The Embedded I/O Company

TEWS TECHNOLOGIES

TIP360

Quad Integrated Communication IP based on MC68360

Version 1.0

User Manual

Issue 1.0.1

January 2011

TEWS TECHNOLOGIES GmbH
Am Bahnhof 7 25469 Halstenbek, Germany
Phone: +49 (0) 4101 4058 0 Fax: +49 (0) 4101 4058 19
e-mail: info@tews.com www.tews.com

TIP360-10

Quad Integrated Communication IP based on MC68360

This document contains information, which is proprietary to TEWS TECHNOLOGIES GmbH. Any reproduction without written permission is forbidden.

TEWS TECHNOLOGIES GmbH has made any effort to ensure that this manual is accurate and complete. However TEWS TECHNOLOGIES GmbH reserves the right to change the product described in this document at any time without notice.

TEWS TECHNOLOGIES GmbH is not liable for any damage arising out of the application or use of the device described herein.

Style Conventions

Hexadecimal characters are specified with prefix 0x, i.e. 0x029E (that means hexadecimal value 029E).

For signals on hardware products, an ‚Active Low' is represented by the signal name with # following, i.e. IP_RESET#.

Access terms are described as:

W	Write Only
R	Read Only
R/W	Read/Write
R/C	Read/Clear
R/S	Read/Set

Issue	Description	Date
	Preliminary Issue	May 2007
1.0	First Issue	September 2007
1.0.1	New Notation for User Manual and Engineering Documentation	January 2011

Table of Contents

List of Figures

List of Tables

1 Product Description

The TIP360 is an IndustryPack® compatible module with a serial communication I/O interface using the Motorola MC68360 Quad Integrated Communication Processor running at 25 MHz.

The MC68360 contains a CPU32 core processor, a micro coded RISC communication processor, 14 serial DMA channels, two independent DMAs, 4 general purpose timers, an interrupt controller, 4 independent full-duplex serial communication controllers (SCCs), two Serial Management Controllers (SMCs) and 2.5 kbytes Dual-Port of internal RAM.

The TIP360 provides TTL level interface for all 4 SCC's of the MC68360 at the IP I/O connector. A variety of physical layer standards, such as RS232, RS422, RS485, ISDN or IEEE802.3/Ethernet can be supported by the development of custom transition modules.

Internal microcode in the MC68360 supports a variety of protocols, such as BISYNC, HDLC, SDLC, UART, DDCMP, AppleTalk, Signaling System #7, Profibus, ATM, V.14, X.21, ISDN and totally transparent modes.

The TIP360 provides additional to the internal RAM 256 kbyte of SRAM, 256 kbyte of Flash memory, 2 Mbytes of "DRAM" (emulated with SRAM and on board logic to maintain existing 68360 board configuration) as dual-port memory. Selection of boot memory (SRAM or Flash memory) is done by a control register. The code is downloaded to the SRAM or Flash memory by the IP host CPU prior to removing the MC68360 reset signal by writing to a control register. Once the MC68360 reset signal is removed, the MC68360 starts fetching code from boot memory.

The SRAM, Flash memory, DRAM, the MC68360 registers and the MC68360 on-chip RAM are accessible by the MC68360 and the IP host CPU by arbitration. The local bus arbitration is transparent to the IP host CPU.

Both the IP host CPU and the MC68360 are capable of generating interrupts to each other.

