Model 3929-Z1B

SCSI Crate Controller

INSTRUCTION MANUAL

May, 1993

Special Option

Model 3929-S002

SCSI Crate Controller

September, 1993

*****Special Option*****

Model 3929-S002

The 3929-S002 is the same as the model 3929-Z1B except that it has been modified to swap bytes for 16-bit transfers when the CAMAC word size bits are set to one. This 3929 must only function in the high-byte first mode.

SCK:rem
June 3, 1993

*****Special Option*****

Model 3929-S003

SCSI Crate Controller

June, 1997

*****Special Option*****

Model 3929-S003

The 3929-S003 is the same as the model 3929-Z1B except that it has been modified with 2.5 version of firmware instead of 2.3. (It isHP compatible)

CONTENTS

TABLES

FIGURES

SCSI Crate Controller

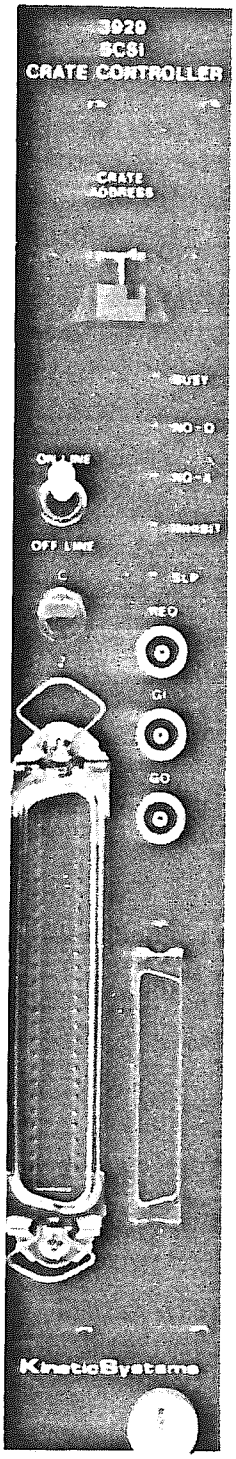
Interfaces a SCSI channel to a CAMAC crate **3929**

Features

- **Interfaces to the Small Computer Systems Interface (SCSI) channel**
- **Complies with the ANSI SCSI standard (X3.131-1986)**
- **Shares a SCSI bus with disk drives and other devices**
- **Multiple crate controllers addressed on a single SCSI bus**
- **Field-selectable as a main or auxiliary crate controller**
- **Supports a SCSI bus length to six meters (20 feet)**
- **Bus length increased by using third-party bus extenders**
- **Hardware list processor option**

Typical Applications

- **Real-time I/O interface for a personal computer or workstation**
- **General-purpose data acquisition and control**
- **Laboratory automation**
- **Industrial process control**



General Description

The 3929 is a double-width CAMAC crate controller that communicates with a host computer via the Small Computer Systems Interface (SCSI) bus and complies with ANSI Specification X3.131-1986. This SCSI specification defines a standard approach for interfacing disk drives, printers, and other devices to one or more host computers. The SCSI bus is one of the more popular peripheral channels for personal computers and workstations.

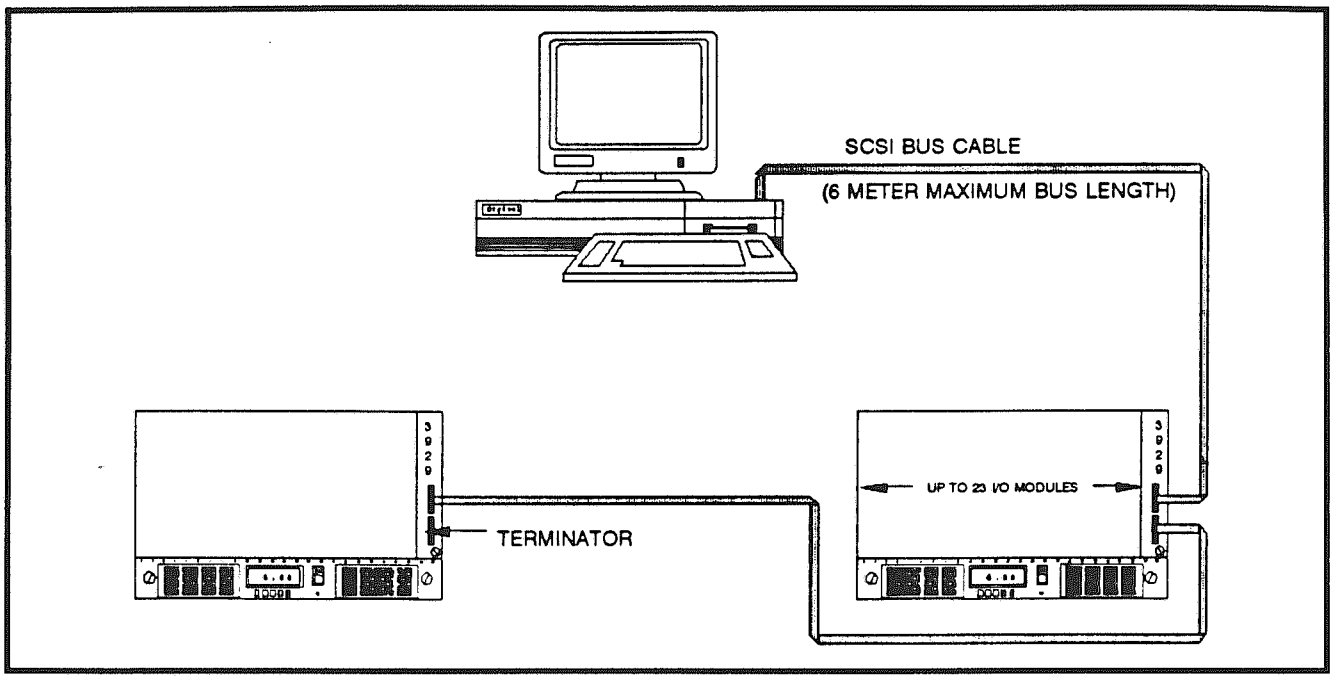
Peripheral controllers are called targets on a SCSI channel. Up to eight target controllers can be addressed on a single SCSI bus. In a typical personal computer or workstation, one or more floppy disk drives form one SCSI target controller, and the hard disk provides a second controller. This allows six additional controllers, such as the 3929, on the SCSI bus. The SCSI controller address is selected by a thumbwheel switch on the front panel of the 3929 crate controller.

The SCSI bus, as used by the 3929, contains an eight-bit data path with parity, nine control signals, terminator power, and associated ground return conductors. All signals are unbalanced, and the maximum total bus length is six meters (20 feet). Third-party bus extenders are available to provide balanced-line or fiber optic extensions to the SCSI bus. Two connectors are provided on the front panel of the 3929 to "pass through" the SCSI bus. On-board termination is provided on the 3929.

The 3929-Z1B option includes a hardware list processor. The list processor executes CAMAC commands contained in a 64 kilobyte storage memory. These commands may be either single CAMAC operation or Block Transfer operations. The list processor allows multiple CAMAC operation to occur within the crate by executing a single SCSI command.

This module is configured at the factory as a main crate controller. It can be changed to an auxiliary crate controller in the field by removing the plug-in resistor packs used to terminate the Dataway. As an auxiliary, the 3929 conforms to the front-panel Request/Grant arbitration protocol specified in IEEE Standard 675-1982.

Typical Configuration



Cable Information

The SCSI specification defines A-Cable and B-Cable transmission schemes. The A-Cable scheme can utilize either a 50-contact low-density connector or a 50-contact high-density connector. The B-Cable scheme uses a 68-contact high-density connector. The 3929 utilizes the A-Cable scheme and includes both the low-density and high-density 50-contact connectors. Several cables are available for interconnecting one or more 3929's and the host computer system. The Model 5809-Cxyz has a 50-contact low-density mating connector at one end of the cable assembly and a 50-contact high-density connector at the other. The Model 5853-Cxyz cable assembly contains a 50-contact low-density connector at both ends. For computers with only the B Cable connector available, one should order the Model 5809-A20J (two meter length) or 5809-A30J (three meter length) cables. These assemblies have a 68-contact high-density connector at one end and a 50-contact low-density connector at the other end. An additional cable assembly—the 5809-Bxyz—has 50-contact high-density connectors at both ends.

Ordering Information

Model 3929-Z1A Crate Controller, for use with SCSI bus

Model 3929-Z1B Crate Controller with list processing, for use with SCSI bus

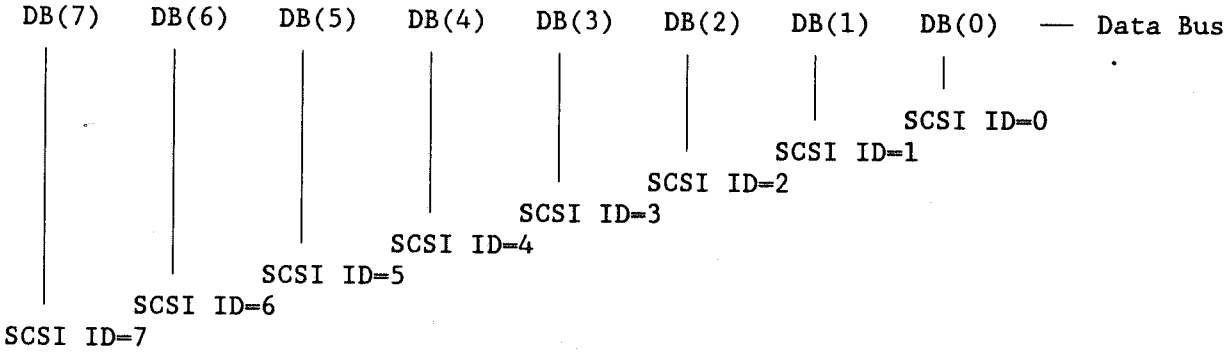
Related Products

Model 5809-Axyz	SCSI Interface Cable, 68P HD to 50P Amp
Model 5809-Bxyz	SCSI Interface Cable, 50P HD Connectors
Model 5809-Cxyz	SCSI Expansion Cable, 50P HD to 50P AMP
Model 5853-Cxyz	50-contact Blue Ribbon Cable Assembly

Note: These cable assemblies are available in lengths of one, two, or three meters.

SCSI Bus

The following is a description of the SCSI specification. Some of the specification details have been omitted as they do not apply to the 3929. Communication on the SCSI bus is allowed between only two SCSI devices at any one given time. A maximum of eight SCSI devices may be present on the SCSI bus. Each SCSI device has an individual SCSI ID bit assigned as shown below.



When two SCSI devices communicate on the SCSI bus, one device acts as the initiator and the other as the target. The initiator invokes an operation and the target performs the operation. A SCSI device usually has a fixed role as an initiator or target. The 3929 is a target device, but can become a temporary initiator. The temporary initiator role is used for sending an Asynchronous Event Notification (AEN) to an initiator. This is used by the 3929 to inform the initiator that a Look-At-Me (LAM) is pending.

Certain SCSI bus functions are assigned to the initiator and also to the target. The initiator arbitrates for the SCSI bus and selects a target. The target may then request the transfer of Command, Status, Data, or other information on the SCSI bus. In some cases, the target may arbitrate for the SCSI bus and reselect an initiator for the purpose of continuing an operation. Information transfers on the SCSI data bus are asynchronous and follow a Request/Acknowledge handshake protocol. One byte of information is transferred per Req/Ack cycle.

SCSI Bus Signals

There are a total of eighteen signals that the 3929 uses to communicate with an initiator. Nine of these signals are control lines and the other nine are for data. Eight of the data signals are for the eight data bits and the ninth is for data parity. The allocation of these signals on the front panel connectors of the 3929 is found in APPENDIX A. The following is a brief description of the SCSI bus signals:

BSY (BUSY)	Busy is an "OR-tied" signal that is asserted to indicate that the SCSI bus is being used.
SEL (SELECT)	Select is an "OR-tied" signal asserted by the initiator to select a target or for a target to select an initiator.

C/D (CONTROL/DATA)	Control/Data is a signal driven by a target to indicate whether control or data information is on the SCSI data bus. When this signal line is asserted, control information is present on the SCSI data bus.
I/O (INPUT/OUTPUT)	Input/Output is a signal driven by a target to control the direction of data transfer on the SCSI data bus with respect to the initiator. This signal is asserted to indicate data transfer from the target to the initiator. This signal has a second purpose which is to distinguish between SELECTION and RESELECTION phases.
MSG (MESSAGE)	Message is a signal asserted by the target during a message phase.
REQ (REQUEST)	Request is a signal asserted by a target to indicate a request for a Req/Ack data transfer handshake on the SCSI bus.
ACK (ACKNOWLEDGE)	Acknowledge is a signal asserted by an initiator to indicate the acknowledgement for a Req/Ack data transfer handshake.
ATN (ATTENTION)	Attention is a signal asserted by an initiator to indicate the Attention condition. The Attention condition is used to inform a target that the initiator has a Message pending.
RST (RESET)	Reset is an "OR-tied" signal which, when asserted, indicates the Reset condition.
DB(7-0,P) (DATA BUS 7-0,PARITY)	DB(7) through DB(0) are the eight data bits on the SCSI bus and DB(P) is the parity signal. DB(7) is the most significant bit and has the highest priority during an ARBITRATION phase for the SCSI bus. The bit number, significance and priority, decreases downward to DB(0). A data bit is defined as one when the signal line is asserted, and as zero when the signal line is negated. DB(P) is generated to maintain odd parity on the SCSI bus.

SCSI Signal Sources

The following table shows which type of SCSI device is allowed to assert each signal. This chart does not show if the source is driving the signal asserted, driving negated, or is passive.

All SCSI device drivers that are not active sources are in the passive state. Note that the RESET signal may be sourced by any device at any time.

TABLE 1 - SCSI Signal Source

	SIGNALS				
Bus Phase	BSY	SEL	C/D,I/O MSG,REQ	ACK/ATN	DB(7-0,P)
BUS FREE	NONE	NONE	NONE	NONE	NONE
ARBITRATION	ALL	WINNER	NONE	NONE	SCSI ID
SELECTION	I & T	INITIATOR	NONE	INITIATOR	TARGET
RESELECTION	I & T	TARGET	TARGET	INITIATOR	TARGET
COMMAND	TARGET	NONE	TARGET	INITIATOR	INITIATOR
DATA IN	TARGET	NONE	TARGET	INITIATOR	TARGET
DATA OUT	TARGET	NONE	TARGET	INITIATOR	INITIATOR
STATUS	TARGET	NONE	TARGET	INITIATOR	TARGET
MESSAGE IN	TARGET	NONE	TARGET	INITIATOR	TARGET
MESSAGE OUT	TARGET	NONE	TARGET	INITIATOR	INITIATOR

ALL: The signal is driven by all SCSI devices that are actively arbitrating.

SCSI ID: A unique data bit is driven by each SCSI device that is actively arbitrating. The other seven data bits are not driven by the arbitrating device.

I & T: The signal is driven by the initiator, target, or both as specified in the SELECTION or RESELECTION phase.

INITIATOR: If this signal is to be driven, it is only driven by an active initiator.

TARGET: If the signal is to be driven, it is only driven by an active target.

WINNER: The signal is driven by the one SCSI device that wins the arbitration.

NONE: The signal is NOT driven by any SCSI device.

SCSI Bus Timing

The following describes some basic timing requirements that are used in the discussion of the SCSI bus phases:

Arbitration Delay (2.2 microseconds).

> This is the minimum time a SCSI device waits from asserting BSY for arbitration until the data bus can be examined to see if arbitration has been won. There is no maximum time specified for this delay.

Assertion Period (90 nanoseconds).

> This is the minimum time a target asserts REQ while using synchronous data transfers. This is also the minimum time that an initiator asserts ACK while using synchronous data transfers.

Bus Clear Delay (800 nanoseconds).

> This is the maximum time for any SCSI device to stop driving all bus signals after:
>
> 1. The BUS FREE phase is detected (indicated by BSY and SEL false for a bus settle delay.
>
> 2. SEL is received from another SCSI device during an ARBITRATION phase.
>
> 3. The transition of RST to asserted.

Bus Free Delay (800 nanoseconds).

> This is the minimum time that a SCSI device waits from detecting the BUS FREE phase until its assertion of BSY when going to the ARBITRATION phase.

Bus Set Delay (1.8 microseconds).

> This is the maximum time for a SCSI device to assert BSY and its SCSI ID on the data bus after it detects a BUS FREE phase for the purpose of entering the ARBITRATION phase.

Bus Settle Delay (400 nanoseconds).

> This is the time to wait for the bus to settle after changing certain control signals as called out in the protocol definitions.

Model 3929-Z1B

Cable Skew Delay (10 nanoseconds).

The maximum difference in propagation time between any two SCSI bus signals when measured between any two SCSI devices.

Data Release Delay (400 nanoseconds).

This is the maximum time for an initiator to release the data bus following the transition of the I/O signal from false to true.

Deskew Delay (45 nanoseconds).

This is the minimum time required for deskew of certain signals.

Hold Time (45 nanoseconds).

This is the minimum time added between the assertion of REQ or ACK and the changing of the data lines to provide hold time in the initiator or target, respectively, using synchronous data transfers.

Negation Period (90 nanoseconds).

This is the minimum time a target negates REQ while using synchronous data transfers. This is also the minimum time an initiator may negate ACK when using synchronous data transfers.

Reset Hold Time (25 microseconds).

This time is the minimum time for which RST is asserted. There is no maximum time specified.

Selection Abort Time (200 microseconds).

This is the maximum time a target (or initiator) may take from its most recent detection of being selected (or reselected) until asserting BSY. This timeout is required to ensure a target (or initiator) does not assert BSY after a SELECTION (or RESELECTION) phase has been aborted.

Selection Timeout Delay (250 milliseconds).

This is the minimum time that an initiator (or target) should wait for a BSY response during a SELECTION (or RESELECTION) phase before starting the timeout procedure.

SCSI Bus Phases

The SCSI protocol describes eight phases as follows:

BUS FREE phase
ARBITRATION phase
SELECTION phase
RESELECTION phase
COMMAND phase
DATA phase
STATUS phase
MESSAGE phase

These phases are collectively termed information transfer phases.

The SCSI bus can never be in more than one phase at any given time. In the following bus phase descriptions, signals that are not mentioned are not asserted.

BUS FREE Phase

The BUS FREE phase indicates that there is no SCSI device currently using the SCSI bus and that it is available for use. Targets may revert to the BUS FREE phase to indicate an error condition that it does not know how to recover from.

SCSI devices detect the BUS FREE phase when SEL and BSY are both false for at least a bus settle delay.

SCSI devices release all SCSI bus signals within a bus clear delay after SEL and BSY are continuously false for a bus settle delay. If a SCSI device requires more than a bus settle delay to detect a BUS FREE phase, it will release all SCSI bus signals within a bus clear delay minus the excess time needed to detect a BUS FREE phase. The total time to clear the SCSI bus does not exceed a bus settle delay plus a bus clear delay.

ARBITRATION Phase

The ARBITRATION phase allows one SCSI device to obtain control of the SCSI bus so that it may assume the role of a target or initiator. The procedure for a SCSI device to obtain control of the SCSI bus is as follows:

1. The SCSI device must first wait for the BUS FREE phase to occur. The BUS FREE phase is detected when both BSY and SEL are continually and simultaneously false for a minimum of a bus settle delay.

2. The SCSI device must wait a minimum of a bus free delay after the detection of a BUS FREE phase before driving any signal.

3. Following the bus free delay as in step 2, the SCSI device may arbitrate for the SCSI bus by asserting both BSY and its own SCSI ID, however, the SCSI device cannot arbitrate if more than a bus set delay has passed since the last BUS FREE phase was last detected.

4. After waiting at least an arbitration delay, which is measured from its assertion of BSY, the SCSI device examines the data bus. If a higher priority SCSI ID bit is asserted on the data bus (DB(7) has the highest priority), then the SCSI device has lost the arbitration and then removes its asserted signals and returns to step 1. If no other higher priority SCSI ID bit is asserted on the SCSI data bus, then the SCSI device has won the arbitration and asserts the SEL signal. Any other SCSI device that was participating in the arbitration has lost and removes its assertion of BSY and its SCSI ID bit within a bus clear delay after SEL is asserted. A SCSI device that loses the arbitration may return to step 1.

5. The SCSI device that wins arbitration, waits for at least a bus clear delay plus a bus settle delay after asserting SEL before changing any signal states on the SCSI bus.

   Note: The SCSI ID bit is a single bit on the data bus that corresponds to the SCSI device's unique SCSI address. Parity, DB(P), is not valid during this phase, but cannot be driven to the false state.

