Greenspring IP-COMM360-25-2

IndustryPack CPU

In Stock

Used and in Excellent Condition

Open Web Page

https://www.artisantg.com/96026-1

IP-COMM360

Quad Integrated Communication Controller (QUICC) IndustryPack®

Manual

Hardware Revision B2
Part # 89002061 Rev. 1.0
20050124

IP-COMM360

Quad Integrated Communication Controller (QUICC) IndustryPack®

SBS Technologies, Inc.
1284 Corporate Center Drive
St. Paul, MN 55121
Tel (651) 905-4700
FAX (651) 905-4701
Email: support.commercial@sbs.com
http://www.sbs.com

Table of Contents

List of Figures

Product Description

The IP-COMM360 is part of the IndustryPack® family of modular I/O components. This IP incorporates the Motorola MC68360 Quad Integrated Communications Controller (QUICC) running at 25 MHz. The 68360 contains a CPU32+ core processor, four independent serial communication controllers, a microcoded RISC communications processor, 14 serial DMAs, two independent DMAs, four general purpose timers, four baud rate generators, two serial management controllers, one serial peripheral interface, time slot assigner, parallel interface port, and support for two TDM channels. Additionally the background debug mode (BDM) is provided to facilitate software development.

A variety of physical layer standards, such as RS-232-D, RS-422, RS-485, ISDN, and Ethernet are supported by the use of transition modules. The architecture makes it easy for customers to implement special physical interface requirements by substituting their own transition module. A transition module is installed between the IP I/O connector and other communication terminals. A number of transition modules are offered by the SBS. Contact factory for the options.

The high density implementation is made possible by the use of TSOP memory package and a Xilinx® field programmable gate array in a fine-pitch SMT package. The Xilinx device design includes the IP interface logic, the IP control registers, and the 68360 control signals.

The IP includes 256 Kbytes of high speed volatile static RAM (SRAM), 256 Kbytes of FLASH memory, and two Mbytes of DRAM for 68360 code and communications data. The boot code may reside in either the SRAM or the FLASH memory. The SRAM, FLASH and DRAM memories, in addition to the 68360 on-chip RAM and registers are all dual ported. Dual port memory can be accessed by both the host CPU and the 68360 masters.

The host CPU must install the boot code into the SRAM or the FLASH prior to releasing the 68360 reset. The reset control and the choice of boot memory is user programmable via IP control registers in the Xilinx device.

A background debug mode (BDM) port is provided for comprehensive software development.

The IP can generate interrupts to the host CPU and the host CPU can generate interrupts to IP.

Key Features

- MC68360 Quad Integrated Communication Controller (QUICC) @ 25 MHz
- Two Mbytes of dual-port high-speed 32-bit DRAM
- 256 Kbytes of dual-port high-speed 16-bit static RAM
- 256 Kbytes of dual-port high-speed 16-bit FLASH memory
- Physical layer implemented with transition modules
- Single-high IndustryPack, Type II (components on back)
- All CMOS surface mount — low power, rugged
- Up to 16 serial channels per VME slot

MC68360 Key Features

- CPU32+ Processor (4.5 MIPs at 25 MHz)
  - 32-Bit Version of the CPU32 Core (fully Compatible with the CPU32)
  - Background Debug Mode
- Four General-Purpose Timers
  - Four 16-bit Timers or Two 32-bit Timers
  - Gate Mode Can Enable/Disable Counting
- Two Independent DMAs
  - Single Address Mode for Fastest Transfers
  - Buffer Chaining and Auto Buffer Modes
  - Automatically Performs Efficient packing
  - 32-bit Internal and External Transfers
- Interrupts
  - Seven External IRQ Lines
  - 12 Port Pins With Interrupt Capability
  - 16 Internal Interrupt Sources
  - Programmable Priority between SCCs
  - Programmable Highest Priority Request
- Communication Processor Module (CPM)
  - RISC Controller
  - 224 buffer descriptors
  - Supports Continuous Mode Transmission and Reception on All Serial Channels
  - 2.5 Kbytes of dual-port RAM
  - 14 Serial DMA (SDMA) Channels
  - Three Parallel I/O Registers with Open-Drain Capability
  - Each Serial Channel Can Have Its Own Pins (NMSI mode)
- Four Baud Rate Generators
  - Independent (Can Be Connected to Any SCC or SMC)
  - Allows Changes During Operation
  - Auto baud Support Option
- Four SCCs
  - Ethernet/IEEE 802.3 Optional on SCC1 (Full 10-Mbps Support)
  - HDLC/SDLC (All Four Channels Supported at 2 Mbps)
  - HDLC Bus (Implements an HDLC-Based Local Area Network (LAN))
  - AppleTalk
  - Signaling System #7
  - Universal Asynchronous Receiver Transmitter (UART)
  - Synchronous UART
  - Binary Synchronous Communication (BISYNC)
  - Totally Transparent (Bit Streams)
  - Totally Transparent (Framed Based with Optional CRC)
  - Profibus (RAM Microcode Option)
  - Asynchronous HDLC (RAM Microcode Option)
  - DCMP (RAM Microcode Option)
  - V.14 (RAM Microcode Option)
  - X.21 (RAM Microcode Option)
- Two SMCs
  - UART
  - Transparent
  - General Circuit Interface (GCI) Controller
  - Can Be Connected to the Time- Division Multiplexed (TDM) Channels
- One SPI
  - Supports Master and Slave Modes
  - Supports Multimaster Operation on the Same Bus

- Time-Slot Assigner
- Supports Two TDM Channels
  - Each TDM Channel Can Be T1, CEPT, PCM Highway, ISDN Basic Rate, ISDN Primary Rate,
    User Defined
  - 1- or 8-Bit Resolution
  - Allows Independent Transmit and Receive Routing, Frame Syncs, Clocking
  - Allows Dynamic Changes
  - Can be Internally Connected to Six Serial Channels (Four SCCs and Two SMCs)
- Parallel Interface Support
  - Centronics Interface Support

Throughout this user manual, references are made to the 68360 features and signals that are specific to its implementation on the IP. The user should consult the MC68360 User's Manual for a thorough understanding of the 68360 features and performance.

The IP-COMM360EN includes the Motorola MC68360EN with Ethernet function on SCC1. SBS offers the XM-360-EN transition module which includes Ethernet 10BaseT connectivity on SCC1 and RS-422 on SCC2, SCC3 and SCC4. See the XM-360-EN user manual for details.