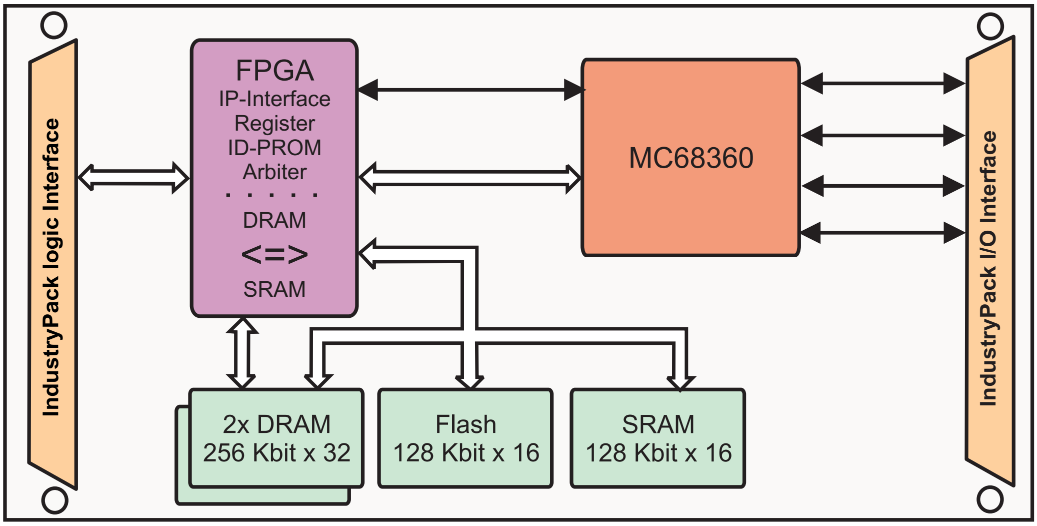


Figure 1-1 : Block Diagram

2 Technical Specification

IP Interface	
Interface	Single Size IndustryPack® Logic Interface compliant to ANSI/VITA 4-1995
ID ROM Data	Format I
I/O Space	Used without wait states
Memory Space	MC68360 RESET STATE - 128 kbit SRAM without wait state - 128 kbit Flash without wait state MC68360 RUNNING STATE - 128 kbit SRAM with one wait state - 128 kbit Flash with two wait states - 512 kbit emulated DRAM with one wait state
Interrupts	INTREQ0# is used for Interrupts from the MC68360 QUICC
DMA	Not supported
Clock Rate	8 MHz
Module Type	Type II
On Board Devices	
Processor	Freescale MC68360- CVR (25 MHz), including CPU32+ Processor, Memory Controller and four SCCs.
Memory	128 kbit x 16 (256 kByte) SRAM 128 kbit x 16 (256 kByte) Flash 512 kbit x 32 (2 MByte) emulated DRAM
I/O Interface	
Interface Connector	50-conductor flat cable
Number of Serial Channels	4
Interface	TTL level interface for all 4 SCC’s
Functional Features	see MC68360 QUICC User Manual
Power Requirements	300 mA typical @ +5V DC
Physical Data	
Temperature Range	Operating -40 °C to +85 °C Storage -40 °C to +85 °C
MTBF	1052000 h MTBF values shown are based on calculation according to MIL-HDBK-217F and MIL-HDBK-217F Notice 2; Environment: G_B 20°C. The MTBF calculation is based on component FIT rates provided by the component suppliers. If FIT rates are not available, MIL-HDBK-217F and MIL-HDBK-217F Notice 2 formulas are used for FIT rate calculation.
Humidity	5 – 95 % non-condensing
Weight	31 g

Table 2-1 : Technical Specification

3 ID PROM Contents

Address	Function	Contents
0x01	ASCII ‘I’	**0x49**
0x03	ASCII ‘P’	**0x50**
0x05	ASCII ‘A’	**0x41**
0x07	ASCII ‘C’	**0x43**
0x09	Manufacturer ID	**0xB3**
0x0B	Model Number	**0x3E**
0x0D	Revision	**0x10**
0x0F	Reserved	**0x00**
0x11	Driver-ID Low - Byte	**0x00**
0x13	Driver-ID High - Byte	**0x00**
0x15	Number of bytes used	**0x0C**
0x17	CRC	**0x75**
0x19 **…** **0x3F**	Not used.	**0x00**

Table 3-1 : ID PROM Contents

4 Address Map(s)

4.1 I/O Addressing

The control register set of the TIP360 is accessible in the IP I/O space.

The control registers are automatically cleared by assertion of IP Bus RESET#.

Address	Symbol	Description	Size (Bit)	Access
0x01	RESETH	Hardware Reset Register	8	R/W
0x03	BOOT	Boot Memory Select Register	8	R/W
0x05	IVEC	Interrupt Vector Register	8	R/W
0x07	ICNT	Interrupt Control Register	8	R/W
0x09	FCODE	Function Code Register	8	R/W
0x0A	TEST_REG	16 bit wide R/W Test Register	16	R/W
0x0C … 0x79	-	Not used and read always as ‘0’	-	R

Table 4-1 : Control Register Set

The Test Register is for TEWS factory test only.

4.2 IP Memory Address Map

4.2.1 MC68360 in Reset State

During PowerUp, IP hardware reset and until the user disables the RST-control bit in the Hardware Reset Register of the TIP360 the MC68360 is kept in reset state.

SRAM and Flash memory could be read and written by 8 bit or 16 bit IP memory accesses. All other MC68360 functions are not accessible as long as the reset for the controller is asserted.

Address	Symbol	Description	Size (Bit)	Access
0x00_0000 … 0x03_FFFF	SRAM	256 kByte SRAM memory	16	R /W
0x08_0000 … 0x0B_FFFF	FLASH	256 kByte Flash memory	16	R /W

Table 4-2 : SRAM / Flash Memory Map in Reset State

4.2.2 MC68360 in Running State

After switching the RST-control bit in the Hardware Reset Register off, the TIP360 memory address map changes. The established MC68360 address map is now also valid for the IP Bus Host access. The IP Host CPU must allocate a minimum of 8 MByte to the IP memory space and locate this memory space on an 8 MByte boundary (for example: 0xC000.0000).

SRAM, Flash, "DRAM" and all internal registers of the MC68260 are then accessible from the IP Bus interface.

Address	Symbol	Description	Size (Bit)	Access
0xC000_0000 … 0xC003_FFFF	CS0	256 kByte SRAM memory or 256 kByte Flash memory - depends on BOOT Register adjustment	16	R /W
0xC004_0000 … 0xC00B_FFFF	CS3	256 kByte Flash memory or 256 kByte SRAM memory - depends on BOOT Register adjustment	16	R /W
0xC008_0000 … 0xC00F_FFFF	-	Not used	-	-
0xC010_0000 … 0xC01F_FFFF	RAS1	1 MByte DRAM	32	R/W
0xC020_0000 … 0xC02F_FFFF	RAS2	1 MByte DRAM	32	R/W
0xC030_0000 … 0xC030_1FFF	MC68360	MC68360 internal RAM and Register	32	R/W
0xC030_2000 … 0xC03F_FFFF	-	Not used	-	-

Table 4-3 : Memory Map in Running State

To use the memory resources a minimum initialization code must be downloaded to the SRAM or Flash Boot memory before switching the MC68360 into the Running State. This boot code must init Base Address Register, Clock Control Register, Global Memory Register and some few other register to meet the on board hardware requirements.

4.3 MC68360 Memory Address Map

The following figure shows a typical memory address map of the MC68360. This memory map is based on the boot code example in chapter 10.1 IP Host Access.

Other configurations are possible, but note that according boot configuration of the MC68360 must be placed at the CS0# memory space.

Address	Symbol	Description	Size (Bit)	Access
0x00_0000 … 0x03_FFFF	CS0	256 kByte SRAM memory or 256 kByte Flash memory - depends on BOOT Register adjustment	16	R /W
0x04_0000 … 0x0B_FFFF	CS3	256 kByte Flash memory or 256 kByte SRAM memory - depends on BOOT Register adjustment	16	R /W
0x08_0000 … 0x0F_FFFF	-	Not used	-	-
0x10_0000 … 0x1F_FFFF	RAS1	1 MByte DRAM	32	R/W
0x20_0000 … 0x2F_FFFF	RAS2	1 MByte DRAM	32	R/W
0x30_0000 … 0x30_1FFF	MC68360	MC68360 internal RAM and Register	32	R/W
0x30_2000 … 0x37_FFFF	-	Not used	-	-
0x38_0000 … 0x3F_FFFF	MAILBOX	Mail-Box	8	R/W

Table 4-4 : Example of MC68360 Memory Map

5 Register Description

5.1 Hardware Reset Register (RESETH)

The Hardware Reset Register is a byte wide read/write register to control reset and running state of the MC68360 controller. The register also indicates a Bus Error during access to the MC68360 resources.