SELECTION Phase

The SELECTION phase allows an initiator to select a target for the purpose of initiating a target function. During the SELECTION phase, the I/O signal is negated to differentiate it from the RESELECTION phase.

The SCSI device that won the arbitration has both BSY and SEL asserted and has delayed at least a bus clear delay plus a bus settle delay before terminating the ARBITRATION phase. The SCSI device that won the arbitration becomes the initiator by releasing the I/O signal. The initiator then sets the data bus to a value that is the "OR" of its SCSI ID bit and the targets SCSI ID bit. The initiator waits for at least two deskew delays and releases BSY. The initiator then waits for at least a bus settle delay before looking from a response from the target.

The target determines it has been selected when the SEL signal is asserted and its SCSI ID bit is true and the BSY and I/O signals are false for at least a bus settle delay. The selected target examines the data bus in order to determine the SCSI ID of the selecting initiator. The selected target then asserts the BSY signal within a selection abort time of its most recent detection of being selected. This is required for proper operation of the selection timeout procedure.

At least two deskew delays after the initiator detects BSY is asserted, it releases SEL and then may change the data bus.

SELECTION Timeout Procedure

There are two optional selection timeout procedures specified for clearing the SCSI bus if the initiator waits a minimum of a selection timeout delay and there has been no BSY response from the target.

1. Optionally, the initiator may assert the RST signal.

2. Optionally, the initiator continues to assert SEL and releases the data bus. If the initiator has not detected BSY to be asserted after at least a selection abort time plus two deskew delays, the initiator releases SEL allowing the SCSI bus to go to the BUS FREE phase. SCSI devices must ensure that when responding to selection that a selection was still valid within a selection abort time of their assertion of BSY. Failure to comply with this requirement could result in an improper selection.

RESELECTION Phase

RESELECTION is an optional phase permitting a target to reconnect to an initiator for the purpose of continuing an operation started by the initiator but was suspended by the target. This occurs when the target disconnects by allowing a BUS FREE phase to occur before the operation was completed. RESELECTION is only used in systems that have the ARBITRATION phase implemented.

After completing the ARBITRATION phase, the winning SCSI device has both the BSY and SEL signals asserted and has delayed at least a bus clear delay plus a bus settle delay. The winning SCSI device becomes the initiator by asserting the I/O signal. The winning SCSI device also sets the data bus to a value that is the "OR" of its SCSI ID and the initiators SCSI ID. The target waits for at least two deskew delays and then releases BSY. The target waits for at least a bus settle delay before looking for a response from the initiator.

The initiator determines if it is reselected when SEL, I/O, and its SCSI ID bit are asserted and BSY is false for at least a bus settle delay. The reselected initiator then examines the data bus to determine the SCSI ID of the reselecting target. The reselected initiator asserts the BSY signal within a selection abort time of its most recent detection of being reselected. This is required for the correct operation of the timeout procedure.

After the target detects BSY, it also asserts BSY and waits at least two deskew delays and then releases SEL. The target may then change the I/O signal and the data bus signals. After the reselected initiator detects SEL is negated, it releases BSY. The target continues to assert BSY until it is ready to relinquish the SCSI bus.

RESELECTION Timeout Procedure

Two optional RESELECTION timeout procedures are specified for clearing the SCSI bus during a RESELECTION phase. If the target waits a minimum of a selection timeout delay and there has been no BSY response from the initiator:

1. Optionally, the target may assert the SCSI RST signal.

2. Optionally, the target may continue asserting SEL and I/O and release the data bus signals. If the target has not detected BSY to be true for at least a selection abort time plus two deskew delays, the target releases SEL and I/O allowing the SCSI bus to go to the BUS FREE phase.

Information Transfer Phases

The COMMAND, DATA, STATUS, and MESSAGE phases are grouped together as information transfer phases. In each of these phases, data is transferred over the SCSI data bus. The actual data transferred depends on the type of operation being performed on the SCSI bus.

The C/D, I/O, and MSG signals are used to distinguish between the different information transfer phases. The target drives these three signals and thereby controls all the changes from one phase to the next. The initiator may request a MESSAGE OUT phase by asserting the ATN signal. The target may cause a BUS FREE phase by releasing the MSG, C/D, I/O and BSY signals.

The following table shows the information transfer phases based upon the MSG, C/D, and I/O signals:

TABLE 2 - SCSI Transfer Phases

SIGNAL					
MSG	C/D	I/O	PHASE	DIRECTION OF TRANSFER	COMMENT
0	0	0	DATA OUT	INITIATOR TO TARGET	DATA PHASE
0	0	1	DATA IN	INITIATOR FROM TARGET	
0	1	0	COMMAND	INITIATOR TO TARGET	
0	1	1	STATUS	INITIATOR FROM TARGET	
1	0	0	RSVD		
1	0	1	RSVD		
1	1	0	MESSAGE OUT	INITIATOR TO TARGET	MESSAGE PHASE
1	1	1	MESSAGE IN	INITIATOR FROM TARGET	

0 = FALSE (NEGATED)
1 = TRUE (ASSERTED)
RSVD = RESERVED FOR FUTURE STANDARDIZATION

The information transfer phases uses one or more REQ/ACK handshakes to control the information transfers. Each REQ/ACK handshake allows the transfer of one information byte. BSY remains asserted and SEL remains negated during the information transfer phases. Also, during the information transfer phases, the target continuously envelops the REQ/ACK handshake(s) with C/D, I/O and MSG. These control signals are valid for a bus settle delay before the assertion of REQ of the first handshake and remain valid until the negation of ACK at the end of the last handshake.

Asynchronous Information Transfers

The target controls the direction of information transfers via the I/O signal. When I/O is asserted, information is transferred from the target to the initiator. When I/O is negated, information is transferred from the initiator to the target.

If I/O is asserted (transfer to the initiator), the target first drives the DB(7-0,P) to their desired values. The target then delays one deskew delay plus a cable skew delay and then asserts REQ. DB(7-0,P) remains valid until ACK is true at the target. The initiator reads DB(7-0,P) after REQ is asserted. The initiator then signals its acceptance of the data by asserting ACK. When ACK becomes true at the target, the target may change or release DB(7-0,P) and negate ACK. After ACK is false, the target may continue the transfer by driving DB(7-0,P) and asserting REQ as above.

If I/O is false (transfer to the target), the target requests information by asserting REQ. The initiator drives DB(7-0,P) to the desired values. It then waits for one deskew delay plus a cable skew delay and asserts ACK. The initiator continues to drive DB(7-0,P) until REQ is false. When ACK becomes true at the target, the target reads DB(7-0,P) and negates REQ. When REQ becomes false at the initiator, the initiator may change or release DB(7-0,P) and negate ACK. The target may continue the transfers by asserting REQ as described above.

Synchronous Information Transfers

Synchronous data transfer is optional and is only used during data phases, DATA IN or DATA OUT. This mode is used in a data phase if a synchronous data transfer agreement has been previously established. Refer to the SYNCHRONOUS DATA TRANSFER REQUEST message for further details.

A REQ/ACK offset is used to specify the maximum number of REQ pulses that can be sent by the target in advance of the number of ACK pulses received from the initiator, which establishes a pacing mechanism. If the number of REQ pulses exceeds the number of ACK pulses by the REQ/ACK offset, the target will not assert REQ signal until after the leading edge of the next ACK pulse is received. For successful completion of the data phase, the number of REQ and ACK pulses must be equal.

The target asserts the REQ signal for a minimum of an assertion period. The target then waits at least the greater of a transfer period from the last transition of the REQ signal to true or a minimum of a negation period from the last transition of the REQ signal to false before re-asserting the REQ signal.

The initiator sends one pulse on the ACK line for each REQ pulse received. The ACK signal may be asserted as soon as the leading edge of the corresponding REQ pulse has been received. The initiator asserts the ACK signal for a minimum of an assertion period. The initiator waits at least the greater of a transfer period from the last transition of the ACK signal to true or for a minimum of a negation period from the last transition of ACK signal to false before re-asserting the ACK signal.

If the I/O signal is true (transfer to the initiator), the target first drives the DB(7-0,P) signals to the desired states. It waits at least one deskew delay plus one cable skew delay and then asserts the REQ signal. The DB(7-0,P) are held valid for a minimum of one deskew delay plus one cable skew delay plus one hold time after the assertion of the REQ signal. The target asserts the REQ signal for a minimum of an assertion period. The target may then negate the REQ signal and change or release the DB(7-0,P) signals. The initiator then reads the data from the DB(7-0,P) lines within one hold time of the transition of the REQ signal to true. The initiator then responds with an ACK pulse.

If the I/O signal is false (transfer to the target), the initiator transfers one byte for every REQ pulse received. After receiving the leading edge of a REQ pulse, the initiator drives the DB(7-0,P) lines to the desired values. It then waits at least one deskew delay plus one cable skew delay and then asserts ACK. The initiator holds the DB(7-0,P) lines valid for a least one deskew delay plus one cable skew delay plus one hold time after the assertion of the ACK signal. The initiator asserts the ACK signal for a minimum of an assertion period. The initiator may then negate the ACK signal and may change or release the DB(7-0,P) signals. The target then reads the data on the DB(7-0,P) lines within one hold time of the transition of the ACK signal to true.

COMMAND Phase

The COMMAND phase allows a target to request command information from an initiator. The target asserts the C/D signal and negates the I/O and MSG signals during the REQ/ACK handshake(s) for this phase.

Data Phase

> The Data phase encompasses both the DATA IN phase and the DATA OUT phase.

DATA IN Phase

> The DATA IN phase allows the target to request that data be sent to the initiator from the target.

DATA OUT Phase

> The DATA OUT phase allows the target to request that data be sent from the initiator to the target.

STATUS Phase

> The STATUS phase allows the target to request that status information be sent from the target to the initiator.

The target asserts the C/D and I/O signals and negates the MSG signal during the REQ/ACK handshake of this phase.

Message Phase

The message phase is a term that refers to either a MESSAGE IN or a MESSAGE OUT phase. Multiple messages may be sent during either phase. The first byte transferred in either of these phases is either a single-byte message or the first byte of a multi-byte message. Multi-byte messages are wholly contained within a single message phase.

MESSAGE IN Phase

The MESSAGE IN phase allows the target to request that a message(s) be sent to the initiator from the target.

The target asserts the C/D, I/O and MSG signals during the REQ/ACK handshake(s) of this phase.

MESSAGE OUT Phase

The MESSAGE OUT phase allows a target to request that message(s) be sent from the initiator to the target. The target may invoke this phase at any time in response to the ATTENTION condition generated by the initiator.

The target asserts the C/D and MSG signal and negates the I/O signal during the REQ/ACK handshake(s) of this phase.

The target handshake(s) bytes in this phase until ATTENTION is negated by the initiator.

Signal Restriction Between Phases

When the SCSI bus is between two information transfer phases, the following restrictions apply to the SCSI bus signals.

1. The BSY, SEL, REQ, and ACK signals must not change.

2. The C/D, I/O, MSG, and data bus signals may change. When switching the data bus direction from out (initiator driving) to in (target driving), the target must delay driving the data bus by at least a data release time plus a bus settle delay after asserting the I/O signal and the initiator must release the data bus no later than a data release delay after the transition of the I/O signal to true.

   When switching the data bus direction from in (target driving) to out (initiator driving), the target must release the data bus no later than a deskew delay after negating the I/O signal.

3. The ATN and RST signals may change as defined under the descriptions for the ATTENTION and RESET conditions.

SCSI Bus Conditions

The SCSI bus has two asynchronous conditions: the ATTENTION condition and the RESET condition. These cause the SCSI device to perform certain actions and thus alter the phase sequence.

ATTENTION Condition

The ATTENTION condition allows an initiator to inform a target that the initiator has a message ready. The target may retrieve this message at its convenience by executing a MESSAGE OUT phase.

The initiator creates the ATTENTION condition by asserting the ATN signal at any time except during the ARBITRATION or BUS FREE phases.

A target responds with a MESSAGE OUT phase as follows:

1. If the ATN signal becomes true during a COMMAND phase, the target enters the MESSAGE OUT phase after transferring part or all of the command descriptor block bytes.

2. If the ATN signal becomes true during a DATA phase, the target enters the MESSAGE OUT phase at the targets earliest convenience. The initiator continues the REQ/ACK handshake(s) until it detects the phase change.

3. If the ATN signal becomes true during a STATUS phase, the target will enter the MESSAGE OUT phase after the status byte has been acknowledged by the initiator.

4. If the ATN signal becomes true during a MESSAGE IN phase, the target enters the MESSAGE OUT phase before it sends another message. This permits a MESSAGE PARITY ERROR message from the initiator to be associated with the appropriate message.

5. If the ATN signal becomes true during a SELECTION phase and before the initiator releases the BSY signal, the target enters the MESSAGE OUT phase immediately after that SELECTION phase.

6. If the ATN signal is becomes true during a RESELECTION phase, the target enters the MESSAGE OUT phase after the target has sent its IDENTIFY message for that RESELECTION phase.

The initiator keeps the ATN signal asserted if more than one message byte is to be transferred. The initiator may negate ATN at any time except while the ACK signal is asserted during a MESSAGE OUT phase. Normally, the initiator negates the ATN signal while the REQ is true and the ACK is false during the last REQ/ACK handshake of the MESSAGE OUT phase.

Reset Condition

The reset condition is used to immediately clear all SCSI devices from the bus. This condition takes precedence over all other phases and conditions. Any SCSI device may generate the reset condition by asserting the RST signal for a minimum of a reset hold time. During the reset condition, the state of all SCSI bus signals is undefined.

All SCSI devices must release all SCSI bus signals (except RST) within a bus clear delay of the transition of the RST to true. The BUS FREE phase always follows a reset condition.

Hard Reset Alternative

The 3929 implements the Hard Reset Alternative. When a reset condition occurs, the 3929:

1. Clears all uncompleted commands.

2. Sets all internal registers to their power-up states.

SCSI Bus Phase Sequences

The order in which phases are used on the SCSI bus follows a prescribed sequence. The reset condition can abort any phase and is always followed by a BUS FREE phase. Also, any other phase can be followed by the BUS FREE phase, but many such instances are error conditions. The additional allowable sequences are shown in Figure 1.

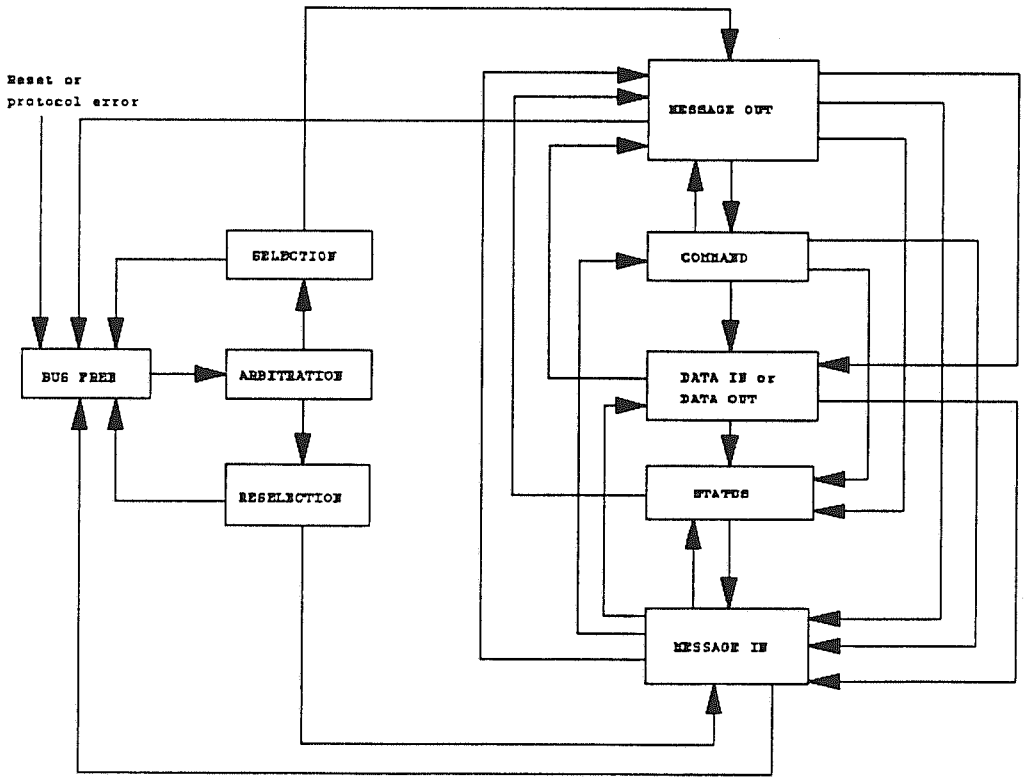


FIGURE 1 - SCSI Bus Phase Sequences

The normal progression is from the BUS FREE phase to ARBITRATION, from ARBITRATION to SELECTION or RESELECTION, and from SELECTION or RESELECTION to one or more of the information transfer phases (COMMAND, DATA, STATUS, or MESSAGE). The final information phase is usually the MESSAGE IN phase where a DISCONNECT or COMMAND COMPLETE message is transferred, followed by the BUS FREE phase.

Message System Specification

The message system allows communication between an initiator and the target for the purpose of physical data path management.

Message Codes

The following table shows the messages that the 3929 implements:

TABLE 3 - 3929 Messages

CODE	SUPPORT INIT	SUPPORT TARG	MESSAGE NAME	DIRECTION		NEGATE ATN BEFORE LAST ACK
06h	O	M	ABORT		OUT	YES
0Ch	O	M	BUS DEVICE RESET		OUT	YES
00h	M	M	COMMAND COMPLETE	IN		---
04h	O	O	DISCONNECT	IN		---
04h	O	O	DISCONNECT		OUT	YES
05h	M	M	INITIATOR DETECTED ERROR		OUT	YES
09h	M	M	MESSAGE PARITY ERROR		OUT	YES
07h	M	M	MESSAGE REJECT	IN	OUT	YES
08h	M	M	NO OPERATION		OUT	YES
***	O	O	SYNCH DATA TRANSFER REQUEST	IN	OUT	YES
80h - FFh	M	M	IDENTIFY		OUT	NO

M -- MANDATORY
IN -- TARGET TO INITIATOR
YES -- INITIATOR MUST NEGATE ATN BEFORE LAST ACK OF MESSAGE
NO -- INITIATOR MAY OR MAY NOT NEGATE ATN BEFORE LAST ACK OF MESSAGE
0 -- OPTIONAL
OUT -- INITIATOR TO TARGET
-- NOT APPLICABLE

SCSI Messages

COMMAND COMPLETE -- 00 Hex

This message is sent from a target to an initiator to indicate that the execution of a command has been completed and that a valid status byte has been sent to the initiator. After successfully sending this message, the target goes to the BUS FREE phase by releasing the

BSY signal. The target considers this message transfer to be successful when it detects the negation of ACK for the COMMAND COMPLETE message with the ATN signal false.

DISCONNECT -- 04 Hex

The DISCONNECT message is sent from a target to inform an initiator that the present connection is going to be broken. This disconnection is accomplished by the target releasing the BSY signal. A later reconnection is required in order to complete the current requested command. After successfully sending this message, the target goes to the BUS FREE phase by releasing the BSY signal. The target considers the message transmission successful when it detects the negation of the ACK signal for the DISCONNECT message with the ATN signal false.

This message may also be sent from an initiator to a target to instruct the target to disconnect from the SCSI bus. If this option is supported, and after the DISCONNECT message is received, the target switches to MESSAGE IN phase, sends the DISCONNECT message to the initiator, and then disconnects from the SCSI bus by releasing the BSY signal. After releasing the BSY signal, the target cannot participate in another ARBITRATION phase until there is a disconnection delay. If this option is not supported, or the target cannot disconnect at the time when it receives the DISCONNECT message from the initiator, the target responds by sending a MESSAGE REJECT message to the initiator.

INITIATOR DETECTED ERROR -- 05 Hex

The INITIATOR DETECTED ERROR message is sent from an initiator to inform a target that an error has occurred. This does not preclude the target from retrying the operation. The source of the error may either be related to previous activity on the SCSI bus or may be internal to the initiator and unrelated to any previous SCSI bus activity.

ABORT -- 06 Hex

The ABORT message is sent from the initiator to the target to clear the present operation. All pending data and status for the initiator that issued the command are cleared, and the target goes to the BUS FREE phase. No status or ending messages are sent for this operation.

MESSAGE REJECT -- 07 Hex

The MESSAGE REJECT is sent from either a target or initiator to indicate that the last message or message byte it received was inappropriate or has not been implemented.

In order to indicate its intentions of sending this message, the initiator asserts the ATN signal prior to its release of the ACK signal for the REQ/ACK handshake of the message byte to be rejected. If the target receives this message under any other circumstances, it rejects this message.