The IP-COMM360MH includes the Motorola MC68360MH with a QUICC multi-channel controller (QMC.) SBS offers the XM-360-T1 transition module which includes T1 connectivity on SCC1 through SCC4. See the XM-360-T1 user manual for details. A transition module for E1 connectivity is planned. Contact the factory for more information.

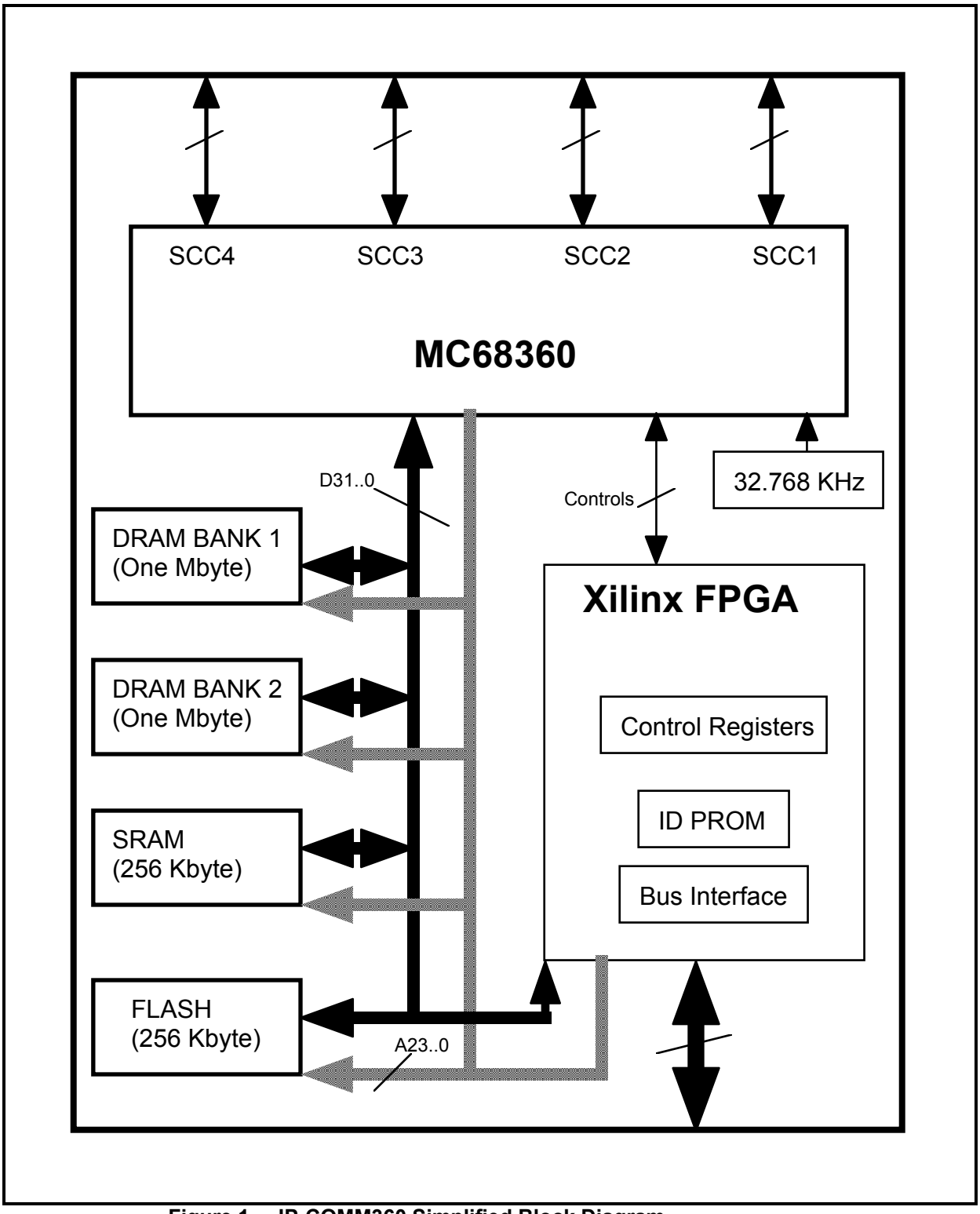

Figure 1 IP-COMM360 Simplified Block Diagram

VMEbus Addressing

The address map of Xilinx based control registers is listed below in figure 2. The registers are accessed on D7..D0 (odd byte).

The SRAM, FLASH, DRAM, and 68360 on-chip RAM and registers can be accessed as bytes or words. The odd byte is on D7..D0, the even byte is on D15..D8, and the word is on D15..D0.

Register Name	68K Address	Access
Reset (RESETH)	$01	Byte, R/W
Boot (BOOT)	$03	Byte, R/W
Interrupt Vector Register (IVR)	$05	Byte, R/W
Interrupt Register (IR)	$07	Byte, R/W
Function Code Register (FCR)	$09	Byte, R/W

Figure 2 VMEbus Address Map

All addresses listed in figure 2 are the offsets from the I/O base address of the host carrier board.

See the Programming section for additional details about the registers listed in figure 2.

Nubus Addressing

NuBus addresses are computed using the following formula and the byte addresses listed in figure 2 in the VMEbus Addressing section. The formula is:

NuBus byte address = (VMEbus byte address * 2) – 1

All byte data is still transferred on data lines D7..D0.

Word addresses on the NuBus are the same as for VME. Word data is transferred on data lines D15..D0.

ISA (PC-AT) Addressing

The byte addresses listed in figure 2 in the VMEbus Addressing section are translated to ISA bus addressing with the formulas below. The ISA bus uses Intel byte ordering as VMEbus uses Motorola byte ordering. The byte ordering of the Intel architecture is the reverse of the Motorola architecture byte ordering.

ISA bus even byte address = VMEbus even address - 1
ISA bus odd byte address = VMEbus odd address +1

Programming

General

This section discusses the 68360 implementation specific to the IP-COMM360. However the user should refer to the MC68360 User's Manual and gain familiarity with the following chapters as a prerequisite to understand the remainder of this section:

- Introduction
- Signal Description
- Memory Map
- Bus Operation
- System Integration Module (SIM60)

Upon power-up or a host CPU reset, the IP-COMM360 holds the MC68360 in its reset state. This reset state is the result of the hard reset (Reseth) pin of 68360 being asserted. During the reset state, both the SRAM and the FLASH memory are ready to be downloaded with boot code. The user should first chose either SRAM or FLASH for boot code and then download the code into that memory. The next step is to program the BOOT bit in the BOOT register to select the memory downloaded with boot code. The host CPU can then program the RESET bit in the RESETH register to release the 68360 reset so the 68360 will begin code execution. See RESETH and BOOT register definition in the Xilinx-Based Register section.