. Bit	Symbol	Description	Access	Reset Value
7:3	-	Always read as ‘0’	R	
2	BERR	Bus Error 0 = Last host access to MC68360 resources was acknowledged 1 = Last host access to MC68360 resources resulted in bus error Always cleared by reading.	R/W	0
1	STATUS	Hardware Reset Status bit. Reflects the MC68360 RESETH pin logic level. 0 = CPU or 68360 is asserting the RESETH# line 1 = RESETH# is not asserted	R	0
0	RST	RESETH Control Bit: 0 = RESETH# line is asserted. 1 = RESETH# line is not asserted.	R/W	0

Table 5-1 : Hardware Reset Register (Address 0x01)

The BERR bit of the Hardware Reset Register should be polled by the IP Host to notice any MC68360 bus problem.

5.2 Boot Memory Select Register (BOOT)

The Boot Memory Select Register is a byte wide read/write register to select the default boot memory (SRAM or Flash).

. Bit	Symbol	Description	Access	Reset Value
7:1	-	Always read as ‘0’	R	
0	BOOT	Selects the boot memory bank which the MC68360 boots from: 0 = SRAM is boot memory 1 = Flash is boot memory	R/W	0

Table 5-2 : Boot Memory Select Register (Address 0x03)

5.3 Interrupt Vector Register (IVEC)

The Interrupt Vector Register can be read and written from the IP Host and from the MC68360.

Bit	Symbol	Description	Access	Reset Value
7:0		Interrupt vector loaded by software	R/W	0x00

Table 5-3 : Interrupt Vector Register (Address 0x05)

The Interrupt Vector Register can be used as an IP interrupt vector register which is initialized by the IP Host Controller. Each interrupt request provides this vector.

Additionally the Interrupt Vector Register can be loaded by the MC68360 before initiating an IP Interrupt Request. This allows the use of different vectors.

Each interrupt which is requested to the IP bus needs to be acknowledged and cleared by an IP Interrupt Acknowledge cycle.

5.4 Interrupt Register (IR)

The Interrupt Register can be used to initiate an Interrupt Request to the MC68360 from the IP Host Controller.

An MC68360 Interrupt Acknowledge cycle clears this register by hardware.

Bit	Symbol	Description	Access	Reset Value
7	IR7	Setting the IR bit to ‘1’ generates an interrupt on the corresponding IRQ Level of the MC68360. 1 = interrupt is generated 0 = no interrupt is generated Setting bit 1 generates an interrupt on level IRQ1. Likewise the other 6 bits.	R/W	0
6	IR6			0
5	IR5			0
4	IR4			0
3	IR3			0
2	IR2			0
1	IR1			0
0	-	Not used		0

Table 5-4 : Interrupt Register (Address 0x07)

The user may set only one bit at a time as the complete register is cleared by the MC68360 Interrupt Acknowledge cycle!

5.5 Function Code Register (FCR)

The Function Code Register provides to write the function code to the MC68360 bus during an IP Host access.

An MC68360 Interrupt Acknowledge cycle clears this register by hardware.

Bit	Symbol	Description	Access	Reset Value
7:4	-	Not used	-	0x00
3	FC3	FC3 is driven always as 0.	R	0
2	FC2	The value of FC2 is driven to the corresponding FC2# line of the MC68360 bus.	R/W	0
1	FC1	The value of FC1 is driven to the corresponding FC1# line of the MC68360 bus.	R/W	0
0	FC0	The value of FC0 is driven to the corresponding FC0# line of the MC68360 bus.	R/W	0

Table 5-5 : Function Code Register (Address 0x09)

The FCR must be programmed to 0x05 or 0x06 before the IP Host can read or write to the internal register of the MC68360.

6 Interrupts

6.1 Interrupt Sources

The IP Host Controller can interrupt the MC68360 via the Interrupt Register (IR) and the MC68360 can interrupt the IP Host via Mail-Box memory access.

6.1.1 Generate an Interrupt to the IP Bus

An Interrupt form the MC68360 to the IP Host controller is generated by writing to the Mail-Box memory space.

Two different functions are implemented. If the address bit A1 is set to logic level 0 the MC68360 is able to read or write the Interrupt Vector Register (IVR). This is used to initialize the IVR with a valid vector which points to the according interrupt routine on the IP Host.

If the address bit A1 is set to logic level 1 then a write to the Mail-Box memory space will generate an interrupt request on the IP Interrupt line INTREQ0#. Reading the Mail-Box memory space shows the interrupt status.

Interrupt Vector Register / Mail-Box = 0x380000

Bit	Symbol	Description	Access	Reset Value
7..0		Interrupt vector loaded by software	R/W	0x00

Table 6-1 : Interrupt Vector Register / Mail-Box

Interrupt Status / Release Register / Mail-Box = 0x380002

Bit	Symbol	Description	Access	Reset Value
7..1		Not used	-	0
0	Interrupt Status	This bit shows the status of the IP Host Interrupt Request INTREQ0#. 1 = Interrupt is still pending 0 = Interrupt was acknowledged	R	0
		Writing to Mail-Box address space with address bit A1 = 1 (as: 0x380002) will generate an IP Host Interrupt.	W	-

Table 6-2 : Interrupt Status/Release Register / Mail-Box

The interrupt status should be checked by the MC68360 before each new interrupt will generated to the IP Host.

6.1.2 Generate an Interrupt to the MC68360

The IP Host Controller can interrupt the MC68360 by setting any bit of the Interrupt Register. This will set the according Interrupt Level on the MC68360.