When a target sends this message, it changes to the MESSAGE IN phase and sends this message prior to requesting additional message bytes from the initiator. This provides an interlock so that the initiator can determine which message byte is rejected.

After the target sends a MESSAGE REJECT message and if the ATN signal is still asserted, it returns to the MESSAGE OUT phase. The subsequent MESSAGE OUT phase begins with the first byte of the message.

NO OPERATION -- 08 Hex

The NO OPERATION message is sent from an initiator, in response to a targets request for a message, when the initiator does not currently have any other message to send.

MESSAGE PARITY ERROR -- 09 Hex

The MESSAGE PARITY ERROR message is sent from the initiator to the target to indicate that the last message byte it received had a parity error.

In order to indicate its intention of sending this message, the initiator asserts the ATN signal prior to its release of the ACK signal for the REQ/ACK handshake of the message byte that has the parity error. This provides an interlock so that the target can determine which message byte had the parity error.

If the target returns to the MESSAGE IN phase before switching to some other phase, after receiving the MESSAGE PARITY ERROR message, the target can resend the entire message that had the parity error.

BUS DEVICE RESET -- 0C Hex

The BUS DEVICE RESET message is sent from an initiator to direct a target to clear all current operations. This message forces a hard reset condition to be entered on the SCSI device. The target then goes to the BUS FREE phase after reception of this message.

IDENTIFY 80-FF Hex

The IDENTIFY message may be sent from either an initiator or target for the purpose of establishing a physical path connection between a target and initiator.

Bit Byte	7	6	5	4	3	2	1	0
0	IDENT	DSC	LUNTR	RSVD	RSVD	LUNTRN		

IDENT -- This bit is always set to a one to indicate an IDENTIFY message.

DSC -- This bit is only set to one by the initiator. When set to 1, it indicates that the initiator has the ability to accommodate disconnection and reconnection.

LUNTR -- This bit is set to zero indicates that the IDENTIFY message is directed to a logical unit. This bit is set to one to indicate the

IDENTIFY message is directed to a target routine that does not involve a logical unit.

RSVD -- These bits are reserved and must be set to zero.

LUNTRN -- This field is for the logical unit number target routine number. This field specifies a logical unit number if the LUNTR is set to zero. If the LUNTR bit is set to one, this field specifies a target routine number.

Since the 3929 does not implement target routines, the LUNTR bit must be set to zero. Also, since the 3929 does not support multiple logical units, the LUNTRN field must be set to zero.

SYNCHRONOUS DATA TRANSFER REQUEST

A Synchronous Data Transfer Request (SDTR) message exchange is initiated by a SCSI device whenever a previously arranged data transfer agreement may have become invalid. The agreement becomes invalid after any condition that may leave the data transfer agreement in an undeterminate state such as:

1. after a HARD reset condition
2. after a BUS DEVICE RESET message
3. after a power-up cycle

Any SCSI device that is capable of synchronous data transfers does not respond to this message with a MESSAGE REJECT message.

The SDTR message exchange establishes the allowable transfer periods and the REQ/ACK offsets of the two devices participating in the synchronous data transfer. The transfer period is the minimum time allowed between leading edges of successive REQ pulses and of successive ACK pulses to meet the requirements for successful reception of data.

The REQ/ACK offset is the maximum number of REQ pulses allowed to be outstanding before the leading edge of its corresponding ACK pulse is received at the target. A REQ/ACK offset of zero indicates that the asynchronous data transfer agreement exists. A value of FF hex indicates unlimited REQ\ACK offset.

The device which first sends the SDTR message sets its REQ/ACK offset to a value which it can successfully receive data. If the responding device can also receive data with these values, it returns the same values in its SDTR message. If it requires a larger transfer period, a smaller REQ/ACK offset, or both, it substitutes values in its SDTR message as required, returning unchanged any value not required to be altered. The successful completion of a SDTR message exchange implies an agreement as follows:

Responding Device SDTR Message	Implied Agreement
1. Non-zero REQ/ACK offset	Each device transmits data with a transfer period equal to or greater than, and a REQ/ACK offset equal to or less than the values received in the other devices SDTR message.
2. REQ/ACK offset equal zero	Asynchronous Transfers
3. MESSAGE REJECT	Asynchronous Transfers

If the initiator recognizes that negotiation is required, it asserts the ATN signal and sends a SDTR message to initiate the negotiation process. After successfully completing the MESSAGE OUT phase, the target responds with the proper SDTR message. Following the targets response, the implied agreement for synchronous data transfers is considered to be negated by both the target and initiator if the initiator asserts ATN and the first message out is either a MESSAGE REJECT or MESSAGE PARITY ERROR. In this case, both devices revert to the asynchronous data transfer mode of operation.

SCSI Command Specification

This section defines the SCSI commands supported by the 3929. Each command is composed of a group of command bytes and optionally, a group of data bytes. The number of data bytes associated with any given command depends on the command executed and/or the transfer count specified within the command. After ARBITRATION, SELECTION, and any optional message transfer phases, the 3929 enters the COMMAND phase and requests that the initiator send the command data bytes. The opcode of the command determines the number of data bytes for the command. After receiving all the requested data bytes, the 3929 enters either the DATA IN, DATA OUT, or STATUS phase.

Upon the successful, or unsuccessful, completion of an operation, the 3929 returns a status byte to the initiator. The 3929 returns a status byte equal to either zero or two. A status return of zero indicates GOOD. A status byte returned equal to two indicates CHECK CONDITION.

Since most error conditions cannot be adequately described with a status return of just one status byte, a REQUEST SENSE operation is then used to retrieve additional information regarding the CHECK CONDITION.

Command Descriptor Block

A request to a device is made by transferring a command to the target device. This set of command bytes is referred to as the Command Descriptor Block (CDB).

The SCSI specification defines three basic command formats, the six-byte command, the ten-byte command and the twelve-byte command. The 3929 supports the six and ten byte commands. The SCSI specification refers to the six-byte commands as Group 0 commands and

the ten-byte commands as Group 1 commands. The format of the command is recognized by its opcode. Six-byte commands have opcodes in the range of 00_H to $1F_H$; ten-byte commands have opcodes in the range of 20_H to $3F_H$; twelve-byte commands have opcodes in the range of $A0_H$ to BF_H. The format of the first byte of the Command Descriptor Block is shown in the table below.

TABLE 4 - First Byte in Command Description Block

Bit	7	6	5	4	3	2	1	0
	GROUP CODE			COMMAND CODE				

The following table shows a generic six-byte Command Descriptor Block.

TABLE 5 - Six Byte in Command Description Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	LOGICAL UNIT NUMBER			(MSB)				
2	LOGICAL UNIT ADDRESS (IF REQUIRED)							
3								
4	TRANSFER LENGTH							
5	CONTROL BYTE							

Logical Unit Number

The Logical Unit Number for the 3929 is zero. Logical Unit Numbers 1 through 7 are not accepted unless the command is preceded by a an IDENTIFY message in which the Logical Unit Number (LUN) is zero. In this case, the LUN specified in the command is ignored, and the LUN in the IDENTIFY message is used. If a LUN other than zero is specified in the Command Descriptor Block (CDB), and an IDENTIFY message with a LUN equal to zero is not issued, then the command is terminated with CHECK CONDITION and the sense key set to ILLEGAL REQUEST.

Logical Unit Address

This field is normally used for specifying the start block address for various operations. The 3929 does not use this field for most commands. Refer to the individual command descriptions for details on this field.

Transfer Length

This field is used to specify the number of bytes to be transferred to or from the 3929. For Group zero commands, the acceptable transfer lengths range from 00 to FF Hex. If a value of zero is specified, the number of bytes to be transferred is 256. See the allocation byte length for each command for further details.

For Group one commands, the acceptable transfer lengths range from 0000 to FFFF Hex. If a value of zero is specified, the number of bytes to be transferred is zero.

Reserved

These fields are reserved and must be set to zero.

Control Byte

The Control Byte is the last byte of any Command Descriptor Block. The following table shows the Control Byte.

TABLE 6 - Command Description Block Control Byte

Bit	7	6	5	4	3	2	1	0
	VENDOR UNIQUE		RESERVED				FLAG	LINK

Vendor Unique — This field is for a vendor unique code. For the 3929, no vendor unique codes are specified in this field. This field must be set to zero.

Reserved — This is a reserved field and must be set to zero.

Flag — This bit is not used by the 3929 and must be set to zero.

Link — This bit is set for using linked commands. Since the 3929 does not support linked commands, this bit must be set to zero.

Completion Status Byte

After the target has completed a command, it returns a status byte to reflect the status of the command. This byte is transferred during the Status phase of the operation. The 3929 returns either one of two status responses. A status return of GOOD status indicates that the command terminated successfully. A status return of CHECK CONDITION indicates that an error has occurred or some other unexpected event which requires servicing. After a CHECK CONDITION has been issued, the initiator should execute a REQUEST SENSE operation to obtain details concerning the CHECK CONDITION.

Model 3929-Z1B

The following table shows the returned Status byte:

TABLE 7 - Status Byte

Bit	7	6	5	4	3	2	1	0
	RESERVED			STATUS BYTE CODE				RSVD

RESERVED This field is reserved and returned as zero.

STATUS BYTE CODE This field is used for specifying the status of the command. The following chart shows the status returns. The 3929 only returns the first two types of status.

RESERVED This bit is reserved and returned as zero.

TABLE 8 - Status Byte Bits

Bits of the Status Byte								
7	6	5	4	3	2	1	0	Status
R	R	0	0	0	0	0	R	GOOD
R	R	0	0	0	0	1	R	CHECK CONDITION
R	R	0	0	0	1	0	R	CONDITION MET
R	R	0	0	1	0	0	R	BUSY
R	R	0	1	0	0	0	R	INTERMEDIATE
R	R	0	1	0	1	0	R	INTERMEDIATE -- CONDITION MET
R	R	0	1	1	0	0	R	RESERVATION CONFLICT
R	R	1	0	0	0	1	R	COMMAND TERMINATED
R	R	1	0	1	0	0	R	QUEUE FULL
ALL OTHER CODES								RESERVED

GOOD This status indicates that the target has successfully executed the command.

CHECK CONDITION This status indicates that an error has occurred or an unexpected condition has occurred that requires service.

CONDITION MET This status or INTERMEDIATE CONDITION MET is returned whenever a requested operation is satisfied. (i.e., SEARCH DATA or PRE-FETCH)

BUSY This status indicates that the target is busy. This status is returned whenever a target is unable to accept a command from an otherwise acceptable initiator. It is recommended that when this status is received that the command be re-issued.

INTERMEDIATE — This status or INTERMEDIATE CONDITION MET is returned for every successfully completed command in a series of linked commands, unless the command is terminated with CHECK CONDITION, RESERVATION CONFLICT, or COMMAND TERMINATED status.

INTERMEDIATE CONDITION MET — This status is the combination of CONDITION MET and INTERMEDIATE.

RESERVATION CONFLICT — This status is returned whenever an initiator attempts to access logical unit or an extent within a logical unit that is reserved with a conflicting reservation type for another SCSI device.

COMMAND TERMINATED — This status is returned whenever the target terminates the current I/O process after receiving a TERMINATE I/O PROCESS message.

QUEUE FULL — This status is returned when a SIMPLE QUEUE TAG, ORDERED QUEUE TAG or TAG HEAD OF QUEUE message is received and the command queue is full.

Commands Supported by the 3929

TABLE 9 - 3929 Commands

OPCODE	COMMAND
00 Hex	TEST UNIT READY
03 Hex	REQUEST SENSE
06 Hex	UNBOOK LAM
08 Hex	RECEIVE
0A Hex	SEND
12 Hex	INQUIRY
1D Hex	SEND DIAGNOSTIC
09 Hex	SINGLE
0C Hex	SETUP
0D Hex	REGISTER ACCESS
0E Hex	RESUME LIST
20 Hex	EXECUTE LIST
22 Hex	BLOCK
23 Hex	LOAD LIST
A0 Hex	BOOK LAM

TEST UNIT READY Command

The TEST UNIT READY command is used to determine if the 3929 is capable of receiving and executing a CAMAC command. There is no data phase associated with this command. After the Command Descriptor Block is received, the 3929 enters the Status phase. The 3929

returns status of either CHECK CONDITION or GOOD. After the Status phase, the 3929 sends the Command Complete Message.

The Status byte returned depends on the state of the 3929's front panel On-Line/Off-Line switch. If the switch is in the On-Line position, the status returned is GOOD. If the switch is in the Off-Line position, CHECK CONDITION status is returned. The following table shows the Command Descriptor Block for the TEST UNIT READY command.

TABLE 10 - Test Unit Ready Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 00 hex							
1	LUN = 0			RESERVED = 0				
2	RESERVED = 0							
3	RESERVED = 0							
4	RESERVED = 0							
5	CONTROL BYTE							

Byte 0: This byte is set to 0 to indicate a TEST UNIT READY command.
Byte 1: This byte contains a Logical Unit and Reserved field. These fields MUST be set to zero.
Byte 2: This is a Reserved byte and MUST be set to zero.
Byte 3: This is a Reserved byte and MUST be set to zero.
Byte 4: This is a Reserved byte and MUST be set to zero.
Byte 5: This is the Control byte and MUST be set to zero.

REQUEST SENSE Command

The following table shows the command descriptor block for the Request Sense operation.

TABLE 11 - Request Sense Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 03 hex							
1	LUN = 0			RESERVED = 0				
2	RESERVED = 0							
3	RESERVED = 0							
4	ALLOCATION LENGTH IN BYTES							
5	VU = 0			RESERVED = 0			F	L

The Request Sense operation requests that the target return sense data to the initiator. This command is normally executed in response to a CHECK CONDITION status return from another SCSI command.

The sense data contains information regarding the previous command executed. This data is saved for the initiator until it is retrieved. Once the Request Sense is executed, the sense data is cleared (i.e., sense key, sense code,...). Table 11 shows the 6-byte Request Sense command.

Byte 0: This byte is set to 03 hex to indicate a Request Sense operation.
Byte 1: This byte contains the Logical Unit Number to be accessed and a reserved field. For the 3929 these fields MUST be set to zero.
Byte 2: This is a reserved byte and MUST be set to zero.
Byte 3: This is a reserved byte and MUST be set to zero.
Byte 4: This byte is used for specifying the maximum number of bytes that the initiator has allocated for returned sense data. This value can range from 0 to FF hex. If an allocation length of 0 is specified, the 3929 returns no data. The 3929 terminates the data transfer phase when the allocated number of sense bytes have been returned OR when all available sense bytes been returned, whichever is less. For most all cases, an allocation length of 18 bytes is sufficient.
Byte 5: This is the Control byte and MUST be set to zero.

The following table shows the 14 bytes of data returned from a Request Sense operation.

TABLE 12 - 14-Byte Request Sense

Bit Byte	7	6	5	4	3	2	1	0
0	VLD	ERROR CODE = 70 hex						
1	SEGMENT NUMBER							
2	FM	EOM	ILI	RSVD	SENSE KEY			
3	(MSB)	INFORMATION BYTE						
4	INFORMATION BYTE							
5	INFORMATION BYTE							
6	INFORMATION BYTE							(LSB)
7	ADDITIONAL SENSE LENGTH IN BYTES							
8	(MSB)	COMMAND SPECIFIC INFORMATION						
9	COMMAND SPECIFIC INFORMATION							
10	COMMAND SPECIFIC INFORMATION							
11	COMMAND SPECIFIC INFORMATION							
12	ADDITIONAL SENSE CODE							
13	ADDITIONAL SENSE CODE QUALIFIER							

Byte 0: This byte contains a "valid" bit and the error code. If the valid bit is returned as zero, the "information" bytes are not defined. The error code is set to 70 hex to indicate the returned data is for current errors not deferred errors. The 3929 always returns this byte set to 70 hex.

Model 3929-Z1B

Byte 1: This byte is used for COPY and COMPARE SCSI commands. Since the 3929 does not implement these commands, this byte is returned as zero.

Byte 2: This byte contains the Sense Key and several bits used for sequential access devices. The 3929 uses only the Sense Key field. The remaining bits in this field are set to zero. The Sense Key is a general indicator for the cause of a CHECK CONDITION. Refer to the Sense Key and Codes section of this manual for further details.

Bytes 3 through Bytes 6 are Information bytes. These bytes are not used by the 3929 and are returned as zero.

Byte 7: This byte contains the Additional Sense Byte Length. This byte specifies the number of additional bytes, from byte 7, that contain valid sense data. The 3929 sets this byte to 22 hex.

Bytes 8 through Bytes 11 are Command Specific Information Bytes. These bytes are not used by the 3929 and returned as zero.

Byte 12: This byte contains the Sense Code. The Sense Code provides additional information regarding the cause of the CHECK CONDITION. The code is used in conjunction with the Sense Key to further define the error.

Byte 13: This byte contains the Sense Code Qualifier. The Sense Code Qualifier provides additional information regarding the cause of the CHECK CONDITION. The qualifier is used in conjunction with the Sense Key and Sense Code to determine the exact cause of the CHECK CONDITION.

Bytes 14 through 85 are used by the KSC Software Driver for the 3929. Those bytes are described below:

BYTE		BYTE	
14	Reserved	50	STAWC BYTE 1
15	Reserved	51	STAWC BYTE 2
16	Reserved	52	STAWC BYTE 3
17	Reserved	53	STAWC BYTE 4
18	STAT BYTE 1	54	STAQXE BYTE 1
19	STAT BYTE 2	55	STAQXE BYTE 2
20	STAT BYTE 3	56	STAQXE BYTE 3
21	STAT BYTE 4	57	STAQXE BYTE 4
22	STACSR BYTE 1	58	WCERS BYTE 1
23	STACSR BYTE 2	59	WCERS BYTE 2
24	STACSR BYTE 3	60	WCERS BYTE 3
25	STACSR BYTE 4	61	WCERS BYTE 4
26	STAERS BYTE 1	62	WCCNT BYTE 1

BYTE		BYTE	
27	STAERS BYTE 2	63	WCCNT BYTE 2
28	STAERS BYTE 3	64	WCCNT BYTE 3
29	STAERS BYTE 4	65	WCCNT BYTE 4
30	STACCS BYTE 1	66	WCLIS BYTE 1
31	STACCS BYTE 2	67	WCLIS BYTE 2
32	STACCS BYTE 3	68	WCLIS BYTE 3
33	STACCS BYTE 4	69	WCLIS BYTE 4
34	STASUM BYTE 1	70	WCDAT BYTE 1
35	STASUM BYTE 2	71	WCDAT BYTE 2
36	STASUM BYTE 3	72	WCDAT BYTE 3
37	STASUM BYTE 4	73	WCDAT BYTE 4
38	STACNT BYTE 1	74	QXEERS BYTE 1
39	STACNT BYTE 2	75	QXEERS BYTE 2
40	STACNT BYTE 3	76	QXEERS BYTE 3
41	STACNT BYTE 4	77	QXEERS BYTE 4
42	STALIS BYTE 1	78	QXELIS BYTE 1
43	STALIS BYTE 2	79	QXELIS BYTE 2
44	STALIS BYTE 3	80	QXELIS BYTE 3
45	STALIS BYTE 4	81	QXELIS BYTE 4
46	STADAT BYTE 1	82	QXEDAT BYTE 1
47	STADAT BYTE 2	83	QXEDAT BYTE 2
48	STADAT BYTE 3	84	QXEDAT BYTE 3
49	STADAT BYTE 4	85	QXEDAT BYTE 4

Control/Status Register (CSR) Read Only

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	
SCSI ID 2	SCSI ID 1	SCSI ID 0	BYTE ORDR	RCV FF	RCV EF	XMT FF	XMT EF	ABORT	0	N>23 / QTMO	RES	RFS	DONE	NO X	NO Q	CONTROLLER STATUS

BIT	MNEMONIC	DESCRIPTION
15-13	SCSI ID 2-0	SCSI ID 2, 1, and 0 contain the Identification value for the 3929 crate controller.
12	BYTE ORDER	BYTE ORDER instructs the 3929 if the host machine expects the high or low byte to be sent first. (1 = High First)

11	RCV FF	RECEIVE FULL FIFO is set if the RECEIVE FIFO is full.
10	RCV EF	RECEIVE EMPTY is set if the RECEIVE FIFO is empty.
9	XMT FF	TRANSMIT FULL is set if the TRANSMIT FIFO is full.
8	XMT EF	TRANSMIT EMPTY is set if the TRANSMIT FIFO is empty.
7	ABORT	ABORT is a read-only bit which is set when an operation was terminated due to one of the following conditions in the ABORT definition.