Figure 3 below lists the address map for the SRAM and FLASH within the IP memory space during reset. This map is hardwired and any access outside of this space is ignored by both memories. The state of BOOT register during this time is a don't care. The only memories accessible to the host CPU during reset are SRAM and FLASH. The other resources, including the DRAM and the 68360 on-chip RAM and registers are not accessible to the host CPU when the reset pin is being asserted.

Address Range	Space	Resource
$00 0000 - 03 FFFF	256 Kbyte	SRAM
$08 0000 - 0B FFFF	256 Kbyte	FLASH

Figure 3 IP Memory Address Map

As soon as the 68360 is released from reset, the SRAM and FLASH only respond to the 68360 chip select signals CS0 and CS3. The memory selected as the boot memory defaults to CS0 and the other memory defaults to CS3. In the 68360 architecture, the CS0 is the global chip select after the release of reset and all the memory cycles default to this bank with 15 wait states. The user boot code should program this bank to a 256 Kbyte space with the number of wait states specified in the Specifications section. Likewise the CS3 bank should also be programmed to a 256 Kbyte space with the number or wait states specified in the Specifications section.

The sample initialization code listed on the next page implements the address map shown in Figure 4. Note that all resources are mapped within the first four Mbyte address space of 68360. This mapping scheme makes these resources accessible to the host CPU within the eight Mbyte IP memory space. This code is part of the production test for this IP. This code is left programmed in the FLASH memory to make it easy for users to

check the IP in their system. Users may re-arrange this code to meet their requirement. The code was built with the Single Step tools from Software Development Systems Inc.

68360 Address Range	Space	Resource
$00 0000 - 03 FFFF	256 Kbyte	SRAM (CS0 bank), 16-bit
$04 0000 - 07 FFFF	256 Kbyte	FLASH (CS3 bank), 16-bit
$08 0000 - 0F FFFF	512 Kbyte	not used
$10 0000 - 1F FFFF	1 Mbyte	DRAM bank 1((RAS1), 32-bit
$20 0000 - 2F FFFF	1 Mbyte	DRAM bank 2 (RAS2), 32-bit
$30 0000 - 30 1FFF	8 Kbyte	On-chip RAM & Regs, 32-bit
$30 2000 - 37 FFFF	504 Kbyte	not used
$38 0000 - 3F FFFF	512 Kbyte	Mail-Box (CS4 bank), 8-bit

Figure 4 68360 Memory Address Map

Note: Figure 4 is based on SRAM boot mode. If FLASH boot is selected, then the CS0 bank is FLASH and CS3 bank is SRAM. The IP-COMM360-xx-1 order options do not include SRAM or RAS2 bank DRAM.

```
; Parameters
STKTOP   EQU      $001A3000        ; Stack Pointer value, in DRAM Bank 1
;
; IP-COMM360 Register Map (partial)
MBAR     EQU      $03FF00          ; Module Base Address Register
BASE     EQU      $301000          ; On-chip SIM registers base address
MCR      EQU      BASE+$0          ; Module Configuration Register
AVR      EQU      BASE+$8          ; Autovector Register
RSR      EQU      BASE+$9          ; Reset Status Register
CLKOCR   EQU      BASE+$0C         ; CLKO Control Register
PLLCR    EQU      BASE+$10         ; PLL Control Register
CDVCR    EQU      BASE+$14         ; Clock Divider Control Register
PEPAR    EQU      BASE+$16         ; Port E Pin Assignment Register
SYPCR    EQU      BASE+$22         ; System Protection Control Register
GMR      EQU      BASE+$40         ; Global Memory Register
BR0      EQU      BASE+$50         ; Base Register 0
OR0      EQU      BASE+$54         ; Option Register 0
BR1      EQU      BASE+$60         ; Base Register 1
OR1      EQU      BASE+$64         ; Option Register 1
BR2      EQU      BASE+$70         ; Base Register 2
OR2      EQU      BASE+$74         ; Option Register 2
BR3      EQU      BASE+$80         ; Base Register 3
OR3      EQU      BASE+$84         ; Option Register 3
BR4      EQU      BASE+$90         ; Base Register 4
OR4      EQU      BASE+$94         ; Option Register 4
;
; Initialize the MBAR which is in CPU space, not supervisor data space.
; The MOVES or MOVEC instruction must be used to access the MBAR
;
         SECTION  code
GO       MOVE     #7,D0            ; Load D0 with Function Code "0x0111"
         MOVEC    D0,DFC           ; Load Src and Dest Function Code Regs
         LEA      MBAR,A0          ; A0 points to Module Base Addr Reg
         MOVE.L   #$00300001,D0    ; Load base address & valid flag to D0
         MOVES.L  D0,(A0)          ; MBAR: Base Address set to 0x300000
;
; Register Initialization
         MOVE.L   #$00006CBF,MCR   ; MCR: set for Async Bus Operation
         MOVE.B   #$00,AVR         ; Disable internal Auto Vector
         MOVE.B   #$0C,SYPCR       ; SYPCR: 2 bus fault Mon, 1K clk timer
         MOVE.B   #$0C,CLKOCR      ; CLKOCR: Disable CLKO2, held high,
                                   ; and Enable CLKO1 Full-Strength
         MOVE.W   #$04C0,PEPAR     ; PEPAR: enable BCLRO, WE3-0, CAS3-0
         MOVE.W   #$D2FA,PLLCR     ; Ramp up clock to 25 MHz
         MOVE.L   #$0B800000,GMR   ; GMR: ena DRAM Refresh, CPU sp CS/RAS
         MOVE.L   #$2FFC0002,OR0   ; OR0: 1 WS, 256K bytes, no FC, SRAM16
```

```
        MOVE.L   #$00000291,BR0   ; BR0: CS0 = $000000-$03FFFF, Rd/Wr
        MOVE.L   #$1FF00001,OR1   ; OR1: 0 WS, 1M bytes, no FC, DRAM32
        MOVE.L   #$00100291,BR1   ; BR1: RAS1 = $100000-$1FFFFF, Rd/Wr
        MOVE.L   #$1FF00001,OR2   ; OR2: 0 WS, 1M bytes, no FC, DRAM32
        MOVE.L   #$00200291,BR2   ; BR2: RAS2 = $200000-$2FFFFF, Rd/Wr
        MOVE.L   #$3FFC0002,OR3   ; OR3: 2 WS, 256K bytes, no FC, SRAM16
        MOVE.L   #$00040291,BR3   ; BR3: CS3 = $040000-$07FFFF, Rd/Wr
        MOVE.L   #$2FF80004,OR4   ; OR4: 1 WS, 512K bytes, no FC, SRAM8
        MOVE.L   #$00380291,BR4   ; BR4: CS4 = $380000-$3FFFFF, Rd/Wr
END     BRA      END              ; Wait here forever while Host access IP
;
; RESET VECTOR: to supervisor program space at address 0.
        SECTION  reset
        DC.L     STKTOP           ; initial stack pointer value
        DC.L     GO               ; initial program counter value
;
;---------------------- end of bootcode.s -----------------------


*
*; Some other code examples:
        MOVE.W   $000000,$400000  ; read from CS0 bank -> store in CS3
bank
        MOVE.L   $100000,$200000  ; read from RAS1 bank -> store in RAS2
bank
```