> **The user may set only one bit at a time as the complete register is cleared by the MC68360 Interrupt Acknowledge cycle!**

This bit is cleared automatically by an MC68360 auto vectored interrupt acknowledge cycle and can be used as an interrupt status bit by the IP Host.

> **This interrupt bit / status should be checked before a new interrupt is generated by the IP Host.**

7 Functional Procedures

7.1 IP Host Access

After Power-Up the TIP360 holds the MC68360 in reset state until the IP Host will release the RESETH# bit in the Hardware Reset Register. The IP Host can read or write to the SRAM or the Flash memory and is able to download the MC68360 boot code. All other memory spaces are not available during MC68360 reset state.

By setting the BOOT bit in the Boot Memory Select Register the boot memory is selected. This bit will assign the chip select line 0 (CS0) and chip select line 3 (CS3) of the MC68360 to SRAM or Flash memory. CS0 is the initial chip select line after reset. The MC68360 fetches the boot code from the memory which is assigned to CS0. Setting the Boot bit = 0 will assign CS0 to the SRAM and if the Boot bit = 1 will assign SC0 to the Flash memory.

Use the following MC68360 sample boot code to initialize all memory spaces, the clock control register, the global memory register and all necessarily used registers.

```
162-Bug>as 00000400
00000400 303C0007          MOVE.W      #$7,D0?
00000404 4E7B0001          MOVEC.L     D0,DFC?
00000408 41F90003 FF00     LEA.L       ($3FF00).L,A0?
0000040E 203C0030 0001     MOVE.L      #$300001,D0?
00000414 0E900800          MOVES.L     D0,(A0)?
00000418 23FC0000 6CBF0030 \
         1000              MOVE.L      #$6CBF,($301000).L?
00000422 13FC0000 00301008 MOVE.B      #$0,($301008).L?
0000042A 13FC000C 00301022 MOVE.B      #$C,($301022).L?
00000432 13FC000C 0030100C MOVE.B      #$C,($30100C).L?
0000043A 33FC04C0 00301016 MOVE.W      #$4C0,($301016).L?
00000442 33FCD2F2 00301010 MOVE.W      #$D2F2,($301010).L?
0000044A 23FC0B00 00000030 \
         1040              MOVE.L      #$B800000,($301040).L?
00000454 23FC2FFC 00020030 \
         1054              MOVE.L      #$2FFC0002,($301054).L?
0000045E 23FC0000 02910030 \
         1050              MOVE.L      #$291,($301050).L?
00000468 23FC1FF0 00010030 \
         1064              MOVE.L      #$1FF00001,($301064).L?
00000472 23FC0010 02910030 \
         1060              MOVE.L      #$100291,($301060).L?
0000047C 23FC1FF0 00010030 \
         1074              MOVE.L      #$1FF00001,($301074).L?
00000486 23FC0020 02910030 \
         1070              MOVE.L      #$200291,($301070).L?
00000490 23FC3FFC 00020030 \
         1084              MOVE.L      #$3FFC0002,($301084).L?
0000049A 23FC0004 02910030 \
         1080              MOVE.L      #$40291,($301080).L?
000004A4 23FC2FF8 00040030 \
         1094              MOVE.L      #$2FF80004,($301094).L?
000004AE 23FC0038 02910030 \
         1090              MOVE.L      #$380291,($301090).L?
000004B8 6000FFFE          BRA.W       $C00004B8?
```

Table 7-1 : Example MC68360 boot code

This initialization code uses the MC68360 Memory Address Map described in chapter 4.3. It is used as an example to demonstrate minimum initialization code required for the MC68360. All memory like SRAM, Flash, DRAM bank 1, DRAM bank 2, internal RAM, internal register of the MC68360 and the Mail-box memory space are setup.

By setting the RESETH# bit to logic level 1 the IP Host releases the reset state of the MC68360 and the MC68360 will start to fetch the boot code. After this initialization the total memory space is accessible from MC68360 and from IP Host.

After changing the PLLCR Register of the MC68360 during initialization the internal PLL needs about 8 ms to recalibrate. During this time the IP Host must avoid accesses to the TIP360.

7.2 Initialization of the MC68360 Hardware Register

Some internal configuration Registers of the MC68360 must be initialized to guarantee a connection to the IP Host interface and the memory devices.

The MC68360 is hardware configured as a single master with CPU enabled, global chip select (CS0) = 16 bit sized bus and Module Base Address Register (MBAR) at 0x003FF00. The shown example of MC68360 boot code set up the following start up configuration.

Hardware configuration via configuration pins:

CONFIG2 … CONFIG0 : 0b101

16BM# : 0b1

MODCK1 … MODCK0 : TIP360-10 - 0b11

Register configuration via boot load:

Module Configuration Register (MCR):	`0x00006CBF`	- asynchronous bus timing - asynchronous arbitration - normal operation
Autovector Register (AVR):	`0x00`	- AVEC is provided externally
CLKO Control Register (CLKOCR):	`0x0C`	- CLKO1 enabled - CLKO2 disabled
PLL Control Register (PLLCR):	`0xD2F2`	- PLL enabled - Prescale disabled for TIP360-10 - MF11… MF0 => Multiplication Factor
Port E Assignment Register (PEPAR):	`0x04C0`	- CPU32+ Enable Mode - BCLRO# output function is used - QUICC normal mode - WE3# … WE0# function are used - AMUX output is used - CAS3# … CAS0# function are used - CS7# output function is used - AVEC# input function is used
Global Memory Register (GMR):	`0x0B800000`	- DRAM refresh is enabled - Refresh period = 15,36 µs - Refresh cycle = 4 clocks - DRAM port = 32 bit - Normal operation - No parity
Base Register (BR):	`0x00000291`	- Parity disabled - Base Address => see memory map - CS = low active - No function code - No timing relax
Option Register 0 (OR0):	`0x2FFC0002`	- 1 wait state if SRAM - 2 wait state if Flash - 16 bit port size - SRAM - Bank
Base Register 1 (BR1):	`0x00100291`	- Parity disabled - Base Address => see memory map - CS = low active - No function code - No timing relax
Option Register 1 (OR1):	`0x1FF00001`	- 1 wait state - 32 bit port size - DRAM - Bank
Base Register 2 (BR2):	`0x00200291`	- Parity disabled - Base Address => see memory map - CS = low active - No function code

