63	M_DATA16	204	V_{ss}	97
ATC_MISS7#	114	PCI_AD18	62	M_DATA17	203	V_{ss}	102
GPIO0	117	PCI_AD19	60	M_DATA18	200	V_{ss}	111
GPIO1	118	PCI_AD20	58	M_DATA19	199	V_{ss}	123
GPIO2	119	PCI_AD21	57	M_DATA20	196	V_{ss}	135
GPIO3	120	PCI_AD22	55	M_DATA21	195	V_{ss}	143
GPIO4	122	PCI_AD23	54	M_DATA22	193	V_{ss}	152
GPIO5	124	PCI_AD24	49	M_DATA23	192	V_{ss}	163
GPIO6	125	PCI_AD25	48	M_DATA24	190	V_{ss}	174
GPIO7/ µA_EN#	126	PCI_AD26	47	M_DATA25	189	V_{ss}	191
MDC	116	PCI_AD27	45	M_DATA26	187	V_{ss}	201
MDIO	115	PCI_AD28	44	M_DATA27	186	V_{ss}	25
TDO	28	PCI_AD29	42	M_DATA28	182	V_{ss}	80
TDI	29	PCI_AD30	40	M_DATA29	181	V_{ss}	129
TMS	30	PCI_AD31	39	M_DATA30	178	V_{ss}	180
TCK	31	M_A0	22	M_DATA31	177	V_{DD}	5
PCI_RESET#	38	M_A1	19	M_DATA32	175	V_{DD}	18
PCI_PAR	74	M_A2	15	M_DATA33	173	V_{DD}	34
PCI_CLK	26	M_A3	11	M_DATA34	171	V_{DD}	43
PCI_STOP#	73	M_A4	8	M_DATA35	170	V_{DD}	53
PCI_FRAME#	68	M_A5	4	M_DATA36	167	V_{DD}	59
PCI_DEVSEL#	72	M_A6	208	M_DATA37	166	V_{DD}	67
PCI_IRDY#	69	M_A7	205	M_DATA38	164	V_{DD}	77
PCI_TRDY#	70	M_A8	202	M_DATA39	162	V_{DD}	85
PCI_IDSEL	52	M_A9	198	M_DATA40	160	V_{DD}	94
PCI_REQ#	33	M_A10	194	M_DATA41	159	V_{DD}	100
PCI_GNT#	32	V_{DD}	155	V_{DD}	197	V_{DD}	108
V_{DD}	121	V_{DD}	168	V_{DD}	27	V_{DD}	132
V_{DD}	133	V_{DD}	184	V_{DD}	78	V_{DD}	183