Host CPU Access

While the 68360 is not in its reset state, the host CPU can access the SRAM, FLASH, DRAM banks, and 68360 on-chip RAM and registers by becoming the master of the 68360 bus through arbitration. Arbitration is handled by the hardware and it is transparent to the host. The host CPU simply accesses the resources using address map established by the 68360 code.

The host CPU must drive the IP address bus with the same address map as 68360 for successful access to the on-board resources. Any mismatch results in access to another resource or the IP not acknowledging the cycle. The user address map must locate the IP on an eight Mbyte boundary and map all resources into a total of eight Mbytes to satisfy this requirement.

Figure 5 illustrates how to access the IP resources with a Motorola MVME162 host CPU using the address map described in Figure 4. The MVME162 is configured to allocate eight Mbytes to the memory space of the IP and locate this space on an eight Mbyte boundary with a base address of $C000 0000 within A32 address space of the MVME 162.

MVME162 Address Range	Space	Resource
$C000 0000 - C003 FFFF	256 Kbyte	SRAM (CS0 bank)
$C004 0000 - C007 FFFF	256 Kbyte	FLASH (CS3 bank)
$C008 0000 - C00F FFFF	512 Kbyte	not used
$C010 0000 - C01F FFFF	1 Mbyte	DRAM (RAS1 bank)
$C020 0000 - C02F FFFF	1 Mbyte	DRAM (RAS2 bank)
$C030 0000 - C030 1FFF	8 Kbyte	68360 on-chip RAM & Regs
$C030 2000 - C037 FFFF	504 Kbyte	not used
$C038 0000 - C03F FFFF	512 Kbyte	not used

Figure 5 MVME162 Memory Address Map

Although the DRAM banks and on-chip RAM and registers are 32-bit wide, the host CPU may only access these resources in bytes or words due to 16-bit width IP data path. Therefor the programmer should avoid implementing 32-bit inter-processor pointers in their code, since the host CPU can not update a long-word pointer in a single access.

Any host CPU accesses to the IP resources that results in 68360 bus timeout is not acknowledged by the IP. The most common bus timeouts are:

- unmapped resource within the 68360 address map.
  For example: MVME162 access in the address range C008 0000 - C00F FFFF
- function code compare mismatch
- Read or write privilege violation
- 68360 catastrophic crash

The host CPU triggered bus timeout sets the BERR bit in the RESETH register and any additional accesses to the IP memory space are inhibited. The host CPU must read the BERR bit to remove this inhibition. Any read of the RESETH register clears the BERR bit. The 68360 bus error handler may report any of its bus time outs to the host CPU via mail-box interrupt. The 68360 code should enable its bus timer to monitor the bus for any timeout event, otherwise the host cycle hangs indefinitely if it does not have its own bus timer enabled.

The function code applied to the FC3..0 lines of the 68360 is programmable in the Xilinx-based FCR register. The user must program this register to the appropriate value if the function code match is enabled in the BRx and ORx registers of the 68360. If different banks require different function codes, the host must write this register with the correct value prior to accessing each bank. Alternatively the code may just disable function code compare. However the FC3..0 value must be programmed to $5 or $6 before host can access the on-chip registers, otherwise the 68360 does not acknowledge the access.

When the 68360 executes the code to increase the clock speed to 25 MHz, it enters a black-out period that lasts a maximum of 2500 clocks. The host CPU must avoid access to the IP memory during this period. The host CPU may wait a few milliseconds after releasing reset to the 68360, or have the 68360 code mail-box interrupt the host upon completion of the 68360 initialization.

Host CPU transfer of data to 68360 resources requires eight 68360 clocks (8x40=320 ns) regardless of the host CPU or the carrier board timing. This is the amount of time that the 68360 has turned its bus over to the host CPU. However the total transfer timing of the host CPU and the carrier board is dependent on their manufacturer. The user is advised to measure it directly with a logic analyzer. Simply measure the time between the assertion of the IP signals MEMSEL and ACK. The Motorola MVME162FX intelligent carrier board was used to benchmark this timing for both 8 MHz and 32 MHz IP clock interface. These timings are seven (7x125 = 875 ns) and 16 (16x31.25 = 500 ns) respectively.

The Xilinx FPGA requires 70 milliseconds to initialize the on-board logic after power up or a reset. Any host CPU access to the IP during this period is not acknowledged. It is recommended for the host CPU to read the ID PROM to verify the completion of initialization.

Reset

In the 68360 architecture, the reseth (hard reset) signal may be asserted by the on-chip sources (watchdog, double bus fault monitor, etc.) or the host CPU. The assertion of this signal clears any pending host CPU bus request or interrupts to the 68360. The user may choose to report any occurrence of reset to the host CPU via mail-box interrupt. See Reset Status Register (RSR) of 68360 for details about the causes of reset.

The Resets (soft reset) is not monitored by the IP logic.