ABORT = (BM·TM2·TM1·NOQ) + (BM·TM2·TM1·N>23)
+ (BM·TM2·TM1·NOX·AD)
+ (TM2·TM1·QRPT TM0)

BM - Block Mode (bit 8 of MCR)
TM2 - Transfer Mode 2 (bit 7 of MCR)
TM1 - Transfer Mode 1 (bit 6 of MCR)
NO-Q - CAMAC Q-response = 0 (bit 0 of ERS)
N>23 - Station number (N) greater than 23 (bit 6 of ERS)
NO-X - CAMAC X=response = 0 (bit 1 of ERS)
AD - Abort disable (bit 3 of MCR)
QRPT TM0 - Q-repeat timeout (bit 6 of ERS)

6	Reserved	Read as zero.
5	N>23/QRPT TM0	STATION NUMBER (N) GREATER THAN 23/Q-REPEAT TIMEOUT. This bit is set by one of two conditions (1) Executing Q-Scan operations and NO-Q's were obtained causing the 3929 to scan past station number 23, or (2) Executing a Q-Repeat operation and a Q=1 was not obtained within 60 milliseconds.
4	Reserved	Read value may vary.
3	RFS	REQUEST-FOR-SERVICE is set when a CAMAC LAM is pending.
2	DONE	DONE is set when the 3929 has completed an operation.
1	NO-X	NO-X. When set, indicates that the last CAMAC operation executed resulted in an X-response of zero.
0	NO-Q	NO-Q. When set, indicates that the last CAMAC operation executed resulted in a Q-response of zero.

Sense Keys and Codes

The following shows the Sense Key, Sense Code, and Sense Code Qualifiers generated by the 3929. All data values are in hex.

Sense Key	Sense Code	Sense Qualifier		Description
00	00	00	NO SENSE	This data is returned to indicate that there is no specific sense key information to be reported. This would be the case for a successful command.
02	04	03	NOT READY	This data is returned if whenever the 3929 cannot be accessed. Operator intervention is required to correct this condition. This key is generated if a TEST UNIT READY command is executed and the 3929 is in the Off-Line state.
04	42	XX	HARDWARE ERROR	This data is returned whenever the 3929 self-test fails. The self-test is entered on power-up or by a self-test request issued by the SEND DIAGNOSTIC command. XX is the code returned to specify the cause of the hardware error. Refer to the SEND DIAGNOSTIC SCSI command description for details regarding the error returns.
05	-	-	ILLEGAL REQUEST	This key is returned to indicate that there was an illegal parameter in the Command Descriptor Block or in the additional parameters supplied for some commands. If the 3929 detects an invalid parameter, it terminates the command without performing any operations.
05	00	00	BAD CONTROL FIELD	This code is returned when the 3929 receives a non-zero control byte in the Command Descriptor Block.
05	20	00	BAD COMMAND RECEIVED	This code is returned when the 3929 was instructed to execute a SCSI command that has not been implemented.

Model 3929-Z1B

05	24	00	BAD RESERVED FIELD	This code is returned when a non-zero reserved field is encountered in the Command Descriptor Block.
05	25	00	BAD LUN FIELD	This code is returned when a non-zero logical unit number (LUN) is found in the Command Descriptor Block.
05	80	00	BAD LIST OPCODE	This code is returned when the 3929 encounters a list processing instruction that the 3929 does not implement.
05	80	01	BAD CAMAC FUNCTION	This code is vendor unique and is generated when the CAMAC function code does not watch the SCSI command requested. For example, the SETUP command specified a CAMAC read operation and a subsequent SCSI SEND command is requested.
05	80	02	BAD CAMAC MODE	This code is vendor unique and is generated when the CAMAC mode byte specifies an operation that the SCSI command does not support. For example, setting the "BLK" mode specifier and requesting a SINGLE SCSI command.
05	80	03	BAD WORD SIZE	This code is vendor unique and is generated when the CAMAC mode byte contains an illegal word size.
05	80	04	LAM OVERFLOW	This code is vendor unique and is generated when more than 100 LAMS are backed up on the 3929-Z1B.
05	81	01	BAD LIST ADDRESS	This vendor unique code is generated when a list processing command is executed with a list address outside the available range.
06	29	00	UNIT ATTENTION	This data is returned when a reset has occurred to the 3929. This reset is generated by a power-up, receiving a SCSI RESET message, or by receiving the assertion of the SCSI bus RESET signal.

09	80	01	DEVICE OFF-LINE, SINGLE NAF	This vendor unique code is generated when a request for a SINGLE CAMAC operation is executed and the 3929 is off-line. This code is KSC ERROR #310.
09	80	02	SOFTWARE TIMEOUT, SINGLE NAF	This vendor unique code is generated when a requested CAMAC Dataway operation is not completed within 200 milliseconds. This code is KSC ERROR #213.
09	80	03	N>23, SINGLE NAF	This vendor unique code is generated when a SINGLE CAMAC operation is requested and the Q-SCAN operation caused the scan increment to station number 24. This code is KSC ERROR #311.
09	80	04	HARDWARE TIMEOUT, SINGLE NAF	This vendor unique code is generated when the 33929 hardware has timed-out trying to obtain a Q = 1 response during a Q-REPEAT operation. If a Q = 1 response is not obtained within 200 milliseconds, the code is returned. This is KSC ERROR #317.
09	80	05	NO-X, SINGLE NAF	This vendor unique code is generated when a SINGLE CAMAC command resulted in a NO-X condition. This code is KSC ERROR #314.
09	80	06	NO-Q, SINGLE NAF	This vendor unique code is generated when a SINGLE CAMAC command resulted in a NO-Q condition. This code is KSC ERROR #312.
09	80	07	DEVICE OFF-LINE, BLOCK NAF	This vendor unique code is returned when a request for a Block Transfer operation is executed and the 3929 is off-line. This is KSC ERROR #301.
09	80	08	SOFTWARE TIMEOUT, BLOCK NAF	This vendor unique code is generated when a requested Dataway operation during a Block Transfer operation did not complete

				within 200 milliseconds. This is KSC ERROR #207.
09	80	09	N>23, BLOCK NAF	This vendor unique code is returned when a Block Transfer operation is requested and the Q-SCAN operation caused the scan to increment to station number 24. This is KSC ERROR #302.
09	80	10	BLOCK UNDEFINED ERROR	This vendor unique code is generated when an undefined block error has occurred.
09	80	11	SINGLE UNDEFINED ERROR	This vendor unique error is generated when an undefined single error has occurred.
OB	-	-	ABORTED COMMAND	This key is returned to indicate that the 3929 could not complete the requested operation due to either a CAMAC Abort condition or a SCSI error.
0B	47	00	SCSI PARITY ERROR	This data is returned when a SCSI parity error is detected.
0B	43	00	MESSAGE REJECT ERROR	This data is returned when the 3929 receives a message it does not implement.
0B	80	01		Single CAMAC Operation Aborted. This data is returned when a CAMAC operation terminates before the requested byte count is exhausted.
0B	80	02		Block CAMAC Operation Aborted. This vendor unique code is generated when a block transfer operation is terminated before the transfer count is exhausted.

RECEIVE Command

The RECEIVE command is used to transfer data from the 3929 to the initiator. This command is used along with the SETUP command. The SETUP command loads the CAMAC Station Number (N), Subaddress (A), Function Code (F) and the Mode Control data for the block transfer operation. If the SETUP command specified a CAMAC read operation, transfer from the CAMAC Dataway to the initiator, a subsequent RECEIVE command must be executed to retrieve the data. The number of bytes to be transferred is specified in the command. Refer to the SETUP command for further details regarding use of the RECEIVE command.

Model 3929-Z1B

The following table shows the Command Descriptor Block for the RECEIVE command.

TABLE 13 - RECEIVE Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 08 HEX							
1	LUN = 0			RESERVED = 0				
2	(MSB)	TRANSFER BYTE COUNT HI						
3	TRANSFER BYTE COUNT MID							
4	TRANSFER BYTE COUNT LO							(LSB)
5	CONTROL BYTE							

Byte 0: This byte is set to 08 Hex to indicate a RECEIVE command.
Byte 1: This byte contains a Logical Unit field and a Reserved field. This Byte MUST be set to zero.
Bytes 2-4: These bytes are used to specify the byte count for the operation. Byte 2 is the most-significant and Byte 4 is the least significant.
Byte 5: This is the Control byte and MUST be set to zero.

SEND Command

The SEND command is used to transfer data from the initiator to the 3929. This command is used along with the SETUP command. The SETUP command loads the CAMAC Station Number (N), Subaddress (A), Function Code (F) and the Mode Control data for the block transfer operation. If the SETUP command specified a CAMAC write operation, transfer from the initiator to the CAMAC Dataway, a subsequent SEND command must be executed in order to send the write data to the 3929. The number of bytes to be transferred is specified in the command. Refer to the SETUP command for further details regarding use of the SEND command.

The following table shows the Command Descriptor Block for the SEND command.

TABLE 14 - SEND Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 0A HEX							
1	LUN = 0			RESERVED = 0				
2	(MSB)	TRANSFER BYTE COUNT HI						
3	TRANSFER BYTE COUNT MID							
4	TRANSFER BYTE COUNT LO							
5	CONTROL BYTE							(LSB)

Byte 0: This byte is set to 0A Hex to indicate a SEND command.
Byte 1: This byte contains a Logical Unit and Reserved field. This byte MUST be set to zero.
Bytes 2-4: These bytes are used to specify the byte count for the operation. Byte 2 is the most significant and Byte 4 is the least significant.
Byte 5: This is the Control Byte and MUST be set to zero.

INQUIRY Command

The INQUIRY command is used by an initiator to determine the peripherals attached to the SCSI bus. The following table shows the Command Descriptor Block for the INQUIRY command.

TABLE 15 - INQUIRY Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 12 hex							
1	LUN = 0			RESERVED = 0				EVPD
2	PAGE CODE							
3	RESERVED = 0							
4	ALLOCATION LENGTH IN BYTES							
5	CONTROL BYTE							

Byte 0: This byte is set to 12 hex to indicate an INQUIRY command.
Byte 1: This byte contains a Logical Unit and Reserved field, along with the Enable Vital Product Information (EVPD) bit. Since the 3929 does not support Vital Product Information or Logical Units, this byte MUST be set to zero.

Byte 2: This byte specifies which Page Code of the Vital Product Information is to be returned. Since the 3929 does not support it, this byte MUST be set to zero.

Byte 3: This is a Reserved byte and MUST be set to zero.

Byte 4: This byte specifies the maximum number of data bytes that the initiator has allocated for returned INQUIRY data. An allocation length of zero indicates that the 3929 should return no INQUIRY data. Any other value indicates the maximum number of bytes that shall be returned. The 3929 terminates the DATA IN phase when the allocation length of bytes is exhausted or when all available INQUIRY data has been sent to the initiator, whichever is less. For the 3929 an allocation length of 56 (38 hex) is suggested.

Byte 5: This is the Control byte and MUST be set to zero.

The INQUIRY command returns an 8-byte header followed by product specific information. The following table shows the 8-byte header.

TABLE 16 - 8-Byte INQUIRY Header Return

Bit / Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER			PERIPHERAL DEVICE TYPE				
1	RMB	RESERVED = 0						
2	ISO VERSION		ECMA VERSION			ANSI VERSION		
3	AENC	TRMIOP	RESERVED = 0			RESPONSE DATA FORMAT		
4	ADDITIONAL INQUIRY BYTE LENGTH							
5	RESERVED = 0							
6	RESERVED = 0							
7	RELADR	WBUS32	WBUS16	SYNC	LINK	RSVD	QUE	SRST

Byte 0: This byte contains the Peripheral Device Type and the Peripheral Qualifier. The 3929 returns this byte with the Peripheral Device Type set to 03 hex, indicating a processor device. The Peripheral Qualifier is returned as zero. If the INQUIRY command is performed to a Logical Unit other than zero, the Peripheral Qualifier returned is set to 03 hex. A return of 03 in this field indicates that the target is not capable of supporting a physical device on this logical unit.

Byte 1: This byte contains the Removable Medium Bit (RMB) and a Reserved field. Since the 3929 does not contain any removable media, this byte is returned as zero.

Byte 2: This byte contains information regarding the guidelines and specifications this product was designed under. The usage of the ISO version and ECMA version fields are defined by the International Organization for Standardization and the European Computer Manufacturers Association, respectively. A value of zero in these fields indicates that the target does not claim compliance with these standards. The ANSI (American National Standard Institute) version field indicates which version of the ANSI SCSI specification this device was implemented under. This field is set to 02 hex by the 3929 to indicate it was implemented under the SCSI-2 specifications. SCSI-2 is an extension to SCSI-1 (X3.131-1986) specification.

Byte 3: This byte contains the Asynchronous Event Notification Capability (AENC) bit, the Terminate I/O Process bit (TRMIOP), and the Response Data Format field. Since the 3929 supports Asynchronous Event Notification for CAMAC Look-At-Me (LAMs), this bit is set to a one. The TRMIOP bit is set to zero indicating that the 3929 does not support the TERMINATE I/O PROCESS message. The Response Data Format field is set to 02 hex by the 3929 indicating that the returned INQUIRY data conforms to SCSI-2 specifications.

Byte 4: This byte specifies the additional length of INQUIRY data words in the parameter list. For the 3929 this value is 52 (34 hex). This value indicates the number of INQUIRY data bytes in the parameter list that are to follow byte four. Therefore, after byte four is received, an additional 52 data bytes are available in the parameter list.

Model 3929-Z1B

Byte 5: This byte is a Reserved byte and returned as zero.

Byte 6: This byte ia a Reserved byte and returned as zero.

Byte 7: This byte contains information regarding supported SCSI features on the target. This byte is returned as 0 hex by the 3929. The following describes each bit and the value that is returned by the 3929.

RELADR - The Relative Addressing bit is set to indicate that the target supports the Relative Addressing Mode for this Logical Unit. Since the 3929 does not support Relative Addressing, this bit is returned as zero.

WBUS32 - The Wide Bus 32 bit is set to indicate that the target supports 32-bit wide data transfers. Since the 3929 supports only 8-bit data transfers, this bit is returned as zero.

WBUS16 - The Wide Bus 16 bit is set to indicate that the target supports 16-bit wide data transfers. Since the 3929 supports only 8-bit data transfers, this bit is returned as zero.

SYNC - The Synchronous Transfer bit is set to indicate that the target supports the synchronous data transfers on the SCSI bus. Since the 3929 currently does not support synchronous data transfers, this bit is returned as zero.

LINK - The Linked bit is set to indicate that the target supports linked commands for this Logical Unit. Since the 3929 does not support linked commands, this bit is returned as zero.

RSVD - This is a Reserved bit and returned as zero.

QUE - The Command Queuing is set to indicate that the target supports tagged Command Queuing. Since the 3929 does not support Command Queuing, this bit is returned as zero.

SRST - The Soft Reset bit is set to indicate that the target responds to the RESET condition using the soft reset alternative. Since the 3929 uses the hard reset alternative, this bit is returned as zero.

Bytes 8 through 15 are allocated for the Vendor Identification field. This field contains eight bytes of ASCII data identifying the vendor of this product. Bytes 8 through 15 are shown below with the hex data returned and the corresponding ASCII character.

Model 3929-Z1B

Byte	Hex data	ACSII character
8	4B	K
9	49	I
10	4E	N
11	53	S
12	59	Y
13	53	S
14	43	C
15	4F	O

Bytes 16 through 31 are allocated for the Product Identification field. This field contains 16 bytes of ASCII data identifying this product. Bytes 16 through 31 are shown below with the hex data returned and the corresponding ASCII character.

Byte	Hex data	ASCII character
16	33	3
17	39	9
18	32	2
19	39	9
20	2D	-
21	5A	Z
22	31	1
23	42	B
24	5F	
25	53	S
26	43	C
27	53	S
28	49	I
29	5F	
30	43	C
31	43	C

Bytes 32 through 35 are allocated for the Product Revision Level field. This field contains four bytes of ASCII data identifying the current revision level of the 3929 hardware. Bytes 32 through 35 are shown below with the hex data returned and the corresponding ASCII characters.

Byte	Hex data	ASCII character
32	31	1
33	2E	.
34	30	0
35	30	0

Bytes 36 through 49 reflect the 3929's current firmware revision level. Bytes 36 through 44 are shown below with the HEX data returned and the corresponding ASCII characters.

Model 3929-Z1B

Byte	Hex data	ASCII character
36	46	F
37	49	I
38	52	R
39	4D	M
40	57	W
41	41	A
42	52	R
43	45	E
44	20	space

Bytes 45 through 49 reflect the current firmware revision number.
Bytes 50 through 55 are returned as 20 Hex, representing spaces.

SEND DIAGNOSTIC Command

The SEND DIAGNOSTIC command requests the target to perform diagnostic operations on itself, on the logical unit, or both. The 3929 only responds to this command if a request for a self-test is issued. There is no DATA phase associated with this command. After the 3929 receives the request for the self-test, the microprocessor exercises write/read tests to the internal registers. Once the self-test is complete, the 3929 enters the STATUS phase and returns the status byte. If the self-test completed successfully, a status of GOOD is returned. A status response of CHECK CONDITION is returned for a failure during the self-test. For self-test failures, the sense key is set to HARDWARE ERROR.

The following occurs during the self-test:

1. If this is the first command the 3929 has received since power-up, the following registers are checked to verify a proper power-up sequence:

   a. Control/Status Register
   b. LAM Mask LO Register
   c. LAM Mask MID Register
   d. LAM Mask HI Register
   e. LAM LO Register
   f. LAM MID Register
   g. LAM HI Register
   h. Crate/MCR Register
   i. CAMAC CSR LO Register
   j. CAMAC CSR HI Register

   If this is not the first command to be issued to the 3929, the 3929 issues an interface reset to itself to clear all internal registers.

2. A Data read/write test is performed on the following registers:

   a. LAM Mask LO Register

b. LAM Mask MID Register
c. LAM Mask HI Register
d. Crate/MCR Register
e. NAF Register
f. XMT FIFO
g. RECV FIFO

3. A test is performed to generate a LAM on the 3929.

4. A test is performed to test all the List Memory on board the 3929.

NOTE: If the 3929 fails during the send diagnostic command, a sense key of 04 Hex, a sense code of 42 Hex, and a sense qualifier of XX will be returned during the next request sense command issued to the 3929. The sense qualifier will reflect which portion of the self test that failed. Below is a listing of all the codes that are returned with the sense qualifier and and the meaning of the code.

Sense Qualifier Meaning

0x00 -- No error.
0x01 -- Controller status register has been powered up with wrong initial value. Expected value is 0x0507.
0x02 -- LAM Mask LO register has been powered up with a non-zero value within it.
0x03 -- LAM Mask MID register has been powered up with a non-zero value within it.
0x04 -- LAM Mask HI register has been powered up with a non-zero value within it.
0x05 -- LAM LO register has been powered up with a non-zero value within it.
0x06 -- LAM MID register has been powered up with a non-zero value within it.
0x07 -- LAM HI register has been powered up with a non-zero value within it.
0x08 -- Crate/Mode register has been powered up with a non-zero value within it.
0x09 -- CSR Low register has been powered up with wrong initial value. Expected value is 0x44.
0x0A -- CSR HI register has been powered up with a non-zero value within it.
0x0B -- LAM Mask LO register has failed its self-test routine. The value written out to it is not the same value that was read in from it.
0x0C -- LAM Mask MID register has failed its self-test routine. The value written out to it is not the same value that was read in from it.
0x0D -- LAM Mask HI register has failed its self-test routine. The value written out to it is not the same value that was read in from it.

0x0E -- Crate/Mode register has failed its self-test routine. The value written out to it is not the same value that was read in from it.

0x0F -- Crate/Mode register has failed its self-test routine. The crate bits did not go HI when the value 0x0300 was written out to it.

0x10 -- NAF register has failed its self-test routine. The value written out to it is not the same value that was read in from it.

0x11 -- XMT FIFO has failed its self-test routine. The value written out to it is not the same value that was read in from it. Value = 0xAA.

0x12 -- RCV FIFO has failed its self-test routine. The value written out to it is not the same value that was read in from it. Value = 0xAA.

0x13 -- XMT FIFO has failed its self-test routine. The value written out to it is not the same value that was read in from it. Value = 0x55.

0x14 -- RCV FIFO has failed its self-test routine. The value written out to it is not the same value that was read in from it. Value = 0x55.

0x15 -- After information was placed within both FIFOs, the Controller Status Register had both Empty FIFO bits on.

0x16 -- After both FIFOs were emptied by means of a reset FIFO command, the Controller Status Register did not reflect this. Both Empty FIFO bits were on.

0x17 -- After both FIFOs were filled, the Controller Status Register did not reflect this by having both FIFO Full bits on.