LS105 [PQFP/208L Package] Pinout: By Pin Sequence

Pin #	Pin Name	Pin #	Pin Name	Pin #	Pin Name	Pin #	Pin Name
1	M_DATA13	53	V_{DD}	105	ATC_MISS0#	157	M_DATA42
2	M_DATA12	54	PCI_AD23	106	ATC_MISS1#	158	M_WE0#
3	V_{SS}	55	PCI_AD22	107	ATC_MISS2#	159	M_DATA41
4	M_A5	56	V_{SS}	108	V_{DD}	160	M_DATA40
5	V_{DD}	57	PCI_AD21	109	ATC_MISS3#	161	M_OE#
6	M_DATA11	58	PCI_AD20	110	ATC_MISS4#	162	M_DATA39
7	M_DATA10	59	V_{DD}	111	V_{SS}	163	V_{SS}
8	M_A4	60	PCI_AD19	112	ATC_MISS5#	164	M_DATA38
9	M_DATA9	61	V_{SS}	113	ATC_MISS6#	165	M_A17
10	M_DATA8	62	PCI_AD18	114	ATC_MISS7#	166	M_DATA37
11	M_A3	63	PCI_AD17	115	MDIO	167	M_DATA36
12	M_DATA7	64	PCI_AD16	116	MDC	168	V_{DD}
13	M_DATA6	65	V_{SS}	117	GPIO0	169	M_A16
14	V_{SS}	66	PCI_CBE2#	118	GPIO1	170	M_DATA35
15	M_A2	67	V_{DD}	119	GPIO2	171	M_DATA34
16	M_DATA5	68	PCI_FRAME#	120	GPIO3	172	M_A15
17	M_DATA4	69	PCI_IRDY#	121	V_{DD}	173	M_DATA33
18	V_{DD}	70	PCI_TRDY#	122	GPIO4	174	V_{SS}
19	M_A1	71	V_{SS}	123	V_{SS}	175	M_DATA32
20	M_DATA3	72	PCI_DEVSEL#	124	GPIO5	176	M_A14
21	M_DATA2	73	PCI_STOP#	125	GPIO6	177	M_DATA31
22	M_A0	74	PCI_PAR	126	GPIO7/µA_EN#	178	M_DATA30
23	M_DATA1	75	V_{SS}	127	µA_AD7	179	M_A13
24	M_DATA0	76	PCI_CBE1#	128	µA_AD6	180	V_{SS}
25	V_{SS}	77	V_{DD}	129	V_{SS}	181	M_DATA29
26	PCI_CLK	78	V_{DD}	130	µA_AD5	182	M_DATA28
27	V_{DD}	79	PCI_AD15	131	µA_AD4	183	V_{DD}
28	TDO	80	V_{SS}	132	V_{DD}	184	V_{DD}
29	TDI	81	PCI_AD14	133	V_{DD}	185	M_A12
30	TMS	82	PCI_AD13	134	µA_AD3	186	M_DATA27
31	TCK	83	V_{SS}	135	V_{SS}	187	M_DATA26
32	PCI_GNT#	84	PCI_AD12	136	µA_AD2	188	M_A11
33	PCI_REQ#	85	V_{DD}	137	µA_AD1	189	M_DATA25
34	V_{DD}	86	PCI_AD11	138	µA_AD0	190	M_DATA24
35	PCI_INTA#	87	PCI_AD10	139	RESERVED	191	V_{SS}
36	V_{SS}	88	V_{SS}	140	RESERVED	192	M_DATA23
37	SATC_MISS#	89	PCI_AD9	141	RESERVED	193	M_DATA22
38	PCI_RESET#	90	PCI_AD8	142	µA_RST	194	M_A10
39	PCI_AD31	91	PCI_CBE0#	143	V_{SS}	195	M_DATA21
40	PCI_AD30	92	V_{SS}	144	µA_RD#	196	M_DATA20
41	V_{SS}	93	PCI_AD7	145	µA_WR#	197	V_{DD}
42	PCI_AD29	94	V_{DD}	146	µA_CS#	198	M_A9
43	V_{DD}	95	PCI_AD6	147	µA_ALE	199	M_DATA19
44	PCI_AD28	96	PCI_AD5	148	TEST	200	M_DATA18
45	PCI_AD27	97	V_{SS}	149	M_DATA47	201	V_{SS}
46	V_{SS}	98	PCI_AD4	150	M_DATA46	202	M_A8
47	PCI_AD26	99	PCI_AD3	151	M_DATA45	203	M_DATA17
48	PCI_AD25	100	V_{DD}	152	V_{SS}	204	M_DATA16
49	PCI_AD24	101	PCI_AD2	153	M_DATA44	205	M_A7
50	V_{SS}	102	V_{SS}	154	M_WE1#	206	M_DATA15
51	PCI_CBE3#	103	PCI_AD1	155	V_{DD}	207	M_DATA14
52	PCI_IDSEL	104	PCI_AD0	156	M_DATA43	208	M_A6

LS105 [PQFP/208L Package] Pinout



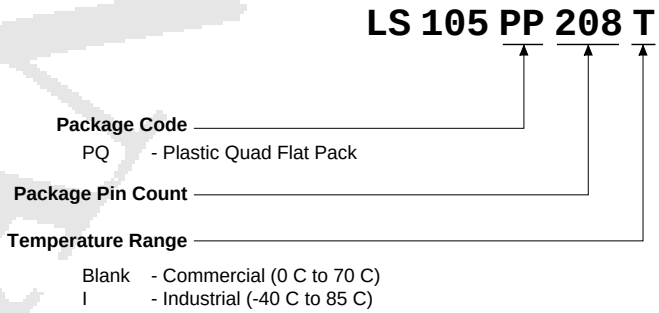
Component Availability and Ordering Information

The following table lists the LS105 package options, and operating temperature ranges that are currently available. Contact I-Cube Marketing for more up-to-date information.

Package Pins	Package Type	Package Code	Temperature Range
208	PQFP	PQ208	C

C = Commercial = 0° to +70° C

Table 10: Component Availability



I-Cube®, Inc.
2605 S. Winchester Blvd.
Campbell, CA 95008, USA

Phone: + (408) 341-1888
Fax: + (408) 341-1899
eMail: marketing@icube.com
Internet: http://www.icube.com

LS105 Data Sheet
July 1998
[Advance - v1.01]
Document #D-11-023