		- No timing relax
Option Register 2 (OR2):	`0x1FF00001`	- 1 wait state - 32 bit port size - DRAM - Bank
Base Register 3 (BR3):	`0x00040291`	- Parity disabled - Base Address => see memory map - CS = low active - No function code - No timing relax
Option Register 3 (OR3):	`0x3FFC0002`	- 2 wait state if Flash - 1 wait state if SRAM - 16 bit port size - SRAM - Bank
Base Register 4 (BR4):	`0x00380291`	- Parity disabled - Base Address => see memory map - CS = low active - No function code - No timing relax
Option Register 4 (OR4):	`0x2FF80004`	- 1 wait state - 8 bit port size - SRAM – Bank

7.3 Read-Modify-Write

The TIP360 supports the Read-Modify-Write function of the MC68360 from the IP Host interface by using a special memory map. Access from IP Host to the memory space with address bit A22 = 1 will always initiate a Read-Modify-Write access. Please note that a write access is necessary after the read access cycle to release the MC68360 bus. Disregarding this will result in a dead-lock!

Address	Symbol	Description	Size (Bit)	Access
0xZZ40_0000…	CS0	256 kByte SRAM / Flash memory	16	R /W
0xZZ44_0000…	CS3	256 kByte Flash / SRAM memory	16	R /W
0xZZ48_0000…	-	Not used	-	-
0xZZ50_0000…	RAS1	1 MByte DRAM	32	R/W
0xZZ60_0000…	RAS2	1 MByte DRAM	32	R/W
0xZZ70_0000…	MC68360	MC68360 internal RAM and Register	32	R/W
0xZZ70_2000…	-	Not used	-	-

Table 7-2 : Read-Modify-Write Memory Map

7.4 Flash Memory

The Spansion S29GL0xxA Flash device requires to wait 4us after the last write of a command sequence, before any status information (i.e. any data) is read from the Flash device. Otherwise the read data can indicate wrong status information.

7.4.1 Used Spansion Flash Types

The used Spansion Flash (S29GL016A-R1) is a 2 Mbyte, 31 sectors a 64 kbyte, 8 Top Boot Sectors an 8 kbyte device which is configured with an 8 bit wide data bus.

Two 8 bit wide data bus configured Spansion Flash devices are combined up to a 16 bit wide data bus.

7.4.1.1 Command Cycles

Command Sequence	Cycles	1st Cycle		2nd Cycle		3rd Cycle		**4th Cycle**		**5th Cycle**		**6th Cycle**	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Reset	1	Base+ 0x0000	0xF0F0										
Auto-Select	4	Base+ 0x1554	0xAAAA	Base+ 0xAAA	0x5555	Base+ 0x1554	0x9090	See below for reading Manufacturer and Device Identifier					
Write to Buffer	5 + x	Base+ 0x1554	0xAAAA	Base+ 0xAAA	0x5555	SA	0x2525	SA	AC	PA	PD	WBL	PD
Program Buffer	1	SA	0x2929										
Write to Buffer Abort Reset	3	Base+ 0x1554	0xAAAA	Base+ 0xAAA	0x5555	Base+ 0x1554	0xF0F0						
Chip Erase	6	Base+ 0x1554	0xAAAA	Base+ 0xAAAA	0x5555	Base+ 0x1554	0x8080	Base+ 0x1554	0xAAAA	Base+ 0xAAA	0x5555	Base+ 0x1554	0x1010
Sector Erase	6	Base+ 0x1554	0xAAAA	Base+ 0xAAA	0x5555	Base+ 0x1554	0x8080	Base+ 0x1554	0xAAAA	Base+ 0xAAA	0x5555	SA	0x3030

Note:
All cycles shown are write cycles except the 4th, 5th & 6th cycle of the Auto-Select command, which are read cycles.
SA = Sector Address, AC = Address Count (Number of Write Buffer Locations to load minus 1),
PA = Program Address (Address must belong to the specified Sector), PD = Program Data,
WBL = Write Buffer Location (Address must be within same Write Buffer Page as PA. Each Write-Buffer-Page has a 32 Byte address boundary).

Table 7-3 : Spansion Flash Command Cycles

7.4.1.2 Auto-Select Identifier

ID Type	4th Cycle		5th Cycle		6th Cycle	
	Addr	Data	Addr	Data	Addr	Data
Spansion S29GL016A-R1						
Manufacturer ID	Base + 0x000	0x0101	---	---	---	---
Device ID	Base + 0x004	0xC4C4	---	---	---	---

Notes: All cycles shown are read cycles for the Auto-Select Command.

Table 7-4 : Spansion Flash Identifier

7.4.1.3 Write Buffer Programming

The Spansion S29GL0xxA Flash devices, when used in Byte-Mode, must be programmed using Write-Buffer programming.

The first two cycles of the Write-to-Buffer command are general unlock cycles.

The 3rd cycle writes the Write-Buffer-Load command to the sector address in which the programming occurs.

The 4th cycle writes the number of Bytes, minus one, to be programmed to the sector address in which the programming occurs.

The 5th cycle writes the first address location and data to the write buffer.

The actual write buffer page is selected by the 32 Byte address boundary used. Write buffer programming must not cross write buffer page or Flash sector boundaries.

Once the specified number of write buffer locations is loaded, the Program Buffer command must be used.

Data Polling should be used while monitoring the last address location loaded into the write buffer. DQ7, DQ6, DQ5 and DQ1 should be monitored to determine the device status during write buffer programming.

Upon successful completion of the Write Buffer Programming operation, the device is ready to execute the next command.