0x18 -- After a LAM has been generated, the SLP bit inside the CSR HI register did not go true.

0x19 -- List Memory has failed its self-test routine. The value written out to it is not the same value that was read in from it. Value = 0xAAAA.

0x1A -- List Memory has failed its self-test routine. The value written out to it is not the same value that was read in from it. Value = 0x5555.

0x1B -- List Memory has failed its self-test routine. The value written out to it is not the same value that was read in from it.

Model 3929-Z1B

The following table shows the Command Descriptor Block for the SEND DIAGNOSTIC Command.

TABLE 17 - SEND DIAGNOSTIC Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 1D HEX							
1	LUN = 0			PF	RSVD	SLFTEST	DEVOFL	UNITOFL
2	RESERVED = 0							
3	PARAMETER LIST							
4	LENGTH							
5	CONTROL BYTE							

Byte 0: This byte is set to 1D Hex to indicate a SEND DIAGNOSTIC command.

Byte 1: This byte contains a Logical Unit field, a Reserved bit, and several other bits for the SETUP Command which must be set to 0.

Byte 2: This is a reserved byte and MUST be set to zero.

Bytes
3 & 4: These bytes are used to instruct the device only to perform certain portions of its selftest.

SETUP Command

The SETUP command is used for loading the internal 3929 registers before a CAMAC block transfer is executed. This command is normally used preceding a RECEIVE or SEND data command. It contains the CAMAC Station Number (N), Subaddress (A), and Function code (F), along with the CAMAC Mode to be used for the block transfer. The following table shows the command descriptor block for the SETUP command.

TABLE 18 - SETUP Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 0C hex							
1	LUN = 0			RESERVED = 0				
2	MODE CONTROL							
3	CAMAC NAF HI							
4	CAMAC NAF LO							
5	CONTROL BYTE							

Model 3929-Z1B

Byte 0: This byte is set to 0C hex to indicate a SETUP command

Byte 1: This byte contains a Logical Unit field and a reserved field. This byte MUST be set to zero.

Byte 2: This byte specifies the CAMAC Mode to be used for the block transfer operation. Refer to the CAMAC Operating Modes section for further details.

Bytes 3 and
Bytes 4:
These bytes are used for specifying the CAMAC Station Number (N), Subaddress (A), and Function Code (F) to be used for the block transfer operation. These bytes MUST be set to zero since the 3929 does not implement this feature.

Byte 5: This is the control byte and MUST be set to zero.

After the SETUP command is executed, the 3929 is awaiting either a SEND or RECEIVE command, unless the CAMAC function code specifies a Control operation. If the CAMAC function to be executed is a Control, the 3929 will execute the command before the SCSI status phase is entered. If the CAMAC function to be executed is a CAMAC write operation, the 3929 is expecting the SETUP command to be followed by a SEND command. If the SETUP requests a CAMAC read operation. The 3929 is expecting a subsequent RECEIVE command. The number of bytes to be transferred for the CAMAC write or read block transfer is specified in the command descriptor block for the SEND or RECEIVE command. Refer to the SEND and RECEIVE command description for further details.

SINGLE CAMAC OPERATION Command

The SINGLE CAMAC operation command is used for executing single CAMAC operations. The opcode for this command is 09 Hex. If a CAMAC control operation is specified in the CAMAC NAF, there is no data phase. If a CAMAC write or read operation is specified, data transfer will follow the Command phase. The number of bytes transferred during the data phase depends on the CAMAC data word size specification in the Mode Control byte of the CDB. 8-bit transfers require one byte, 16-bit requires two bytes and 24-bit requires four bytes.

After the CAMAC operation is executed by the 3929, a Status byte is sent. Either a GOOD status or CHECK CONDITION is returned. CHECK CONDITION is sent if the CAMAC operation was terminated due to an error. This error is dependent on such parameters as the CAMAC Q-response, CAMAC X-response, etc. Refer to the CAMAC Operating Modes section of this manual for further details.

The following table shows the Command Descriptor Block for the Single CAMAC Operation command:

TABLE 19 - Single CAMAC Command Descriptor Block

Bit / Byte	7	6	5	4	3	2	1	0
0	OPCODE = 09 HEX							
1	LUN = 0				RESERVED = 0			
2	MODE CONTROL							
3	NAF HI							
4	NAF LO							
5	CONTROL BYTE							

Byte 0: This byte is set to 09 Hex to indicate a SINGLE operation.

Byte 1: This contains a Logical Unit field and a reserved field. This byte MUST be set to zero.

Byte 2: This byte is used to specify the Mode Control for the CAMAC operation.

Bytes 3 and
Bytes 4:
These bytes are used to specify the CAMAC Station Number (N), Subaddress (A), and Function Code (F).

Byte 5: This is the Control Byte and must be set to zero.

BLOCK TRANSFER CAMAC OPERATION Command

The BLOCK Transfer CAMAC operation command is used for executing block transfer operations to the CAMAC Dataway. The opcode for this command is 22 Hex. After the 3929 receives the command descriptor block, the Data phase is entered. Data transfer direction depends on the CAMAC Function Code selected. Once data transfers begin, they continue until an error condition occurs or the Transfer Count has been exhausted.
After the Block Transfer operation has been completed by the 3929, a Status byte is sent. Either a GOOD or CHECK CONDITION status is returned. CHECK CONDITION is returned if the CAMAC Block Transfer operation was terminated due to an error. This error is dependent on such parameters as the CAMAC Q-response, X-response, etc. Refer to the CAMAC Operating Modes section of this manual for additional information.

Model 3929-Z1B

The following table shows the command descriptor block for the BLOCK Transfer CAMAC Operation SCSI command:

TABLE 20 - BLOCK Transfer CAMAC Operation Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 22 HEX							
1	LUN = 0			RESERVED = 0				
2	MODE CONTROL							
3	NAF HI							
4	NAF LO							
5	TRANSFER BYTE COUNT HI							
6	TRANSFER BYTE COUNT MID							
7	TRANSFER BYTE COUNT LO							
8	RESERVED = 0							
9	CONTROL BYTE							

Byte 0: This byte is set to 22 hex to indicate a BLOCK operation.

Byte 1: This byte contains a Logical Unit field and a reserved field. This byte MUST be set to zero.

Byte 2: This byte is used to specify the Mode Control for the operation.

Bytes 3 and
Bytes 4:
These bytes are used to specify the CAMAC Station Number (N), Subaddress (A), and Function Code (F) to be used for the Block Transfer operation.

Bytes 5-7:
These bytes are used to specify the number of data bytes to be sent or received during a Block Transfer operation. Byte 5 is the most significant byte and byte 7 is the least significant byte. This data is loaded into the SCSI Bus Interface Controller (SBIC) for specifying the total number of data bytes to be transferred. The two's complement of this number is loaded into the 3929 internal byte count register which controls the number of CAMAC Dataway cycles executed.

For example, if a CAMAC Block Transfer of 260 16-Bit words was desired, byte 5 is set to zero, byte 6 is set 2, and byte 7 is set to 8. This would generate 520 byte transfers on the SCSI bus for the requested operation (260 X 2).

Byte 8: This is a Reserved byte and MUST be set to zero.

Byte 9: This is the Control Byte and MUST be set to zero.

BOOK LAM Command

The BOOK LAM command is used to enable the generation of an Asynchronous Event Notification (AEN) when a selected LAM occurs. This command enables the LAM MASK register for the specified LAM and specifies two CAMAC control NAFs that are executed on the occurrence of the LAM.

The BOOK LAM command has no data phase associated with it. All necessary information is contained in the Command Descriptor Block. After the 3929 receives this command, it enters the status phase. The 3929 returns status of either CHECK CONDITION or GOOD. After the status phase, the 3929 sends the Command Complete Message. The following table shows the Command Descriptor Block for the BOOK LAM command.

TABLE 21 - BLOCK Transfer CAMAC Operation Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = A0 HEX							
1	LUN = 0			RESERVED = 0				
2	LAM IDENTIFICATION							
3	LAM TYPE							
4	USER DEFINED FIELD 1							
5	USER DEFINED FIELD 2							
6	USER DEFINED FIELD 3							
7	CLEAR COMMAND NAF HIGH							
8	CLEAR COMMAND NAF LOW							
9	DISABLE COMMAND NAF HIGH							
10	DISABLE COMMAND NAF LOW							
11	CONTROL BYTE							

Byte 0: This byte is set for A0 Hex to indicate a BOOK LAM command.

Byte 1: This byte contains a logical unit field and a reserved field.

Byte 2: This byte specifies the identification of the LAM to be booked. The identification refers to the station number from within the crate of the LAM to be booked. The identification number ranges from LAM 1 through LAM 24.

Byte 3: This byte specifies the type of LAM being booked. This can be either type 0 or 1. When a type 0 LAM occurs, both the CLEAR and DISABLE NAF specifications are executed. When a type 1 LAM occurs, only the CLEAR NAF specification is executed.

Byte 4 through 6:
These bytes are user-defined bytes. These information bytes are returned to the SCSI initiator during the Asynchronous Event Notification.

Bytes 7 and 8:
These bytes are used for specifying the CAMAC CLEAR NAF to be executed when the selected LAM occurs.

Bytes 9 and 10:
These bytes are used for specifying the CAMAC DISABLE NAF to be executed when the selected LAM occurs.

Byte 11:
This is the Control Byte and **MUST** be set to zero.

Please refer to the Asynchronous Event Notification section of this manual for further information.

UNBOOK LAM COMMAND

The UNBOOK LAM command is used to prevent a specific LAM from generating an Asynchronous Event Notification (AEN). After a LAM is booked using the BOOK LAM command, it may be disabled (unbooked) by using the UNBOOK LAM command. This command disables the LAM MASK register for the specified LAM.

The UNBOOK LAM command has no data phase associated with it. All necessary information is contained in the Command Descriptor Block. After the 3929 receives the command, it enters the status phase. The 3929 returns status of either CHECK CONDITION or GOOD. After the status phase completes, the Command Complete Message is sent. The following table shows the Command Descriptor Block for the UNBOOK LAM command.

TABLE 22 - Command Descriptor Block for UNBOOK LAM Command

BIT / BYTE	7	6	5	4	3	2	1	0
0	OPCODE = 06 Hex							
1	LUN = 0				RESERVED = 0			
2	LAM IDENTIFICATION							
3	RESERVED = 0							
4	RESERVED = 0							
5	CONTROL BYTE							

BYTE 0: This byte is set to 06 Hex to indicate an UNBOOK LAM command.

BYTE 1: This byte contains a logical unit field and a reserved field. This byte **MUST** be set to zero.

BYTE 2: This byte specifies the identification of the LAM to be unbooked. The identification refers to the station number within the crate of the LAM to be unbooked. The identification numbers range from LAM1 through LAM24.

BYTES 3-4:
These are reserved bytes and **MUST** be set to zero.

BYTE 5: This is the Control Byte and **MUST** be set to zero.

REGISTER ACCESS COMMAND

The REGISTER ACCESS command allows access to all the registers on the 3929. The registers are accessed via the microprocessor. The microprocessors PROM and internal RAM cannot be accessed with this command. This command can be used to either write or read a byte of data to one of the internal registers. Each command transfers only one byte of data. This command is used by the manufacturer for diagnostic purposes only. This command is normally not executed by the user. The command is shown here for completeness only.

After the 3929 receives the Command Descriptor Block, it obtains the address of the data to be accessed by reading Bytes 2 and 3 of the CDB. The 3929 then reads Byte 4 to determine the direction of the transfer. If Byte 4 is set to zero, the addressed register is written with the data obtained from the subsequent DATA OUT phase. If Byte 4 is set to one, the addressed register is read and the data is sent to the initiator with a subsequent DATA IN phase. After the DATA phase has been executed, the 3929 returns the Completion Status Byte followed by a COMMAND COMPLETE message. The following table shows the Command Descriptor Block for the REGISTER ACCESS Command:

TABLE 23 - REGISTER ACCESS Command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPCODE = 0D HEX							
1	LUN = 0			RESERVED = 0				
2	ADDRESS HIGH							
3	ADDRESS LOW							
4	RESERVED = 0							R/W
5	CONTROL BYTE							

Byte 0: This byte is set to 0D Hex to indicate a REGISTER ACCESS Command.

Byte 1: This byte contains a Logical Unit and Reserved field. This byte MUST be set to zero.

Byte 2: This byte is used for specifying the most significant eight address bits for the data access. This is used in conjunction with Byte 3 for determining the 16-bit address to be accessed.

Byte 3: This byte is used for specifying the least significant eight address bits for the data access. This is used in conjunction with Byte 2 for determining the 16-bit address to be accessed.

Byte 4: This byte contains a Reserved field and a READ/WRITE bit. The Reserved field MUST be set to zero. For reading a 3929 register, the R/W bit must be set to one. A zero indicates the register is to be written.

Byte 5: This is the Control Byte and MUST be set to zero.

LOAD LIST Command

The LOAD LIST command is used for loading the List Processing Memory within the 3929. The List Processing Memory is a 32K x 16 static RAM. The address range for this memory is from 0 to FFFF hex. The memory is used for storing the list(s) to be executed. Refer to the List Processing section of this manual for further details.

The following table shows the Command Descriptor Block for the LOAD LIST command.

TABLE 24 - LOAD LIST Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 23 HEX							
1	LUN = 0				RESERVED = 0			
2	LIST ADDRESS HI							
3	LIST ADDRESS LO							
4	TRANSFER BYTE COUNT HI							
5	TRANSFER BYTE COUNT MID							
6	TRANSFER BYTE COUNT LO							
7	RESERVED							
8	RESERVED							
9	CONTROL BYTE							

Byte 0: This byte is set to 23 hex to indicate a LOAD LIST command.

Byte 1: This byte contains a Logical Unit field and a reserved field. This byte MUST be set to zero.

Bytes 2 and
Bytes 3:

These bytes specify the starting address for the list load operation. The allowable range for this address is from 0 to FFFF hex. Byte 2 is the most significant address byte and byte 3 is the least significant.

Bytes 4-6:

These bytes are used to specify the total number of bytes to be sent to the 3929 during the LOAD LIST operation. Byte 4 is the most significant byte and byte 6 is the least significant. The maximum number of transfers is FFFF hex. For example, if it was desired to transfer 576 bytes to the 3929, byte 4 is set to zero, byte 5 is set to 2 and byte 6 is set to 40 hex.

Bytes 7 and Bytes 8:

These are reserved bytes and MUST be set to zero.

Byte 9: This is the Control Byte and MUST be set to zero.

EXECUTE LIST Command

The EXECUTE LIST command is used for initiating the execution of a previously loaded list. The opcode for this command is 20 hex. This command instructs the 3929 to execute the list at the specified starting address. The command also includes the transfer count specification which determines the total number of words to be transferred for the entire list. Individual Block Transfers within the list have their own transfer count specification embedded in the list. Refer to the List Processing section of this manual for further details.

The following table shows the Command Descriptor Block for the EXECUTE LIST command:

TABLE 25 - EXECUTE LIST Command Descriptor Block

Bit / Byte	7	6	5	4	3	2	1	0
0	OPCODE = 20 HEX							
1	LUN = 0			RESERVED = 0				
2	LIST ADDRESS HI							
3	LIST ADDRESS LO							
4	TRANSFER BYTE COUNT HI							
5	TRANSFER BYTE COUNT MID							
6	TRANSFER BYTE COUNT LO							
7	RESERVED = 0							R/W
8	RESERVED = 0							
9	CONTROL BYTE							

Byte 0: This byte is set to 20 hex to indicate an EXECUTE LIST command.

Byte 1: This byte contains a Logical Unit field and a reserved field. This byte MUST be set to zero.

Bytes 2 and
Bytes 3:

These bytes specify the starting address for the execute list operation. The allowable range for this address is from 0 to FFFF hex. Any specification outside this range results in a CHECK CONDITION with the Sense Key set to Illegal Request, the Sense Code set to 81 hex, and the Sense Code Qualifier set to 01. Byte 2 is the most significant address byte and byte 3 is the least significant.

Bytes
4-6:

These bytes specify the total number of bytes to be transferred to or from the 3929 during list execution. This number is the total transfer length and includes any single transfer and block transfer specifications in the list. Byte 4 is the most significant byte and byte 6 is the least significant. For example, if the list entries require 1056 bytes to be transferred, byte 4 is set to zero, byte 5 is set to 4, and byte 6 is set to 20 hex.

Byte 7: This byte contains a reserved field and a READ/WRITE bit. The reserved field must be set to zero. The READ/WRITE bit must be set to reflect the direction of the SCSI data transfers during the list operation. If the list requires data from the SCSI bus to be used in CAMAC write operations, the R/W bit is set to zero. If the list executes CAMAC read operations, the R/W bit must be set to one.

Byte 8: This byte is reserved and must be set to zero.

Byte 9: This is the Control Byte and MUST be set to zero.

RESUME LIST Command

The RESUME LIST command is used to resuming list execution after an error termination. The 3929 uses the last internal list address for the continuation. The word count specification is already stored in the 3929. Refer to the List Processing section of this manual for further details.

The opcode for this command is 0E hex. The following table shows the Command Descriptor Block for the RESUME LIST command.

TABLE 26 - RESUME LIST Command Descriptor Block

Bit Byte	7	6	5	4	3	2	1	0
0	OPCODE = 0E HEX							
1	LUN = 0			RESERVED = 0				
2	RESERVED = 0							
3	RESERVED = 0							
4	RESERVED = 0							
5	CONTROL BYTE							

Model 3929-Z1B

Byte 0: This byte is set to 0E hex to indicate a RESUME LIST command.

Byte 1: This byte contains a Logical Unit field and a reserved field. This byte MUST be set to zero.

Bytes
2-4: These bytes are reserved and MUST be set to zero.

Byte 5: This is the Control Byte and MUST be set to zero.

CAMAC Operating Modes

The 3929 may execute CAMAC operations in either Single Transfer of Block Transfer fashion. During Single Transfer operations, only one CAMAC data word is transferred for each SCSI command executed. During Block Transfer operations, multiple CAMAC data words are transferred for each SCSI command executed.

Single Transfer operations are executed using the SINGLE SCSI command. Block Transfer operations can be executed using the BLOCK SCSI command or by using the SETUP SCSI command in conjunction with either a SEND or RECEIVE SCSI command.

CAMAC NAF Specifications

Before any addressed CAMAC operation is executed, the CAMAC Station Number (N), Subaddress (A) and Function Code (F) must be specified. This specification is split into two bytes. The following table shows the bits defined in each of the NAF bytes. The bytes are sent to the 3929 in the order shown.

TABLE 27 - NAF High Byte

7	6	5	4	3	2	1	0
0	0	N 16	N 8	N 4	N 2	N 1	A 8

TABLE 28 - NAF Low Byte

7	6	5	4	3	2	1	0
A 4	A 2	A 1	F 16	F 8	F 4	F 2	F 1

The CAMAC Station Number bits, N16 through N1, are used for specifying the CAMAC Station Number to be used during the addressed CAMAC operations. These 5 bits yield a Station Number range from 0 to 31. Station Numbers 0, 24 through 29, and 31 are not used. Station Number 30 is a pseudo-address and is used to address the internal registers of the

3929. Please refer to the Single Transfer N(30) Commands section of this manual for further details.

The CAMAC Subaddress bits, A8 through A1, are used to specify the CAMAC Subaddress to be used during an addressed CAMAC operation. These 4 bits yield a subaddress range from 0 to 15.

The CAMAC Function Code bits, F16 through F1, are used to specify the CAMAC Function Code to be used during an addressed CAMAC operation. These 5 bits yield a Function Code range from 0 to 31. The binary combination of the F16 and F8 bits determine the type of CAMAC operation to be performed as shown below:

F16	F8	Operation
0	0	CAMAC READ
0	1	CAMAC CONTROL
1	0	CAMAC WRITE
1	1	CAMAC CONTROL

The NAF specification for the SCSI commands that execute CAMAC operations is found in bytes 4 and 5 of the Command Descriptor Block (CDB). The NAF High is byte 4 and the NAF Low is byte 5. Refer to the individual SCSI Command Descriptions for additional information.

Executing Single Transfer Operations

Single Transfer CAMAC operations are executed using the SINGLE SCSI command. The Mode Control byte of the Command Descriptor Block (CDB) for the SINGLE SCSI command has the following table.

TABLE 29 - SINGLE SCSI Command Mode Control Byte

7	6	5	4	3	2	1	0
0	0	0	0	TM1	WS2	WS1	AD

Bits 7 through 4 must be set to zero. These bits are used to define Block Transfer operations. If these bits are non-zero, the operation terminates with a CHECK CONDITION.