Read-Modify-Write Cycle

The read-modify-write cycle is supported from the host CPU to the 68360 resources. The host simply reads and then writes the memory with address bit A22 driven to logic high. Please note that write must follow read cycle to release the 68360 bus. Any violation of this protocol results in logic hang-up and the necessity of resetting the IP. The following instructions illustrates a read-modify-write cycle to the DRAM bank 1 of figure 5:

```
MOVE.B $C0500000,D0
MOVE.B D1,$C0500000
```

The 68360 local initiated read-modify-write cycle is supported by the CPU32+.

The 68360 bus is indivisible during the read-modify-write cycle. This is not a problem with a 68360 initiated cycle due to the speed of local operation. However a host initiated cycle would most likely take considerably more time and may therefore prevent the 68360 from performing a timing critical mission. For example, a host initiated read-modify-write cycle may delay an SCC that needs the bus to transfer its data to a buffer descriptor before it runs into data overrun error. The user may choose to tolerate such errors. The length of read-modify-cycle is very dependent on the particular host CPU and carrier board and can be measured with a logic analyzer. Measure the time between the assertion of IP signal MEMSEL for the read cycle and end of ACK for the write cycle. Some users implement mail-box interrupt as an alternative to the read-modify-cycle.

68360 Hardware Specific Registers

A few 68360 registers must be programmed with specific values to meet the on-board hardware requirements. These registers are defined below. The IP-COMM360 is a single-master QUICC, non-MC68040 companion hardware, and all bits associated with slave and companion mode are don't care (CONFIG2..0 pins are 101 during reset). The sample code above implements all of these requirements.

Module Base Address Register Enable (MBARE):
- This register should be ignored since there aren't any QUICC slaves on-board

Module Configuration Register (MCR):
- BSTM = 0; asynchronous bus timing
- ASTM = 0; asynchronous arbitration
- SHEN1, 0 = 0; Normal operation

Autovector Register (AVR):
- All bits = 0; AVEC is provided externally

CLKO Control Register (CLKOCR):
- COM2 = 11; CLKO2 disabled (driving 1)
- COM1 = 00; CLKO1 enabled, full-strength output buffer

PLL Control Register (PLLCR):
- PREEN = 0; divide-by-128 prescaler is disabled
- MF11-MF0 = $2FA; 25 MHz PLL

Port E Pin Assignment Register (PEPAR):
- SINTOUT = 000 ; default (used only in CPU enable mode)
- CF1MODE = 10 ; BCLRO output function is chosen
- IPIPE1/RAS1DD = 0 ; QUICC in normal mode
- A31-A28/WE0-WE3 = 1 ; WE0-WE3 output functions are selected
- OE/AMUX = 1 ; AMUX output function is selected
- CAS2, CAS3/IACK3, IACK6 = 0; CAS2 and CAS3 output functions are selected
- CAS0, CAS1/IACK1, IACK2 = 0; CAS0 and CAS1 output functions are enabled
- CS7/IACK7 = 0 ; CS7 output function is selected
- AVEC = 0 ; AVEC input function is selected

Global Memory Register (GMR)
- RCNT7-RCNT0 = 0B ; 15.36 µs refresh period for DRAM banks 1 and 2
- RFEN = 1 ; DRAM refresh is enabled
- RCYC1-RCYC0 = 00 ; refresh cycle is 4 clocks long
- DPS1-DPS0 = 00 ; DRAM port size is 32 bits
- WBTQ = 0 ; RAS is negated for 4 phases
- DWQ = 0 ; reads and writes are the same length
- EMWS = 0 ; normal operation
- SYNC = 0 ; asynchronous operation of the memory controller
- PBEE = 0 ; disable internal parity bus error
- GAMX = 0 ; disable internal address multiplexing for all DRAM banks

Base Register (BR)
- PAREN = 0 ; parity checking is disabled
- CSNTQ = 0 ; CS is negated normally
- TRLXQ = 0 ; do not relax timing

Option Register 0 (OR0)
- DSSEL = 0 ; SRAM bank
- SPS1-SPS0 = 01 ; 16-bit port size
- TCYC3-TCYC0 = 2 ; one wait state is SRAM
- TCYC3-TCYC0 = 3 ; two wait state if FLASH

Option Register 1 (OR1)
- DSSEL = 1 ; DRAM bank
- SPS1-SPS0 = 00 ; 32-bit port size
- TCYC3-TCYC0 = 1 ; one wait state

Option Register 2 (OR2)
- DSSEL = 1 ; DRAM bank
- SPS1-SPS0 = 00 ; 32-bit port size
- TCYC3-TCYC0 = 1 ; one wait state

Option Register 3 (OR3)
- DSSEL = 0 ; SRAM bank
- SPS1-SPS0 = 01 ; 16-bit port size
- TCYC3-TCYC0 = 3 ; two wait state if FLASH
- TCYC3-TCYC0 = 2 ; one wait state if SRAM

Option Register 4 (OR4)
- DSSEL = 0 ; SRAM bank
- SPS1-SPS0 = 10 ; 8-bit port size
- TCYC3-TCYC0 = 2 ; one wait state

Clock

The basic clock is provided by an external 32.768 KHz crystal. This clock is used by the on-chip PLL to ramp up the system clock (CLKO1) to 25 or 32 MHz. The choice of 25 or 32 MHz depends on the order options. The frequency tolerance of this crystal is +/- 20 ppm at 25 C and -0.042 ppm/C over temperature range.

Interrupt

The host CPU can interrupt the 68360 and vice versa. The 68360 may interrupt the host to report 68360 bus error, watchdog timer expiration, communication channel receive or transmit buffer full or empty, mail-box, etc. The host CPU may interrupt the 68360 as a mail-box, a common inter-processor communication.

The host CPU must set any bit of the Xilinx-based control register IR to generate an interrupt to the 68360. The user may set only one bit at a time. This bit is automatically cleared by the 68360 interrupt acknowledge cycle. The user should check the IR register bits to verify that the previous interrupt was acknowledged before generating another interrupt. All host CPU interrupts are autovectored.