The abort condition is indicated by DQ1 = 1, DQ7 = DATA# (for the last address location loaded), DQ6 = toggle, and DQ5= 0. A Write-to-Buffer-Abort Reset command sequence must be written to reset the device for the next operation.

However, since the 16 bit Application Flash is build by using two parallel x8 Flash devices, any check shown for DQx means that for the Application Flash four datalines must be checked.

8 bit Flash Port Status	64 bit Flash Port Status
DQ7	DQ7, DQ15
DQ5	DQ5, DQ13
DQ1	DQ1, DQ8

Table 7-5 : Spansion Application Flash Status Bits

7.4.1.4 Flash Flowcharts

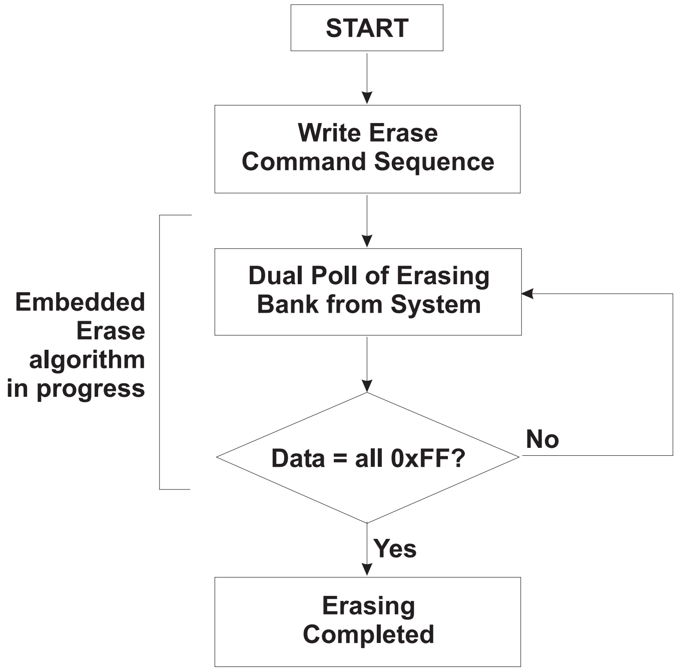
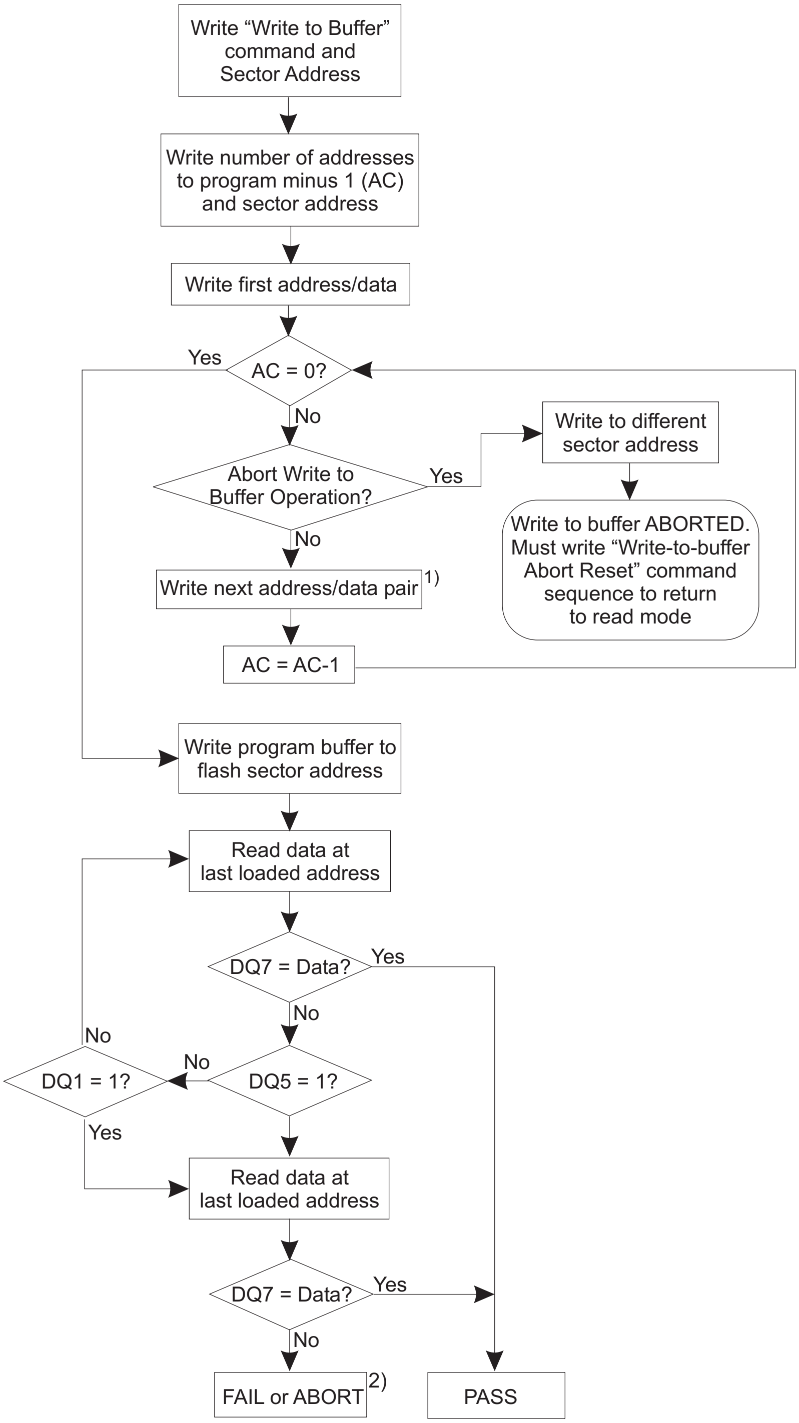

START
Write Erase Command Sequence
Embedded Erase algorithm in progress
Dual Poll of Erasing Bank from System
Data = all 0xFF?
No
Yes
Erasing Completed