Bit 3 is used to select the desired Transfer Mode. The Q-Stop mode is selected by setting this bit to a zero. The Q-Ignore mode is selected when this bit is set to a one. The difference between these two modes is the Status byte returned when the 3929 enters the Status phase after the command is executed. If the Q-Ignore mode was selected, the Q-response is ignored in the generation of the status response. If the Q-Stop mode was selected, and the CAMAC operation resulted in a CAMAC Q-response of zero, the 3929 returns a Status byte set to CHECK CONDITION.

Bits 2 and 1 are set to the desired CAMAC Data Word Size. If the CAMAC function code specified in the CDB is a CAMAC control function, these bits are ignored. The following chart shows the CAMAC data word sizes obtained for the various binary combinations of these bits:

WS2	WS1	CAMAC Data Word Size
0	0	24-Bit
0	1	16-Bit
1	0	8-Bit
1	1	Reserved

Bit 0 is used to enable/disable the generation of CHECK CONDITION based on the CAMAC X-Response. If this bit is set to a zero, a CAMAC X-response of zero causes a status response of CHECK CONDITION. If this bit is set to a one, a CHECK CONDITION status is not returned due to a CAMAC X-Response of zero.

After the 3929 receives the SINGLE SCSI command, it first examines the Mode Control byte. If bits 7 through 4 are not zero, the 3929 switches to the status phase and returns a CHECK CONDITION with the Sense Key set to 5 (Illegal Request). The Sense Code is set to 80 hex and the Sense Code Qualifier is set to 2. After the status is sent, the 3929 then sends the COMMAND COMPLETE message. If the Mode Control byte is acceptable, it is then loaded into the 3929 internal Mode Control Register.

Before continuing, the 3929 reads the on-board CAMAC Control/Status Register to determine the status of the front-panel On-Line/Off-Line switch. If the switch is in the On-Line position, processing continues. If the switch is in the Off-Line position, the 3929 then switches to the Status phase and sends the CHECK CONDITION status. The Sense Key is set to 2 to indicate that the device is not ready. The Sense Code is set to 4 and the Sense Code Qualifier is set to 3. After the status is sent, the 3929 sends the COMMAND COMPLETE message.

The CAMAC function code specified in the CDB is examined and loaded into the 3929 CAMAC Command Register (NAF). If the CAMAC function code specified a CAMAC Control operation, no SCSI data phase is entered. After the CAMAC control function is detected, the 3929 executes the requested cycle. Once the cycle is complete, the 3929 enters the Status phase and returns the appropriate status. After the status is sent, the 3929 then sends the COMMAND COMPLETE message.

If the CAMAC function code specified a CAMAC write operation, the 3929 switches to the Data In phase. The 3929 expects to receive either one, two or four bytes of data before executing the CAMAC cycle. The byte count depends on the CAMAC Data Word Size specified. 24-Bit cycles require four bytes, 16-Bit cycles require two bytes, and 8-bit cycles require one byte. Data is transferred to the 3929 as low byte first. Therefore, if the 24-bit Data Word Size was selected, the data transfer would be CAMAC Write Data bits 1-8 first, 9-16 second, and 17-24 third and a zero byte for the fourth. Once the 3929 has received the correct number of bytes, it executes the CAMAC cycle. After the cycle is complete, the Status phase is entered and the appropriate status is returned. After the status is sent, the 3929 sends the COMMAND COMPLETE message.

If the CAMAC function code specified a CAMAC read operation, the 3929 executes the cycle as soon as the CAMAC command has been loaded. Once the cycle is complete, the 3929 enters

the Data Out phase and returns the read data obtained from the CAMAC cycle. The number of bytes returned depends on the CAMAC Data Word Size selected. Four bytes are returned for 24-bit operations, two bytes are returned for 16-bit operations, and one byte is returned for 8-bit operations. CAMAC Read Data is returned low-byte first, as in the case of CAMAC Write Data. Once the CAMAC Read Data has been sent, the Status phase is entered and the appropriate status is returned. After the status is sent, the 3929 sends the COMMAND COMPLETE message.

The Status byte returned depends on several parameters. These include the selections of the AD (Abort Disable) bit, the TM1 (Transfer Mode) bit, and the CAMAC Q and X responses obtained for the CAMAC operation. If the CAMAC operation resulted in Q and X responses of one, a status of GOOD is returned. If the CAMAC operation resulted in an error, the CHECK CONDITION status is returned with the Sense Key set to 0B hex, the Sense Code set to 80 hex, and the Sense Code Qualifier set to 1.

The following equation describes the definition of an error for the single CAMAC operations.

$$\text{ERROR} = \overline{X} * \overline{AD}$$

$$\# \overline{Q} * \overline{TM1}$$

Single Transfer N(30) Commands

Along with accessing other modules in the crate, the single transfer commands may also be used to access the on-board registers of the 3929. These registers include the LAM Register, LAM Mask Register, and CAMAC Control/Status Register.

These registers are accessed by using the pseudo-address of Station Number 30. These accesses to the 3929 do not perform any CAMAC Dataway cycles, but access the internal registers of the 3929. The following is a list of the valid CAMAC commands used to access the internal registers.

Command	Q	Description
F(1)A(0)	On-Line	Reads the CAMAC Control/Status Register
F(1)A(12)	On-Line	Reads the LAM Pattern
F(1)A(13)	On-Line	Reads the LAM Mask Register
F(17)A(0)	On-Line	Writes the CAMAC Control/Status Register
F(17)A(13)	On-Line	Writes the LAM Mask Register

Notes:

1. N=30 for all commands.

2. X = 1 for all valid addressed commands.

3. A Q=0 response is obtained for if the front-panel On-Line/Off-Line switch is in the Off-Line position. This response is used along with the Transfer Mode (TM1) bit to determine the Status byte to return. The only command that returns correct data when the 3929 is Off-Line is the F(1)A(0) command.

These N(30) operations are executed in the same manner as the normal CAMAC Dataway operation commands. The status response returned is also based on the CAMAC Q and X responses, along with the Transfer Mode (TM1) selected. The CAMAC Data Word Size bits are also used while accessing the N(30) registers. The following describes the commands used to access the 3929 on-board registers.

CAMAC Control/Status Register

The CAMAC Control/Status Register is a 16-bit write/read register. This register is written with an F(17)A(0) command and read with an F(1)A(0) command. This register may be accessed using any of the CAMAC Data Word Size selections. If an 8-bit write operation is selected, only the lower 8-bits, bits 1 through 8, are updated with the specified write data. If an 8-bit read is executed, only the lower 8-bits of the CAMAC Control/Status Register are returned. The high-byte of this register, bits 17 through 24, is always written and read as zero.

The following shows the bits defined for the CAMAC Control/Status Register. Since this register contains write/read, read-only, and write-only bits, two patterns are shown.

Bit	Write Operation	Read Operation
1	Generate CAMAC Initialize Cycle (Z)	0
2	Generate CAMAC Clear Cycle (C)	0
3	Set Inhibit	Read Inhibit
4	0	0
5	0	0
6	0	0
7	0	Read Dataway Inhibit
8	0	0
9	Enable Service Request	Service Request Enabled
10	Set Internal LAM 24	Internal LAM 24 Set
11	0	0
12	0	0
13	0	0
14	0	Front-Panel Switch Off-Line
15	0	0
16	0	Selected LAM Present
17-24	0	0

Bit **Description**

1 By writing this bit to a one with the 3929 On-Line, the crate controller executes a CAMAC Initialize (Z) cycle. This bit is always read as a zero.

2 By writing this bit to a one with the 3929 On-Line, the crate controller executes a CAMAC Clear (C) cycle. This bit is always read as a zero.

3 By writing this bit to a one, the 3929 asserts the Dataway Inhibit line. With the 3929 asserting the Inhibit line, both bits 3 and 7 are returned as ones when the Control/Status Register is read.

4-6 These bits are not used and read as zeros.

7 This read-only bit indicates the state of the Inhibit line on the CAMAC Dataway. The assertion of this bit does not necessarily indicate that the 3929 is asserting its Inhibit line; it may be another module in the crate asserting Inhibit.

8 This bit is not used and read as a zero.

9 By setting this bit to a one, a Service Request is made to the interface whenever a module Look-At-Me (LAM) is pending AND that LAM is enabled in the LAM Mask Register.

10 This is a write/read bit used to set or clear the Internal LAM 24. Writing this bit to a one asserts the LAM 24 signal. Writing this bit to a zero negates the LAM 24 signal. The LAM 24 signal can be used for hardware and software testing associated with the Service Request.

11-13
These bits are not used and read as zeros.

14 This read-only bit indicates the state of the front-panel On-Line/Off-Line switch. This bit is returned as a one when the switch is in the Off-Line position. A zero is returned for this bit if the switch is in the On-Line position.

15 This bit is not used and read as a zero.

16 This is a read-only bit which indicates the status of the Selected-LAM-Present (SLP) signal. This bit is read as a one when one or more LAM requests are asserted AND the LAM Mask bit(s) corresponding to the LAM request is true. This bit is read as a zero as long as no Selected LAMs are present.

17-24
These bits are not used and read as zeros.

LAM Register

The LAM Register is a 24-bit read-only register which indicates the current state of all LAM requests in the CAMAC crate. The LAM Register is read with a F(1) A(12) command. Each bit that corresponds to the appropriate CAMAC Station Number in the crate (LAM(9) is associated with N(9), etc.).

This register may be read using any of the three CAMAC Data Word Size formats. If the 8-bit format is selected, only the lower 8 LAMs, L(1) through L(8), are returned. The 16-bit format returns LAMs 1-16 and the 24-bit format returns LAMs 1 through 24.

LAM Mask Register

The LAM Mask Register is a 24-bit write/read register used to enable/disable the generation of a Selected-LAM-Present (SLP) on the assertion of predetermined LAM requests. The LAM Mask Register is written using the F(17)A(13) command and read using the F(1) A(13) command. Each bit in the LAM Mask Register has a one-to-one correspondence to the LAM request bits (Mask bit 6 corresponds to LAM request bit 6). A LAM request is enabled to generate a Selected-LAM-Present by writing the corresponding bit to a one. The LAM request is disabled by writing the bit to a zero. This register is cleared on power-up.

Executing Block Transfer Operations

Block Transfer operations are used to transfer multiple CAMAC data words to/from an addressed CAMAC module. The Block Transfer operations can be executed by one of two methods. One method is by using the BLOCK SCSI command. The second method involves the use of the SETUP SCSI command in combination with either the SEND or RECEIVE SCSI commands. Whichever method is used, three basic parameters are loaded into the 3929 before the Block Transfer occurs. These parameters are the CAMAC NAF (Station Number, Subaddress, and Function Code), the Mode Control register data, and the transfer count. By using the SCSI BLOCK command, all of the required parameters are passed in the Command Descriptor Block. When using the second method, the CAMAC NAF and Mode Control data are specified in the SCSI SETUP Command Descriptor Block and the transfer count accompanies the SEND or RECEIVE SCSI command.

The CAMAC NAF is used to determine the CAMAC Station Number, Subaddress and Function Code to be executed during the Block Transfer operation. CAMAC Control commands are not allowed using the Block Transfer commands. If a CAMAC Control operation is specified, a CHECK CONDITION status is returned.

The Mode Control byte for the Block Transfer commands is used to specify the type of Transfer Mode, the CAMAC Data Word Size, and several other parameters concerning the Block Transfer operation. The following table shows the Mode Control byte for the Block Transfer commands. The Transfer Count is used to specify the total number of bytes to be transferred to or from the 3929 during the Data phase of the Block Transfer operation.

Block Transfer Mode Control

TABLE 30 - Block Transfer Mode Control Byte

7	6	5	4	3	2	1	0
0	FAST	1	TM2	TM1	WS2	WS1	AD

Model 3929-Z1B

Bit 7 is not used and must be set to zero.

Bit 6 is used to enable/disable the FAST Dataway access feature of the 3929. This feature is enabled by writing this bit to a one and disabled by writing it to a zero. Refer to the Fast Block Transfer section of this manual for further details.

Bit 5 must be set to a one for Block Transfer operations.

Bits 4 and 3 are used to specify the Transfer Mode to be used for the Block Transfer operation. The binary combination of these bits selects the various Transfer Modes as follows:

TM2	TM1	Transfer Mode
0	0	Q-Stop
0	1	Q-Ignore
1	0	Q-Repeat
1	1	Q-Scan

Bits 2 and 1 are used to specify the CAMAC Data Words Size to be used for the Block Transfer operation. The binary combination of these bits determine the CAMAC Data Word Size as follows:

WS2	WS1	CAMAC Data Word Size
0	0	24-Bit Data Words
0	1	16-Bit Data Words
1	0	8-Bit Data Words
1	1	Reserved

Bit 0 is used to enable/disable the termination of a Block Transfer operation on the occurrence of a CAMAC X-response of zero. Setting this bit to a one disables the termination of a Block Transfer operation due to a X=0 condition. The X=0 termination is enabled by setting this bit to a zero.

After the 3929 receives a SCSI command instructing it to execute a Block Transfer operation, it first examines the CAMAC Function Code. If the function specified is a CAMAC Control operation, the 3929 enters the Status phase and returns a CHECK CONDITION with the Sense Key set to 6, the Sense Code set to 80 hex, and the Sense Code Qualifier set to 1. If the CAMAC function is a read or write, it loads the internal NAF register with the specified data. The 3929 loads the SCSI Bus Interface Controller (SBIC) and internal CAMAC transfer count registers with the data passed in the CDB. Once the registers have been loaded, the internal CAMAC GO bit is set which initiates the transfer.

If the CAMAC Function Code specified was a CAMAC read operation, the CAMAC state machine starts executing CAMAC operations immediately. Once the dataway cycle is complete, the data is checked for validity. Data validity depends on such parameters as the Transfer Mode selected (TM2,TM1) and the CAMAC Q and X responses. Refer to the individual Transfer Mode sections for further information regarding data validity. Once valid data is received, it is stored into a First-In-First-Out (FIFO) memory. The CAMAC state machine continues to execute dataway cycles until the transfer count is exhausted or an error condition

occurs. A second state machine, the FIFO state machine, reads the data from the FIFO and transfers it to the SBIC. The SBIC then sends the data out onto the SCSI bus. The FIFO state machine continues this data transfer until the FIFO no longer contains data AND the transfer count has been exhausted. If an error condition occurs, the CAMAC state machine ceases to execute CAMAC cycles. The FIFO state machine stops sending data to the SBIC when all valid CAMAC words stored in the FIFO have been sent to the SBIC. After all data has been transferred on the SCSI bus, the 3929 enters the Status phase and returns a CHECK CONDITION and then the COMMAND COMPLETE message. After the error, the Sense Key is set to 0B hex, the Sense Code is set to 80 hex, and the Sense Code Qualifier is set to 1. If all requested CAMAC cycles are executed successfully, the 3929 returns a GOOD status.

If the CAMAC Function Code specified was a CAMAC write operation, the SBIC is instructed to start obtaining CAMAC write data from the SCSI bus. Once a byte of data has been received from the SBIC, the FIFO state machine transfers the data to the FIFO. The SBIC and FIFO state machine continue to transfer data in this manner until the transfer count is exhausted or an error condition occurs. If the FIFO becomes more than half full during the Block Transfer operation, the SBIC stops transferring data from the SCSI bus. SCSI transfers are resumed once the FIFO becomes less than half full. While the FIFO state machine is transferring data, the CAMAC state machine is busy executing dataway cycles. The CAMAC state machine continues executing dataway operations as long as there is CAMAC write data available and no errors have occurred. If an error occurs, both the CAMAC and FIFO state machines are shut off. The 3929 terminates the Data phase and switches to the Status phase. The CHECK CONDITION status is returned with the Sense Key set to 0B hex, the Sense Code set to 80 hex, and the Sense Code Qualifier set to 1. If all the requested CAMAC operations are successfully executed, the 3929 returns a GOOD status.

Q-Stop Block Transfer Operations

To select the Q-Stop Transfer Mode, set both the TM2 and TM1 bits to zeros when loading the Mode Control byte in the CDB. During Q-Stop Block Transfer operations, the CAMAC command specified is repeated until the transfer count is exhausted or a Q response of zero is received. The transfer also terminates when an error condition occurs. The following equation describes the ERROR for Q-Stop Block Transfers:

$$\text{ERROR} = \overline{Q} + \overline{AD} * \overline{X}$$

Figure 2 is a simplified flow diagram for Q-Ignore Block Transfer reads:

Q-Ignore Block Transfer Operations

To select the Q-Ignore Transfer Mode, set the Transfer Mode bit TM2 to a zero and the TM1 bit to a one when loading the Mode Control byte in the CDB. During Q-Ignore Block Transfers, the CAMAC command specified is repeated until the transfer count is exhausted. The transfer also terminates if an error is encountered. The following equation describes the ERROR for Q-Ignore Block Transfers:

$$\text{ERROR} = \overline{\text{AD}} * \overline{\text{X}}$$

Figure 3 is a simplified flow diagram for Q-Ignore Block Transfer writes.

Q-Repeat Block Transfer Operations

To select the Q-Repeat Transfer Mode, set the TM2 bit to a one and the TM1 bit to a zero when loading the Mode Control byte in the CDB. During a Q-Repeat Block Transfer operation, the CAMAC command specified is repeated for each data word until a Q response of one is obtained. A Q response of one causes either new write data to be retrieved or read data to be stored. The command is repeated for each data word until the transfer count is exhausted. If a Q response of one is not received for a data word within 200 milliseconds, the transfer terminates and CHECK CONDITION is returned. This Q-Repeat timeout feature can be disabled by a strap located on the 3929 B-board. Refer to the Strap Options section of this manual for further information. The Block Transfer also terminates if an error is encountered. The following equation describes ERROR for Q-Repeat Block Transfer operations:

$$\text{ERROR} = \text{Q-TIMEOUT} + \overline{\text{AD}} * \overline{\text{X}}$$

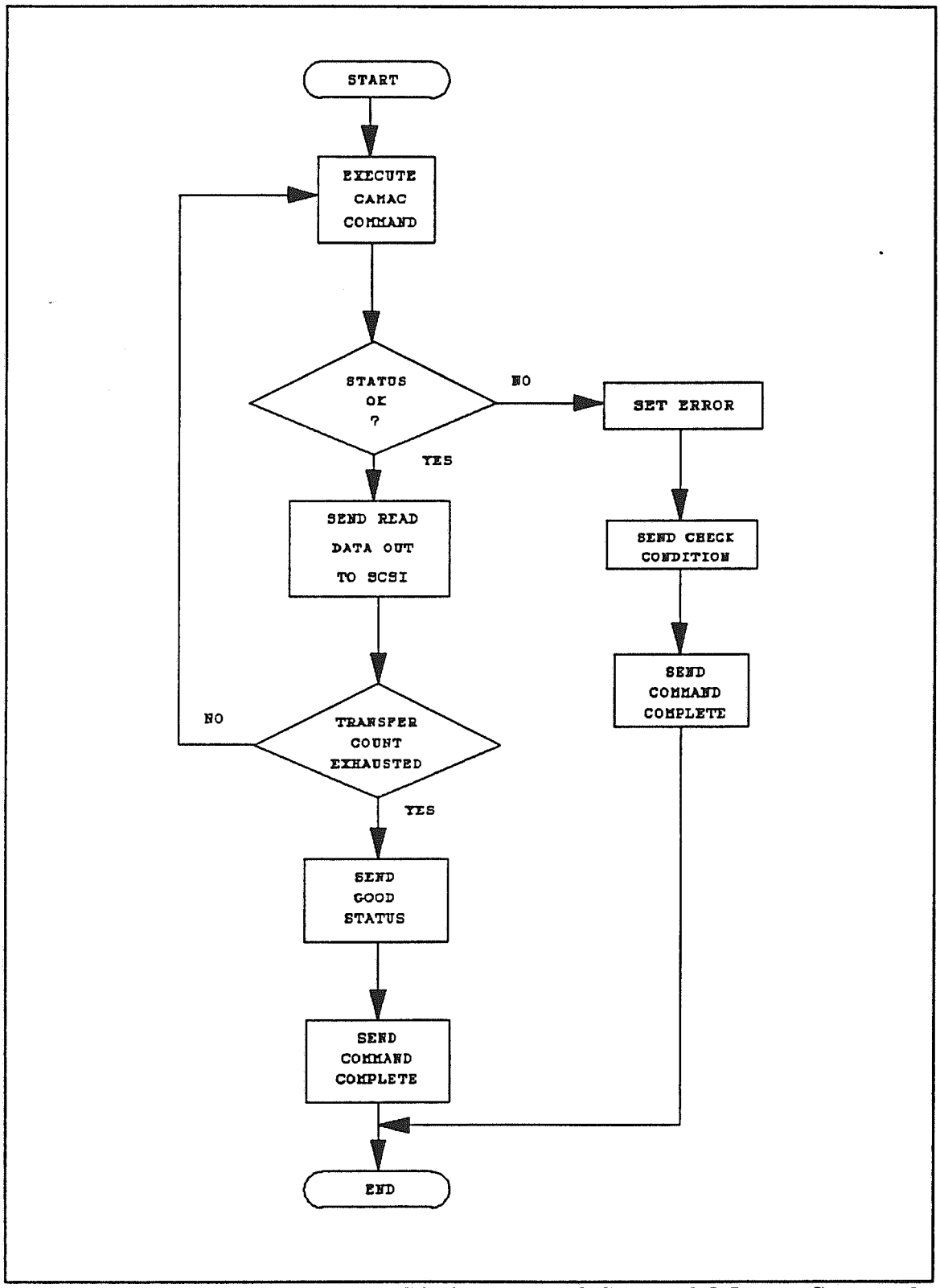


Figure 2 - Flow Diagram for 3929 Block Transfer Q-Stop and Q-Ignore Commands

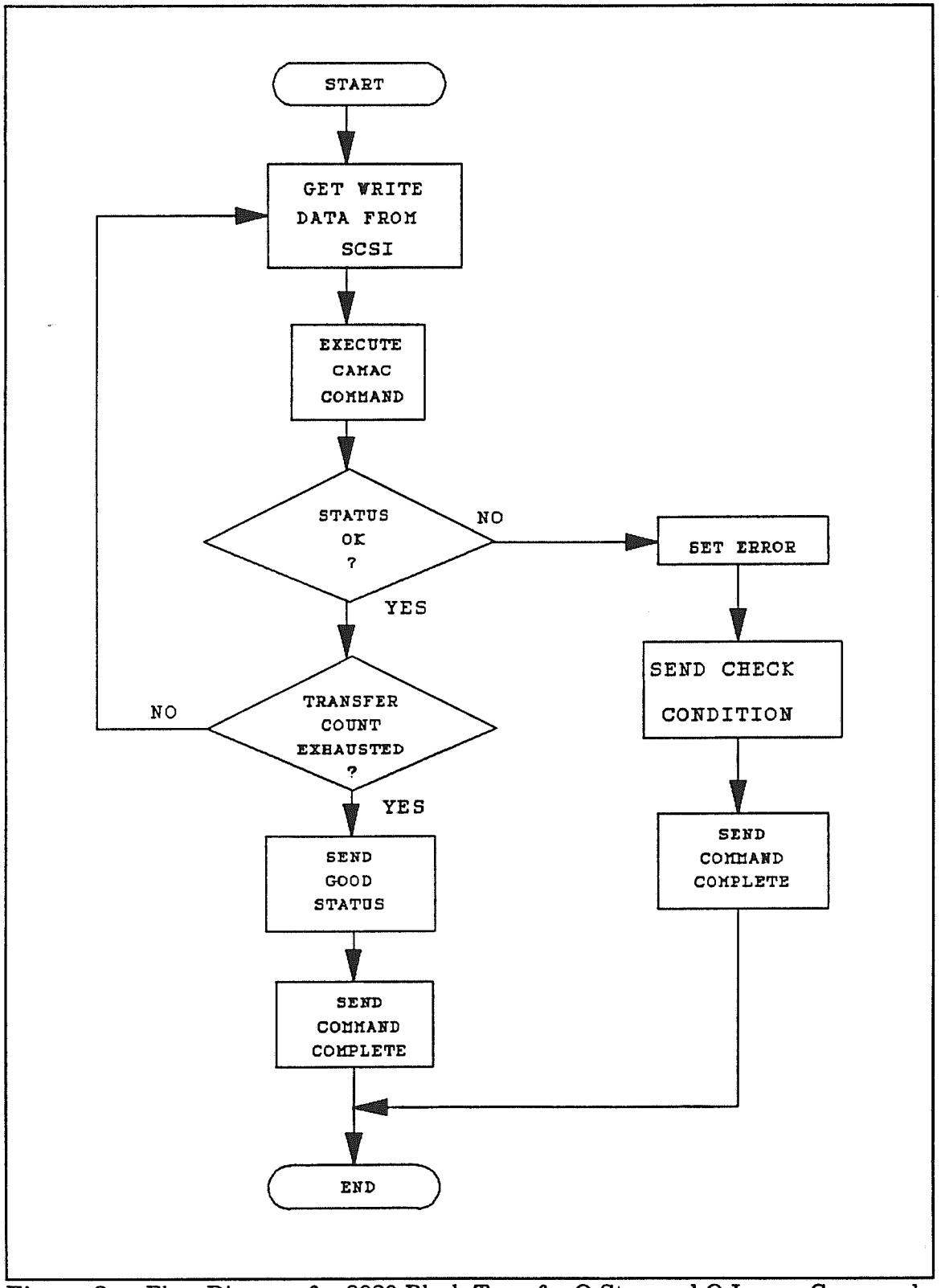


Figure 3 - Flow Diagram for 3929 Block Transfer Q-Stop and Q-Ignore Commands

Model 3929-Z1B

Figures 4 and 5 are simplified flow diagrams for Q-Repeat reads and writes:

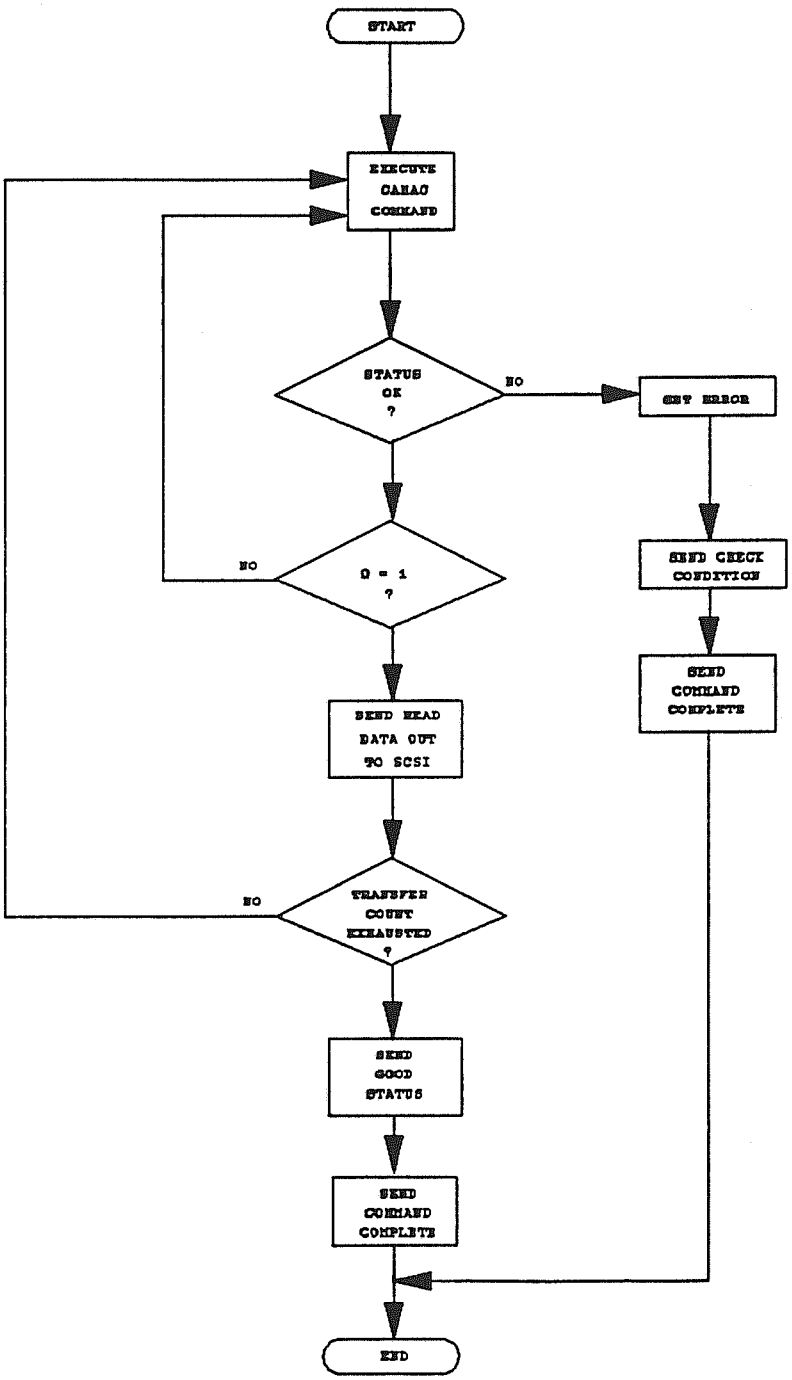


Figure 4 - Flow Diagram for 3929 Block Transfer Q-Repeat Read Commands

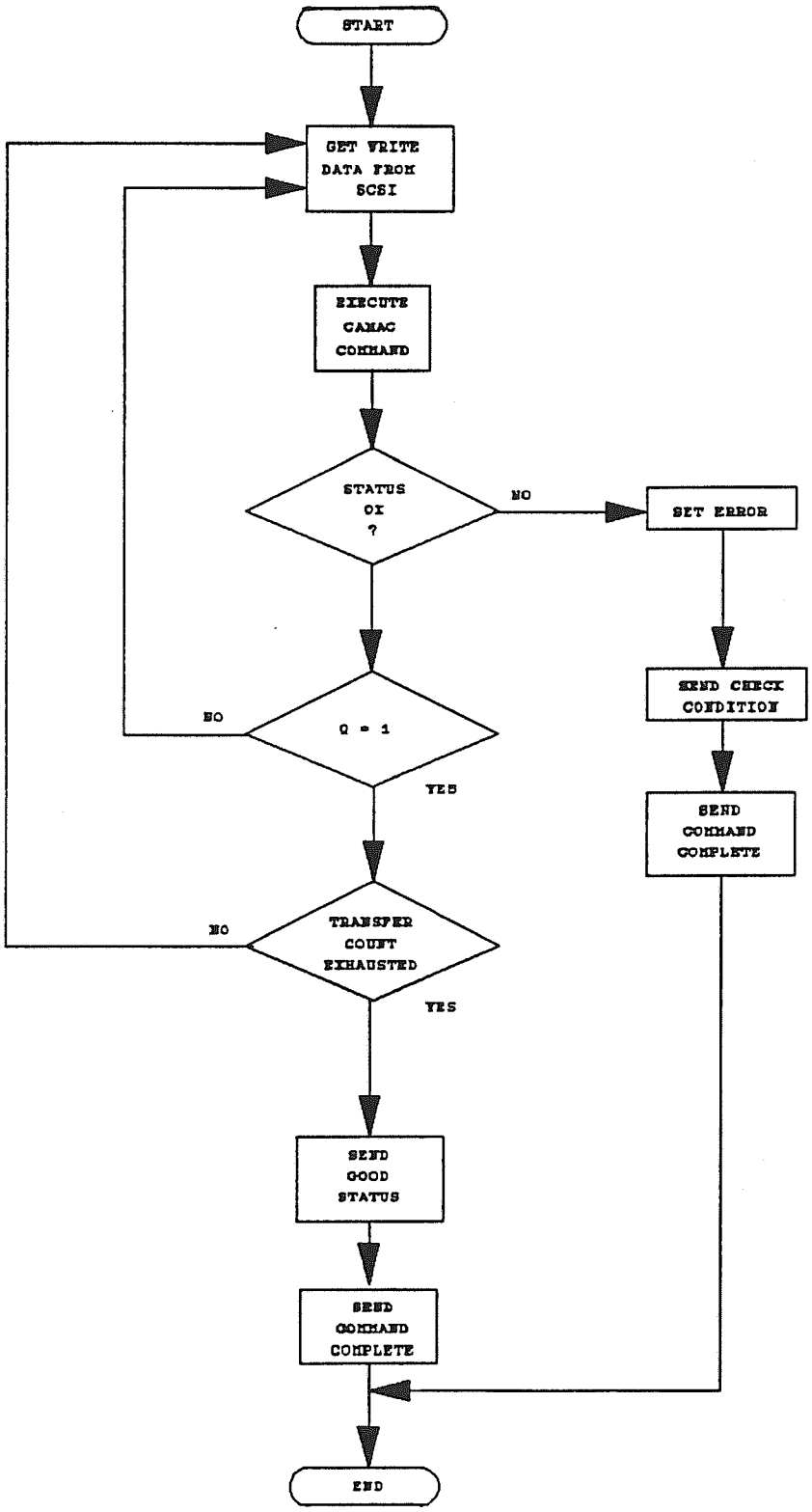


Figure 5 - Flow Diagram for 3929 Block Transfer Q-Repeat Write Commands

Q-Scan Block Transfer Operations

To select the Q-Scan Transfer Mode, set both the Transfer Mode bits, TM2 and TM1, to ones when loading the Mode Control byte in the CDB. During Q-Scan Block Transfer operations, the 3929 uses the Q response from the previously executed CAMAC command to determine the Station Number (N) and Subaddress (A) for the next operation. A Q response of zero indicates that the last valid subaddress of the current Station Number has been accessed. The 3929 responds to a Q response of zero by resetting the Subaddress, incrementing the Station Number, and continuing the scan. A Q response of one indicates that the last CAMAC command was executed to a valid CAMAC address. The 3929 responds to a Q response of one by either storing the read data or retrieving new write data. After a Q response of one is received, the 3929 updates the CAMAC address as follows: the Subaddress is incremented or, if the Subaddress was 15, it is reset to zero, and the Station Number is incremented. Figure 6 illustrates the updating of the CAMAC address based on the CAMAC Q response:

If, due to a programming error, the 3929 increments beyond Station Number 23, the Block Transfer operation is terminated and a CHECK CONDITION is returned.

Figures 7 and 8 are simplified flow diagrams for Q-Scan read and write operations:

Fast Block Transfer Operations

The 3929 can execute Block Transfer operations in either a Fast or Conservative mode. The Conservative Block Transfer operations are enabled by setting the FAST bit in the Mode Control byte to a zero when loading the CDB. The Conservative Block Transfer operations arbitrate for the CAMAC Dataway for every CAMAC cycle. This arbitration conforms to the Request/Grant protocol used for multiple controllers in a CAMAC crate. Refer to the Auxiliary Controller Configuration section of this manual for further details regarding the Request/Grant protocol.

The Fast Block Transfer operations are enabled by setting the FAST bit in the Mode Control byte to a one when loading the CDB. During Fast Block Transfer operations, the CAMAC Dataway is only arbitrated for the initial cycle. After the first cycle is executed, the 3929 maintains the assertion of the CAMAC BUSY signal and the Auxiliary Controller Bus (ACB) REQUEST INHIBIT signal. These signals are asserted until the 3929 has completed the entire Block Transfer operation. The CAMAC Dataway is only relinquished if the Auxiliary Controller Lockout (ACL) signal is asserted. This mechanism provides a speed increase in CAMAC data acquisition since the 3929 does not need to arbitrate for every CAMAC cycle. This mode should be used with care since all other crate controllers that do not use the ACL method of arbitration are prevented from accessing the Dataway.