The 68360 interrupts the host CPU on IRQ0 by driving CS4 signal and address bit A1. Two distinct operations are possible: 1) read/write IR; 2) generate/check interrupt status. Activating CS4 with A1=0 performs the read/write of the IR register. Activating CS4 with A1=1 performs the generate/check interrupt status. The CS4 bank is on data bus bits D31..24 and therefore any CS4 access is byte wide. The following code writes the IR and then reads it:

```
MOVE.B #FC,$380000     ; initialize IVR to $FC
MOVE.B $380000,D0      ; read IVR
```

The following code generates an interrupt to the host CPU and then checks the host interrupt status:

```
MOVE.B #0,$380002      ; generate interrupt, write CS4 with any value
MOVE.B $380002,D0      ; read the interrupt status into D0
```

The interrupt status is checked by reading only the LSB of the data bus (D24) from address $380002. Reading an LSB value of 1, indicates the interrupt has not yet to been acknowledged. The LSB is automatically cleared by the host CPU interrupt acknowledge cycle. The user should check this bit to verify the previous interrupt was acknowledged before generating another interrupt.

Since the IP must provide an interrupt vector to the host CPU, the Xilinx-based dual port register IVR must be initialized to a vector by either the host CPU or the 68360. This register may be read by both processors at the same time, but there is no write arbitration support, and simultaneous writes may corrupt its content. The user should assign only one processor to have write privilege to this register. Typically the 68360 is chosen for this privilege since it may need to use unique vectors to report different interrupt situations to the host.

Xilinx-Based Register Definition

There are five control registers resident inside the Xilinx device that participate in the interface between the host CPU and the 68360. These registers are mapped in the IP I/O space (See Fig. 2). The functions of Xilinx control registers are described below.

Reseth Register (RESETH)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	-	-	-	-	BERR	STATUS	RST

The RESETH register is written by the host CPU to assert and release the Reseth signal to the 68360. This register is used to report the 68360 bus timeout status. The initial value of this register after power-up or system reset is $00.

Bit [0] RST [R/W]
0 = Reseth is asserted from the host CPU
1 = Reseth is released from the host CPU.

Bit [1] STATUS [R]
The 68360 Reseth pin logic level.
0 = host CPU or 68360 is asserting Reseth line.
1 = reseth is not asserted by anyone

Bit [2] BERR [R/W]
0 = Last host CPU access to 68360 resources was acknowledged.
1 = last host CPU access to 68360 resources resulted in bus error.

Bits [7..3] are don't care and any read is undefined.

Boot Register (BOOT)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	-	-	-	-	-	-	BOOT

The BOOT register determines whether the CS0 (boot) bank defaults to SRAM or FLASH. This register should be written prior to releasing reset to the 68360. The initial value of this bit after power-up or system reset is 0.

0 = SRAM is boot memory
1 = FLASH is boot memory

Bits [7..1] are don't care and any read is undefined.

Interrupt Vector Register (IVR)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IV7	IV6	IV5	IV4	IV3	IV2	IV1	IV0

The IVR may be written or read by the host CPU or the 68360. This register provides the interrupt vector to the host CPU during an interrupt acknowledge cycle. The initial value of this register after power-up or system reset is $00.

Bits [7..0] IV7..0 [R/W]
IV7 is the most significant and IV0 is the least significant bit.

Interrupt Register (IR)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	IR7	IR6	IR5	IR4	IR3	IR2	IR1	-

The IR is written by the host CPU to generate an interrupt to the 68360. The entire register is cleared by the hardware during the 68360 interrupt acknowledge cycle or reset. The initial value of this register upon power-up or system reset is $00.

Bit [7..1] IRQ7..1 [R/W]
Setting bit 1 generates an interrupt on level 1. Likewise setting one of bits 2..7 generates an interrupt on level 2..7. Bit 0 is don't care and any read is undefined.

Function Code Register (FCR)

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	-	-	-	-	FC2	FC1	FC0

The FCR provides the function code that is driven onto the 68360 bus during a host CPU access to the 68360 resources. FC3 is not shown here, but is always driven to zero during an access. The initial value of this register upon power-up or system reset is $00.

Bit [2..0] FC2..0 [R/W]

Bits [7..3] are don't care and any read is undefined.

Note: Program this register to either $5 or $6 unless the BR and OR registers for the bank that is being accessed are programmed to enable function code matching.

Erasing and Programming Flash Memory

While the IP-COMM360 is in the reset state with Reseth active, the host CPU can erase and program the Flash Memory on the IP. The base address of the Flash Memory while the IP is reset is listed in Figure 3. The base address is at offset $80000 from the base address of the IP. The following Erasing and Programming procedures use the addresses of an IP-COMM360 installed in IP slot A on a MVME162. The base address for these examples is $C0080000.

Flash Memory bits can be programmed to zero but not to one. The Flash Memory must be erased to make all Flash Memory bits one. When the erase commend sequence listed below is written to the Flash Memory devices, the chips first program all of the memory with zeros and then erase all of the memory. The data in all of the memory locations in the erased Flash Memory will read $FFFF. Erasing the two byte wide Flash Memory devices on an IP-COMM360 requires six word writes to the Flash Memory.

1. Write $AAAA to the reset Flash Memory Base + $AAAA ($C008AAAA)
2. Write $5555 to the reset Flash Memory Base + $5554 ($C0085554)
3. Write $8080 to the reset Flash Memory Base + $AAAA ($C008AAAA)
4. Write $AAAA to the reset Flash Memory Base + $AAAA ($C008AAAA)
5. Write $5555 to the reset Flash Memory Base + $5554 ($C0085554)
6. Write $1010 to the reset Flash Memory Base + $AAAA ($C008AAAA)
7. To determine when the Flash Memory devices have finished erasing, read a Flash Memory location such as $C0080000 until the data reads $FFFF.

Programming each word in the Flash Memory devices on an IP-COMM360 requires four word writes to the Flash Memory.

1. Write $AAAA to the Flash Memory Base + $AAAA ($C008AAAA)
2. Write $5555 to the Flash Memory Base + $5554 ($C0085554)
3. Write $A0A0 to the Flash Memory Base + $AAAA ($C008AAAA)
4. Now write the data you want programmed into the Flash Memory to the address in Flash Memory where you want the data.
5. To determine when the Flash Memory devices have finished programming, read the Flash Memory address being programmed until the data read matches the data given to the Flash Memory to program that address.

For example, use the following procedure to program the Flash Memory with an initial Stack Pointer value of $1A3000 and an initial Program Counter value of $400 to be loaded by the 68360 when the Reseth line is released

1. Write $AAAA to the Flash Memory Base + $AAAA ($C008AAAA)
2. Write $5555 to the Flash Memory Base + $5554 ($C0085554)
3. Write $A0A0 to the Flash Memory Base + $AAAA ($C008AAAA)
4. Write $001A to the Flash Memory Base + $0000 ($C0080000)
5. Now read address $C0080000 until $001A is read.