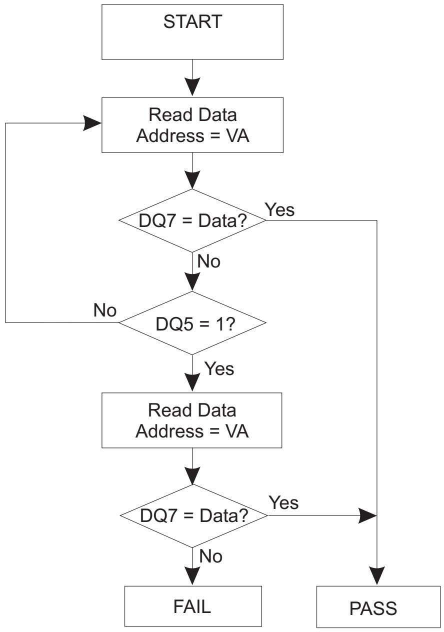
Figure 7-1 : Flash Erase Operation


Write "Write to Buffer" command and Sector Address
Write number of addresses to program minus 1 (AC) and sector address
Write first address/data
Yes
AC = 0?
No
Abort Write to Buffer Operation?
Yes
Write to different sector address
Write to buffer ABORTED. Must write "Write-to-buffer Abort Reset" command sequence to return to read mode
No
Write next address/data pair 1)
AC = AC-1
Write program buffer to flash sector address
Read data at last loaded address
DQ7 = Data?
Yes
No
No
No
DQ1 = 1?
DQ5 = 1?
Yes
Read data at last loaded address
DQ7 = Data?
Yes
No
FAIL or ABORT 2)
PASS

1) When Sector Address ist specified, any address in the selected sector is acceptable. However, when loading Write-Buffer address locations with data, all addresses must fall within the selected Write-Buffer Page.

2) If this flowchart location was reached because of DQ5=1, then the device FAILED. If this flowchart location was reached because of DQ1=1, then the Write-to-Buffer operation was ABORTED. In either case, the proper reset command must be written before the device can begin any next operation. If DQ1=1, write the Write-Buffer-Programming-Abort-Reset command. If DQ5=1, write the Reset command.

Figure 7-2 : Flash Write Buffer Programming



VA = Valid address for programming. During a sector erase operation, a valid address is any sector address within the sector being erased. During chip erase, a valid address is any non-protected sector address.

Figure 7-3 : Flash Data Polling

7.4.1.5 Sector Maps

The two **Spansion S29GL016A Model R1** Flash provides 31 sectors (2 x 64 Kbyte each) plus 8 Top Boot Sectors (2 x 8 Kbyte each):

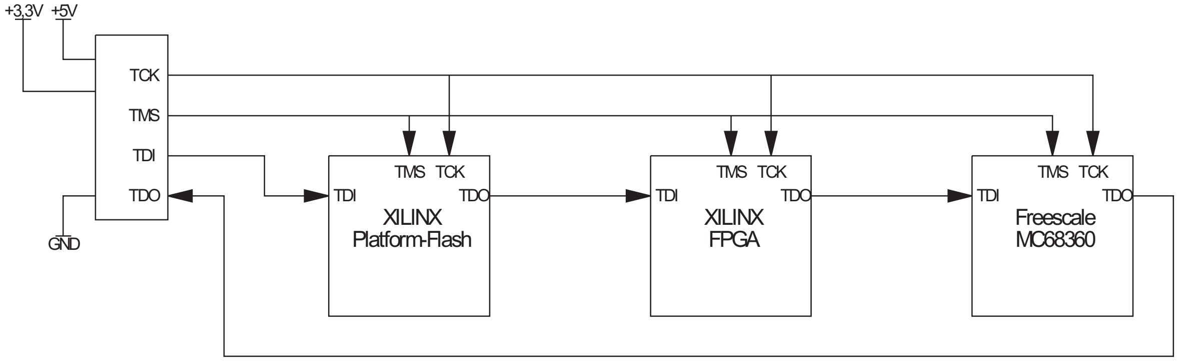
Sector	Sector Size (Byte)	Sector Address Range
SA0	2 x 64K	0x0004_0000 - 0x0005_FFFF
SA1	2 x 64K	0x0006_0000 - 0x0007_FFFF
Sectors below these address could not used on the TIP360		
SA2	2 x 64K	0x0008_0000 - 0x0009_FFFF
SA3	2 x 64k	0x000A_0000 - 0x000B_FFFF
…	…	…
SA28	2 x 64K	0x003C_0000 - 0x003D_FFFF
SA29	2 x 64K	0x003E_0000 - 0x003F_FFFF
SA30	2 x 64K	0x0040_0000 - 0x0041_FFFF
SA31	2 x 8K	0x0042_0000 - 0x0042_3FFF
SA32	2 x 8K	0x0042_4000 - 0x0042_7FFF
SA33	2 x 8K	0x0042_8000 - 0x0042_BFFF
SA34	2 x 8K	0x0042_C000 - 0x0042_FFFF
SA35	2 x 8K	0x0043_0000 - 0x0043_3FFF
SA36	2 x 8K	0x0043_4000 - 0x0043_7FFF
SA37	2 x 8K	0x0043_8000 - 0x0043_BFFF
SA38	2 x 8K	0x0043_C000 - 0x0043_FFFF

Table 7-6 : S29GL016A Model R1 Sector Map

8 JTAG and BDM Port

8.1 JTAG/Boundary Scan

Three devices will be connected to the JTAG chain for boundary scan.



With these three devices it will be possible to test most of TIP360 device and connector interconnections against interruptions and shorts. For this test two TIP360 could be connected to each other with IP Interface connector to IP Interface connector and I/O connector to I/O connector.

8.1.1 Pin Assignment JTAG connector

Pin	Signal
1	+3,3V
2	GND
3	+5V
4	GND
5	TDO
6	TDI
7	TCK
8	TMS

Table 8-1 : JTAG Pin Assignment

8.2 BDM – Port

The TIP360 supports a background debug mode port (BDM port) for debugging the MC68360.