1. Write $AAAA to the Flash Memory Base + $AAAA ($C008AAAA)
2. Write $5555 to the Flash Memory Base + $5554 ($C0085554)
3. Write $A0A0 to the Flash Memory Base + $AAAA ($C008AAAA)
4. Write $3000 to the Flash Memory Base + $0002 ($C0080002)
5. Now read address $C0080002 until $3000 is read.

1. Write $AAAA to the Flash Memory Base + $AAAA ($C008AAAA)
2. Write $5555 to the Flash Memory Base + $5554 ($C0085554)
3. Write $A0A0 to the Flash Memory Base + $AAAA ($C008AAAA)
4. Write $0000 to the Flash Memory Base + $0004 ($C0080004)
5. Now read address $C0080004 until $0000 is read.

1. Write $AAAA to the Flash Memory Base + $AAAA ($C008AAAA)
2. Write $5555 to the Flash Memory Base + $5554 ($C0085554)
3. Write $A0A0 to the Flash Memory Base + $AAAA ($C008AAAA)
4. Write $0400 to the Flash Memory Base + $0006 ($C0080006)
5. Now read address $C0080006 until $0400 is read.

BDM Port

The 68360 contains a background debug mode (BDM) port. The BDM port is essentially a debugger that is built into the CPU32+ on the 68360. The user can communicate with the BDM through the on-board 10-pin E2 header. This header is also referred to as the Berg connector.

The user should keep the length of their BDM interface cable to less than three feet to provide noise immunity and prevent false Freeze and Reset signals. Also the user should take extreme ESD precaution when installing their BDM cable. This is to avoid possible ESD damage to the on-board logic.

The E2 pin 1 is clearly marked in silk-screen on the back of the IP-COMM360.

The figure 6 below shows the pin assignment of E2 header.

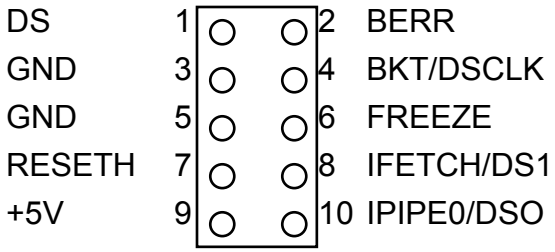

Figure 6 BDM (E2) Pin Assignment

ID PROM

Every IP contains an ID PROM, whose size is at least 32 x 8 bits. The ID PROM aids in software auto-configuration and configuration management. The user's software, or a supplied driver, may verify that the device it expects is actually installed at the location it expects, and is nominally functional. The ID PROM contains the manufacturing revision level of the IP. If a driver requires a particular revision IP, it may check for it directly.

Standard data in the ID PROM on the IP-COMM360 is shown below in Figure 7. For more information on IP ID PROMs refer to the IndustryPack Logic Interface Specification, available from SBS. The ID PROM is implemented in the Xilinx FPGA device.

The location of the ID PROM in the host's address space is dependent on the carrier used. Normally for VMEbus carriers the ID PROM space is directly above the IP's I/O space, or at IP-base + $80. Macintosh drivers use the ID PROM automatically. RM1260 address may be derived from Figure 7 below by multiplying the addresses given by two, then subtracting one. RM1270 addresses may be derived by multiplying the addresses given by two, then adding one.

3F 2D	(available for user)	
17	CRC for bytes used: order option IP-COMM360-25-1 order option IP-COMM360-25-2 order option IP-COMM360-32-1 order option IP-COMM360-32-2	 (E9) (5D) (FA) (26)
15	No of bytes used	(0C)
13	Driver ID, high byte	(00)
11	Driver ID, low byte: order option IP-COMM360-25-1 order option IP-COMM360-25-2 order option IP-COMM360-32-1 order option IP-COMM360-32-2	 (00) (01) (11) (12)
0F	Reserved	(00)
0D	Revision	(A1)
0B	Model No IP-COMM360	(55)
09	Manufacturer ID SBS	(F0)
07	ASCII "C"	(43)
05	ASCII "A"	(41)
03	ASCII "P"	(50)
01	ASCII "I"	(49)

Figure 7 ID PROM Data (hex)

I/O Pin Wiring

This section provides the I/O pin assignment for the IP-COMM360. All the signals from ports A, B, and C are available at the IP 50-pin I/O connector.

The pin numbers given in Figures 8 correspond to numbers on the 50-pin IndustryPack I/O connector, to the wires on a 50-pin flat cable plugged into a standard IP carrier board, and to the terminal screw numbers on the IP-Terminal block. The first column indicates the corresponding 68360 signals.

All the I/O lines from the 68360 pins to the IP I/O connector are buffered by 33 Ω series resistors for added ESD protection. However the user should take extreme ESD precaution when installing their I/O cable. There are no pull-ups or pull-downs on the I/O lines.

If a port pin is selected as a general-purpose I/O pin, it may be accessed through the port data register (PxDAT). Data written to the PxDAT register is stored in an output latch. If a port pin direction is configured as an output, the output latch data is gated onto the port pin. In this case, when PxDAT is read, ideally the contents of the output latch associated with the output port pin are read. However due to the in line 33Ω series resistors, the read of PxDAT for an output pin is unreliable and should be ignored. Other than this anomaly, the general purpose I/O lines operate normally as documented in the Motorola MC68360 User's Manual.

68360 Signal	IP I/O Pin No. (P2)
(PORT A)	
RXD1/PA0	1
TXD1/PA1	3
RXD2/PA2	17
TXD2/PA3	19
L1TXDB/RXD3/PA4	30
L1RXDB/TXD3/PA5	32
L1TXDA/RXD4/PA6	43
L1RXDA/TXD4/PA7	45
TIN1/L1RCLKA/BRGO1/CLK1/PA8	5
BRGCLK1/TOUT1/CLK2/PA9	7
TIN2/L1TCLKA/BRGO2/CLK3/PA10	9
TOUT2/CLK4/PA11	11
TIN3/BRGO3/CLK5/PA12	13
BRGCLK2/L1RCLKB/TOUT3/CLK6/PA13	15
TIN4/BRGO4/CLK7/PA14	21
L1TCLKB/TOUT4/CLK8/PA15	23
(PORT B)	
PRJCT1/SPISEL/PB0	2
RSTRT2/SPICLK/PB1	25
RRJCT2/SPIMOSI(SPITXD)/PB2	4
BRGO4/SPIMISO(SPIRXD)/PB3	50
DREQ1/BRGO1/PB4	34
DACK1/BRGO2/PB5	36
DONE1/SMTXD1/PB6	29
DONE2/SMRXD1/PB7	31
DREQ2/SMSYN1/PB8	49
DACK2/SMSYN2/PB9	48
L1CLKOB/SMTXD2/PB10	47
L1CLKOA/SMRXD2/PB11	27
L1ST1/RTS1/PB12	46
L1ST2/RTS2/PB13	44
L1ST3/L1RQB/RTS3/PB14	42
L1ST4/L1RQA/RTS4/PB15	38
STRBO/BRGO3/PB16	39
STRBI/RSTRT1/PB17	41
(PORT C)	
L1ST1/RTS1/PC0	8
L1ST2/RTS2/PC1	10
L1ST3/L1RQB/RTS3/PC2	12
L1ST4/L1RQA/RTS4/PC3	14
CTS1/PC4	16
TGATE1/CD1/PC5	18
CTS2/PC6	20
TGATE2/CD2/PC7	24
SDACK2/L1TSYNCB/CTS3/PC8	28
L1RSYNCB/CD3/PC9	33
SDACK1/L1TSYNCA/CTS4/PC10	35
L1RSYNCA/CD4/PC11	37
Fused +5V @ 1/2 Amp.	26
GND	6, 22, 40

Figure 8 IP-COMM360 I/O Pin Assignment

IndustryPack Logic Interface Pin Assignment

Figure 9 below gives the pin assignments for the IndustryPack Logic Interface on the IP-COMM360. Pins marked n/c below are defined by the specification, but not used on IP-COMM360. Also see the User Manual for your IP Carrier board for more information.

GND	GND	1	26	
CLK	+5V		2	27
Reset*	R/W*	3	28	
D0	IDSel*		4	29
D1	n/c	5	30	
D2	MEMSel*		6	31
D3	n/c	7	32	
D4	INTSel*		8	33
D5	n/c	9	34	
D6	IOSel*		10	35
D7	n/c	11	36	
D8	A1		12	37
D9	n/c	13	38	
D10	A2		14	39
D11	n/c	15	40	
D12	A3		16	41
D13	IntReq0*	17	42	
D14	A4		18	43
D15	n/c	19	44	
BS0*	A5		20	45
BS1*	n/c	21	46	
n/c	A6		22	47
n/c	Ack*	23	48	
+5V	n/c		24	49
GND	GND	25	50	

Note 1: The no-connect (n/c) signals above are defined by the IndustryPack Logic Interface Specification, but not used by this IP. See the Specification for more information.

Note 2: The layout of the pin numbers in this table corresponds to the physical placement of pins on the IP connector. Thus this table may be used to easily locate the physical pin corresponding to a desired signal. Pin 1 is marked with a square pad on the IndustryPack.

Figure 9 Logic Interface Pin Assignment

Construction and Reliability

IndustryPacks were conceived and engineered for rugged industrial environments. The IP-COMM360 is constructed out of 0.062 inch thick FR4 V0 material. The eight copper layers consist of two signal layers on the top and bottom, and six internal layers. Two internal layers are dedicated to power and ground planes. Four additional layers are used for signal wiring.

The IndustryPack connectors are keyed, shrouded and gold plated on both contacts and receptacles. They are rated at 1 Amp per pin, 200 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable.

The IP is secured to the carrier with four metric M2 stainless steel screws. The heads of the screws are countersunk into the IP. The four screws provide significant protection against shock, vibration, and incomplete insertion. For most applications they are not required.

The IndustryPack provides a low temperature coefficient of 0.89 W/°C for uniform heat. This is based on the temperature coefficient of the base FR4 material of .31 W/m-°C, and taking into account the thickness and area of the IP. This coefficient means that if 0.89 Watts is applied uniformly on the component side, that the temperature difference between the component and the solder side is one degree Celsius.

Repair

Before returning a product for repair, verify as well as possible that the suspected unit is at fault. Then call the Customer Service Department for a RETURN MATERIAL AUTHORIZATION (RMA) number. Carefully package the unit, in the original shipping carton if this is available, and ship prepaid and insured with the RMA number clearly written on the outside of the package. Include a return address and the telephone number of a technical contact. For out-of-warranty repairs, a purchase order for repair charges must accompany the return. SBS will not be responsible for damages due to improper packaging of returned items. For service on SBS products not purchased directly from SBS contact your reseller. Products returned to SBS for repair by other than the original customer will be treated as out-of-warranty.

Specifications

This section gives the technical specification for the IP-COMM360

LSI Chip:	Motorola MC68360 running at 25 MHz
SRAM:	256 Kbyte Dual-Port, 16-Bit @ One Wait State
FLASH:	256 Kbyte Dual-Port, 16-Bit @ Two Wait State, Advanced Micro Devices Am29F010
DRAM:	Two MB Dual-Port, Fast Page Mode, 32-Bit @ One Wait State, no parity
Number of Channels:	Four Serial Communication Controllers (SCC)
I/O Interface:	Serial synchronous and asynchronous. RS-232, RS--422, RS-485, Ethernet, ISDN
Protocols:	Ethernet/IEEE 802.3, ATM, AppleTalk, Signaling #7, ISDN, UART, HDLC, SDLC, BISYNC, DCMP, V.14, X.21
Performance:	See the MC68360 User's Manual
Interrupt:	Host CPU to 68360 on levels 1 Through 7. IP to host CPU on level 0
IndustryPack Specification:	Logic Interface Specification Revision 0.7.1
Dimensions:	Standard Single IndustryPack width and length. 1.8 x 3.9 inches. Type II (TSOP parts on back), maximum component height on the back at 0.0625 inches
Construction:	8 Layer Printed Circuit, Through Hole and Surface Mount Components. Programmable parts are socketed.
Temperature Coefficient:	.89 W/°C for uniform heat across IP
Test conditions	20°C, typical
Power Requirements	+5 VDC, 350 mA typ. +12 VDC, 0 mA typ. –12 VDC, 0 mA typ.
Environmental	Operating temperature: 0° to +70°C Humidity: 5% - 95% non-condensing Storage temperature: –10° to +85°C