8.2.1 Pin Assignment BDM connector

Pin	Signal
1	DS#
2	BERR#
3	GND
4	BKT#/DSCLK
5	GND
6	FREEZE
7	RESETH#
8	IFETCH/DS1
9	+5V
10	IPIPE0/DSO

Table 8-2 : BDM Pin Assignment

9 Pin Assignment – I/O Connector

All 46 serial I/O lines of the MC68360 (I/O Port A, B, C) are routed directly (TTL level) to the local I/O connector P2 of the IP. I/O port A, B, C contain the I/O lines of the four SCC channels.

For electrical characteristics please refer to the Datasheet of the MC68360.

Pin	Signal	Add. MC68360 Function	Pin	Signal	Add. MC68360 Function
1	PA0	RXD1	26	+5V fused	
2	PB0	PRJCT1 / SPISEL	27	PB11	L1CLKOA / SMRXD2
3	PA1	TXD1	28	PC8	SDACK2 / L1TSYNCB / CTS3
4	PB2	RRJCT2 / SPIMOSI(SPITXD)	29	PB6	DONE1 / SMTXD1
5	PA8	TIN1 / L1RCLKA / BRGO1 / CLK1	30	PA4	L1TXDB / RXD3
6	GND		31	PB7	DONE2 / SMRXD1
7	PA9	BRGCLK1 / TOUT1 / CLK2	32	PA5	L1RXDB / TXD3
8	PC0	L1ST1 / RTS1	33	PC9	L1RSYNCB / CD3
9	PA10	TIN2 / L1TCLKA / BRGO2 / CLK3	34	PB4	DREQ1 / BRGO1
10	PC1	L1ST2 / RTS2	35	PC10	SDACK1 / L1TSYNCA / CTS4
11	PA11	TOUT2 / CLK4	36	PB5	DACK1 / BRGO2
12	PC2	L1ST3 / L1RQB / RTS3	37	PC11	L1RSYNCA / CD4
13	PA12	TIN3 / BRGO3 / CLK5	38	PB15	L1ST4 / L1RQA / RTS4
14	PC3	L1ST4 / L1RQA / RTS4	39	PB16	STRBO / BRGO3
15	PA13	BRGCLK2 / L1RCLKB / TOUT3 / CLK6	40	GND	
16	PC4	CTS1	41	PB17	STRBI / RSTRT1
17	PA2	RXD2	42	PB14	L1ST3 / L1RQB / RTS3
18	PC5	TGATE1 / CD1	43	PA6	L1TXDA / RXD4
19	PA3	TXD2	44	PB13	L1ST2 / RTS2
20	PC6	CTS2	45	PA7	L1RXDA / TXD4
21	PA14	TIN4 / BRGO4 / CLK7	46	PB12	L1ST1 / RTS1
22	GND		47	PB10	L1CLKOB / SMTXD2
23	PA15	L1TCLKB / TOUT4 / CLK8	48	PB9	DACK2 / SMSYN2
24	PC7	TGATE2 / CD2	49	PB8	DREQ2 / SMSYN1
25	PB1	RSTRT2 / SPICLK	50	PB3	BRGO4 / SPIMISO(SPIRXD)

Table 9-1 : Pin Assignment I/O Connector

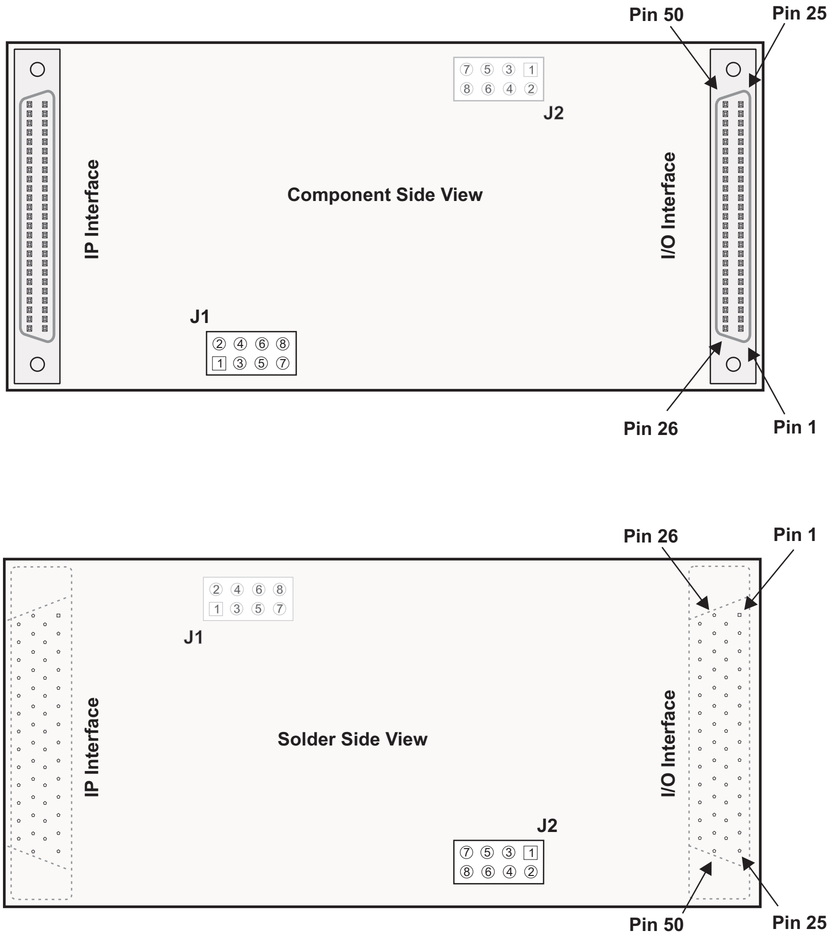


Figure 9-1 : IP Connector Orientation