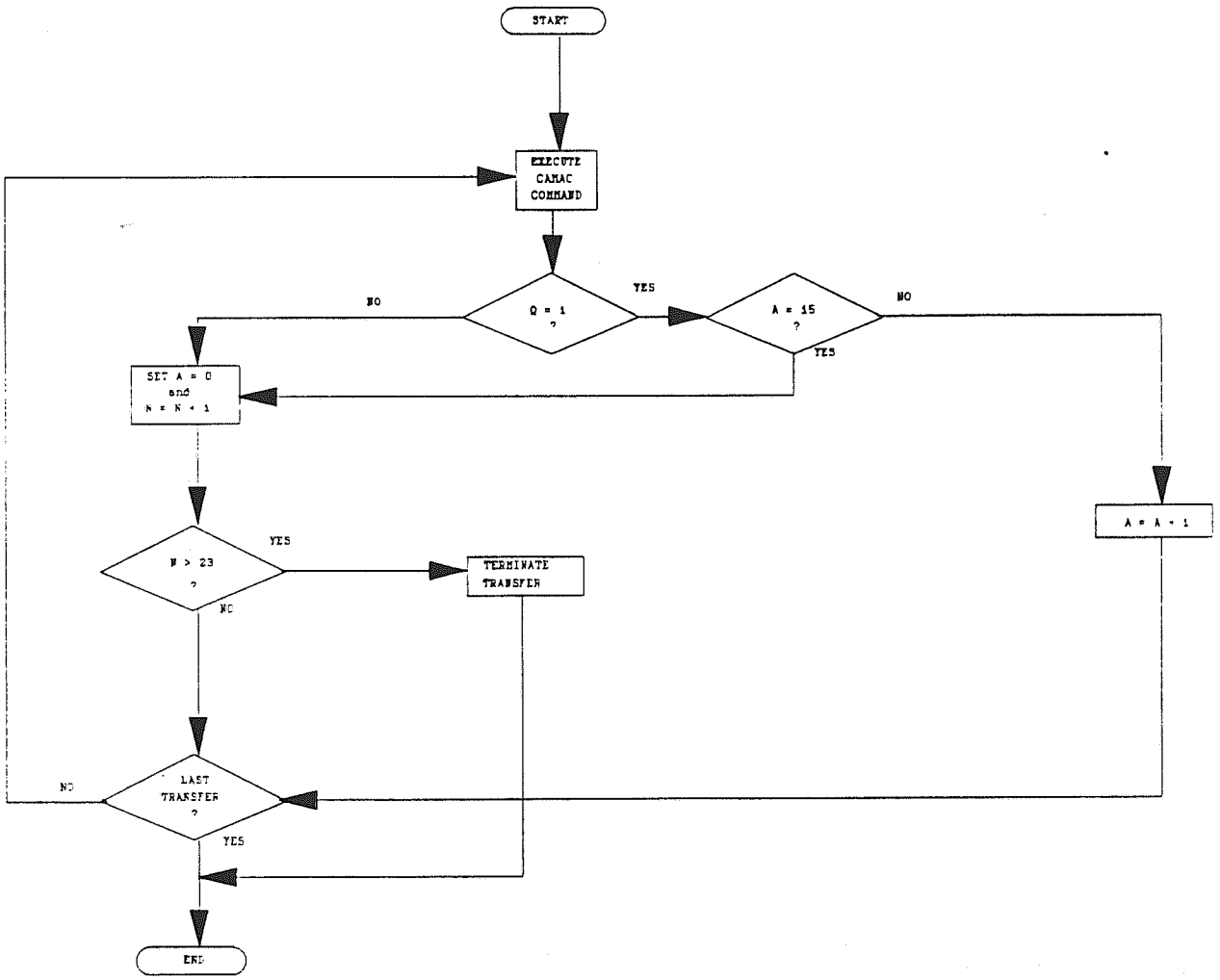


Figure 6 - Flow Diagram for 3929 Q-Scan Address Updating

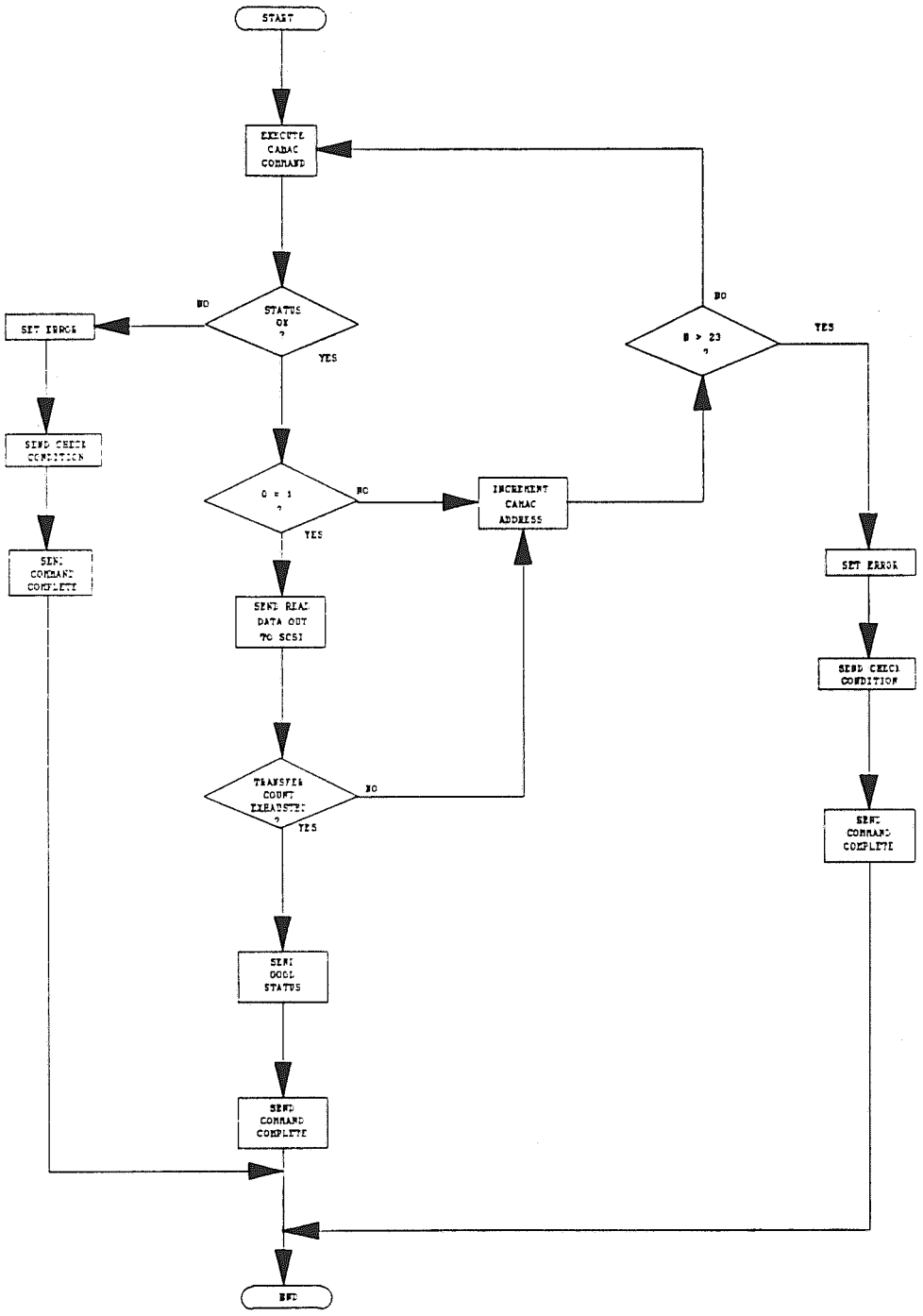


Figure 7 - Flow Diagram for 3929 Block Transfer Q-Scan Read Commands

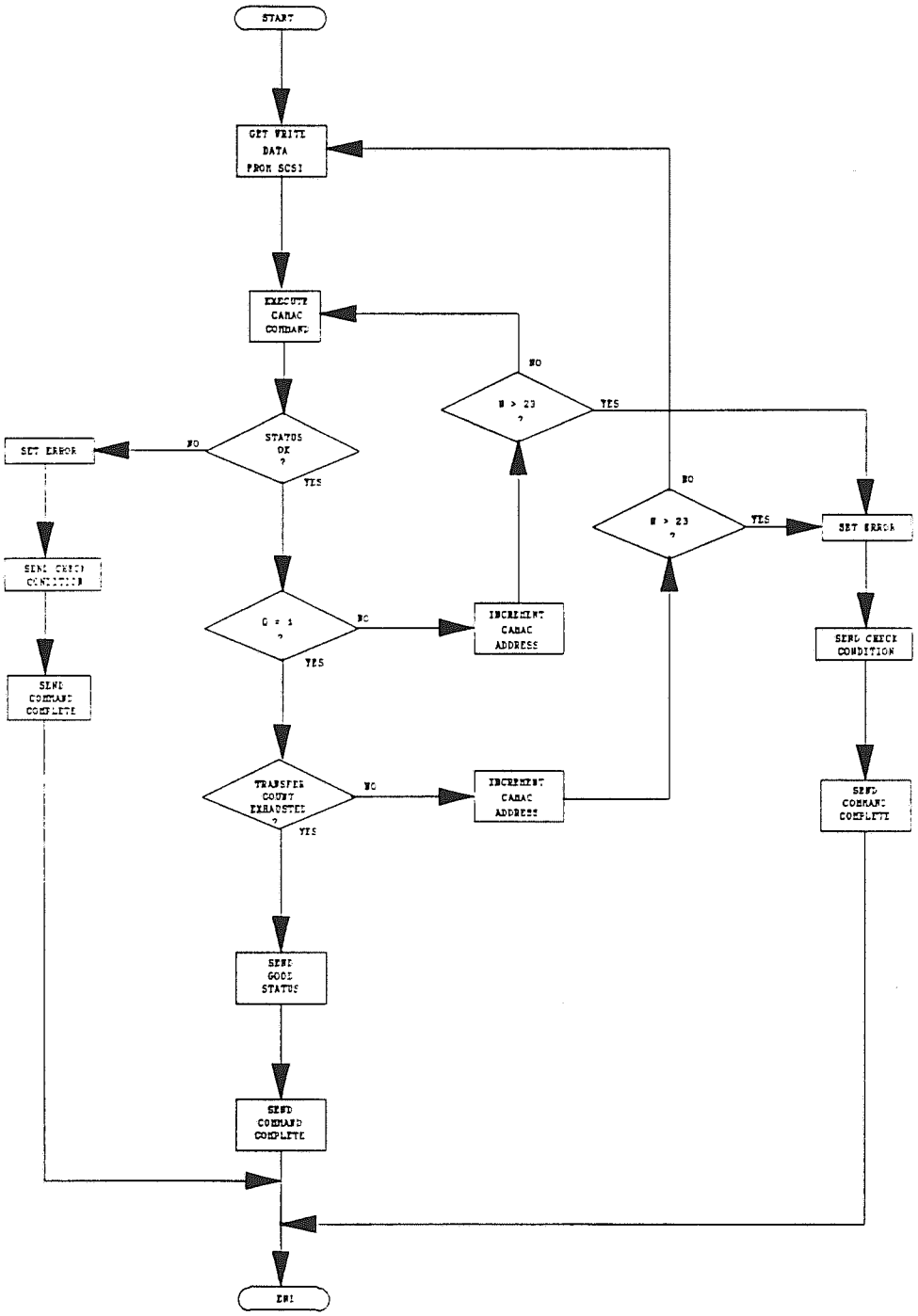


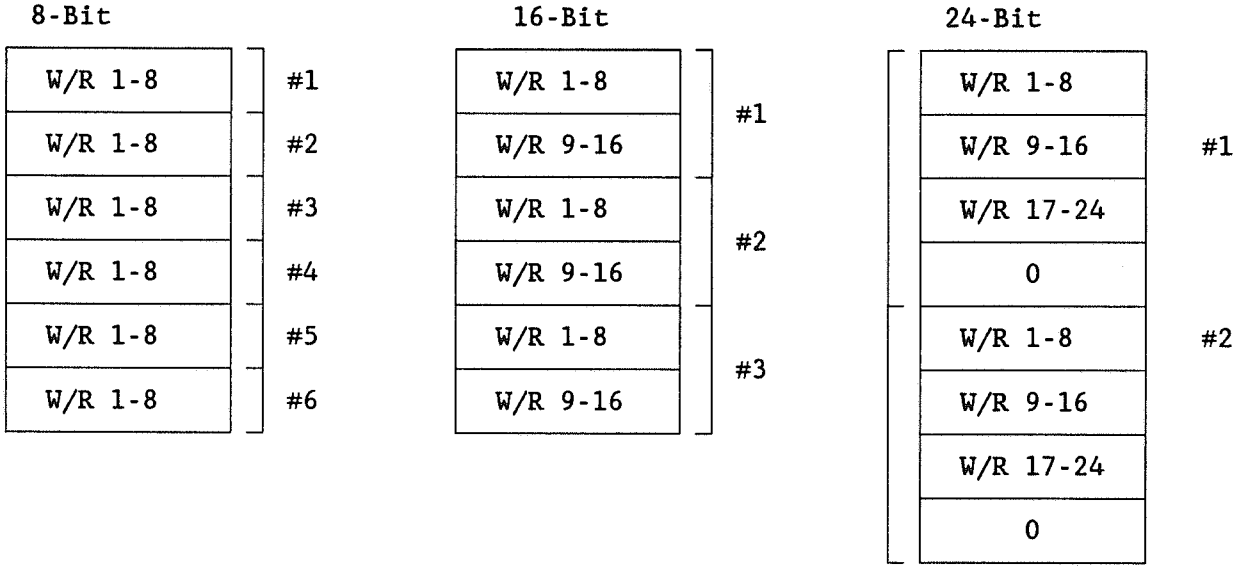
Figure 8 - Flow Diagram for 3929 Block Transfer Q-Scan Write Commands

Data Transfer Formats

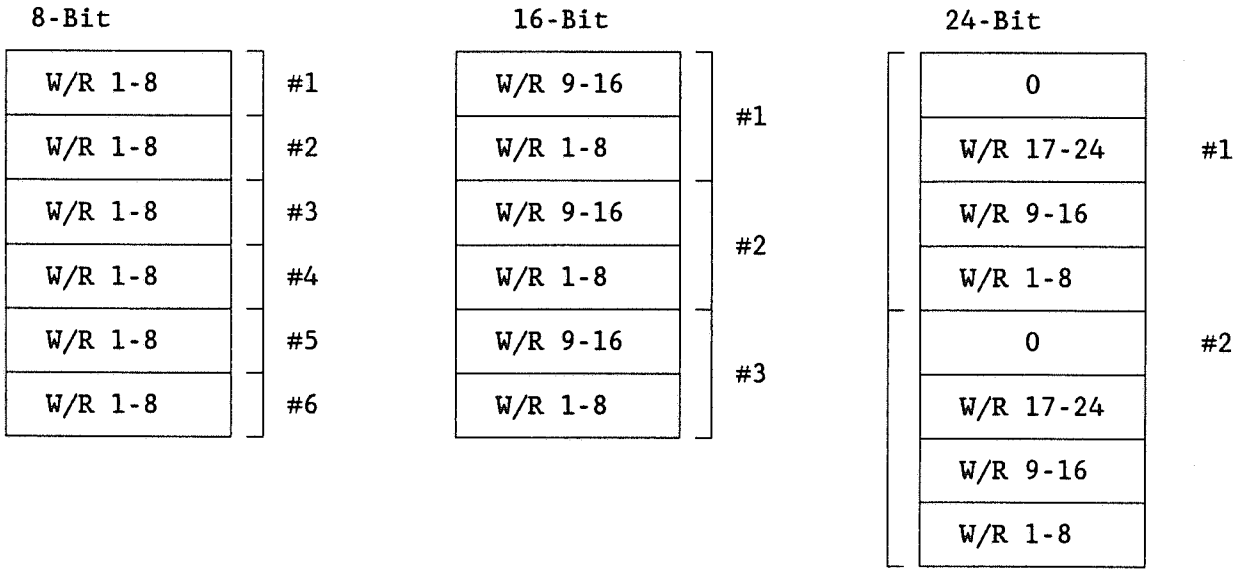
The Single and Block Transfer operations in the 3929 may use CAMAC data word sizes of either 8, 16, or 24-bits. This selection is made when the Mode Control byte of a CDB is loaded.

The data for CAMAC operations may be sent/received to/from the 3929 as low-byte first or high-byte first. The 3929 is configured at the factory as low-byte first. The following diagrams show the order in which data is sent/received to/from the 3929. The first byte transferred is shown at the top of the diagram and the last byte transferred is at the bottom. Refer to the Strap Option Section of this manual for further information.

Low-byte First Transfers



High-byte First Transfers



Asynchronous Event Notification

The SCSI Asynchronous Event Notification (AEN) protocol defines a mechanism whereby a target may assume the role of a temporary initiator and request the transfer of data. The 3929 uses the AEN mechanism to report the occurrence of predetermined LAM's to the SCSI initiator.

Before the 3929 can execute an AEN, the desired LAM's must be booked. Booking a LAM refers to enabling the 3929 to sense a specified LAM and take action based upon that LAM. The SCSI BOOK LAM command is used to setup and enable individual LAMs. The BOOK LAM command consists of a LAM Identification, a LAM Type, a CLEAR LAM NAF, a DISABLE LAM NAF, and a few user defined fields. Refer to the BOOK LAM Command section of this manual for further information regarding booking LAMs.

The LAM identification parameter defines the LAM number. This number corresponds to the station number in which the LAM is to be setup and enabled. (LAM Identification 17 corresponds to the LAM generated from station number 17.) The LAM Type parameter specifies the action to be taken once the LAM has been generated. A Type 0 LAM indicates that both the CLEAR and DISABLE NAF's are to be executed. A Type 1 LAM indicates that only the CLEAR NAF is to be executed. The three user defined fields are general purpose and are returned to the "normal" initiator when the AEN occurs.

There are two CAMAC commands specified with the BOOK LAM Command. One command is for the CLEAR operation and the other is for the DISABLE operation. When a selected LAM is generated, the 3929 examines the LAM type and executes the predetermined CAMAC commands. The two commands specified must be CAMAC control operations as there is no mechanism for associating write or read data with a particular LAM.

After the predetermined CAMAC commands are executed, the 3929 becomes a temporary initiator and selects the "normal" initiator as a target. Once the connection is made, a SCSI SEND command is executed. This command instructs the "normal" initiator to receive four bytes of data as specified by the Transfer Count fields in the SEND commands Command Descriptor Block. The "normal" initiator then accepts the four bytes of data from the 3929. This data contains the LAM identification and the three user defined fields. The following shows the order in which the data bytes are transferred.

Byte 1	LAM Identification
Byte 2	User Defined Field 1
Byte 3	User Defined Field 2
Byte 4	User Defined Field 3

These four bytes that are returned are the same data bytes used for the selected LAM when the BOOK LAM was executed. After the "normal" initiator receives the four bytes, it must send the Status Byte and a COMMAND COMPLETE MESSAGE that ends the AEN.

List Processing

The 3929 provides for a means of executing a preloaded list of CAMAC commands, termed List Processing. This list of commands may contain block transfer operations, single CAMAC operations, or CAMAC write operations with the write data embedded in the list. List processing provides a mechanism for executing multiple CAMAC operations with minimum software overhead. Once the list has been loaded in the 3929, the only operation needed to execute the list is the EXECUTE LIST SCSI command. Once the list is started, all elements in the list are executed until a HALT instruction is encountered.

Two other SCSI commands are used for list processing. These commands are the LOAD LIST and RESUME LIST commands. Refer to the individual SCSI command descriptions for the format of each command. The LOAD LIST command is used for sending the list elements to the 3929. The RESUME LIST command is used for resuming list execution once the list has been prematurely terminated due to an error condition.

List Processing Format

The following describes the format of the four types of list instructions. The command instruction types include Single CAMAC operations, Conservative Block Transfer operations, Fast Block Transfer operations, and Single Write With In-Line data.

All instructions have a minimum of four bytes. The first byte contains the instruction type, the second is reserved, the third contains the NAF low and the fourth contains the NAF high. The following describes the first four bytes of any list processing instruction.

Instruction Byte 1

The first byte of the instruction contains the opcode for the instruction. This byte also specifies the CAMAC Transfer Mode, CAMAC Data Word Size, and the Abort Disable selection for CAMAC type instructions. The following table shows the layout of the fourth instruction byte, the opcode byte.

TABLE 31 - First Instruction Byte

7	6	5	4	3	2	1	0
CM	OP2	OP1	TM2	TM1	WS2	WS1	AD

Bit 7 is used to specify either CAMAC or non-CAMAC instructions. If this bit is set to a zero, the instruction is a CAMAC type instruction. If this bit is set to one, the instruction is a non-CAMAC operation. The only non-CAMAC operation currently defined is the HALT instruction.

Bits 6 and 5 are used to define the type of CAMAC operation (CM bit is set to zero). If the CM bit is set to one, these bits should be set to zero, which defines the HALT instruction. The following chart shows the type of CAMAC operation defined for the binary combination of these two bits.

OP2	OP1	Instruction
0	0	Single CAMAC Operation
0	1	Conservative Block Transfer Operation
1	0	Fast Block Transfer Operation
1	1	Single Write With In-Line Data

Bits 4 and 3 are used to specify the Transfer Mode for the CAMAC operation (CM bit is zero). If the CM bit is set to one, these bits should be set to zero to indicate a HALT instruction. The following chart shows the CAMAC Transfer Modes obtained with the binary combination of these bits. Please refer to the Single CAMAC Operations and Block Transfer Operations section of this manual for further information regarding the Transfer Modes.

TM2	TM1	Transfer Mode
0	0	Q-Stop Transfers
0	1	Q-Ignore Transfers
1	0	Q-Repeat Transfers
1	1	Q-Scan Transfers

Bits 2 and 1 are used to specify the CAMAC Data Word Size to be used during a CAMAC operation (CM bit is set to zero). If the CM bit is set to one, these bits should be set to zero for the HALT instruction. The following chart shows the CAMAC Data Word Sizes obtained by the binary combination of these bits. Please refer to the Single CAMAC Operations and Block transfer Operation of this manual for additional information.

WS2	WS1	CAMAC Data Word Size
0	0	24-Bit Data Size
0	1	16-Bit Data Size
1	0	8-Bit Data Size
1	1	Reserved

Bit 0 is used to enable or disable the termination of a CAMAC operation based on the CAMAC X-response. Refer to the Single CAMAC Operations and Block Transfer Operations section of this manual for further information on the use of the ABORT DISABLE bit. For non-CAMAC operations, this bit should be set to zero to indicate a HALT instruction.

Instruction Byte 2

Instruction byte 2 is reserved and must be set to zero.

Instruction Byte 3

The third byte of the instruction contains a portion of the CAMAC NAF to be used during a CAMAC operation. The fourth byte of the instruction contains the remaining bits for the NAF specification. The following table shows the bits defined in the second instruction byte.

TABLE 32 - Third Instruction Byte

7	6	5	4	3	2	1	0
A4	A2	A1	F16	F8	F4	F2	F1

Bits 7 through 5 are used in conjunction with bit 0 of the fourth instruction byte for specifying the CAMAC Subaddress to be used during the CAMAC operation.

Bits 4 through 0 are used for specifying the CAMAC Function Code to be executed during the CAMAC operation.

Instruction Byte 4

The fourth byte of the instruction contains a portion of the CAMAC Station Number, Subaddress, and Function Code (NAF) to be used for the command. The third byte of the instruction contains the remaining bits of the NAF. The following table shows the bits defined for the first instruction byte:

TABLE 33 - Fourth Instruction Byte

7	6	5	4	3	2	1	0
0	0	N16	N8	N4	N2	N1	A8

Bits 7 and 6 are not used and must be set to zero.

Bits 5 through 1 are used to specify the CAMAC Station Number to be accessed during the CAMAC operation.

Bit 0 is the most significant bit of the CAMAC Subaddress field. This bit is used in conjunction with bits 7 through 5 of the third instruction byte to form the Subaddress to be used during the CAMAC operation.

Instruction Bytes 5 through 8

These instruction bytes are only required for CAMAC Block Transfer and the Single Write with In-Line Data operations. The Single CAMAC operations and HALT instructions do not use these bytes.

For the Block Transfer operations, these four bytes contain the transfer count for the operation. This transfer count specification is the two's complement form of the actual transfer count. Note that this is different from the transfer count specifications for the SCSI commands. The last and most-significant byte of the four must be set to FF hex. The remaining three bytes are actually used for the transfer count specification.

For the Single Write With In-Line Data instruction, these four bytes are used to contain the CAMAC write data for the operation. The last of the four bytes is not used and should be set to zero. The other three bytes contain the actual CAMAC write data for the command. Byte 5 contains write data bits 1-8, byte 6 contains write data bits 9-16, and byte 7 contains write data bits 17-24.

Single CAMAC Operation Instruction

This instruction is used to execute a single CAMAC operation within the CAMAC crate. This instruction can be used to execute CAMAC read, write, or control operations. For control operations, no data is associated with the instruction. For CAMAC read and write operations, data is transferred to or from the on-board FIFO as required. As with non-list processing, the number of data bytes transferred depends on the CAMAC Data Word Size selection specified with the instruction. The following table shows the instruction format for the Single CAMAC Operation instruction.

TABLE 34 - Single CAMAC Instruction Format

7	6	5	4	3	2	1	0	
0	0	0	TM2	TM1	WS2	WS1	AD	Byte 1
0	0	0	0	0	0	0	0	Byte 2
A4	A2	A1	F16	F8	F4	F2	F1	Byte 3
0	0	N16	N8	N4	N2	N1	A8	Byte 4

Single Write With In-Line Data Instruction

This instruction is used to execute a single CAMAC write operation. The CAMAC write data to be used for the operation is contained in the list. This command is useful for embedding CAMAC write operations in a read list. This instruction may only be used for CAMAC write operations. If a CAMAC read or control operation is specified with this instruction, the processing terminates and a CHECK CONDITION is returned. The Sense Key is set to 5, the Sense Code is set to 80 hex, and the Sense Code Qualifier is set to 1. The following table shows the format of the Single Write with In-Line Data instruction:

TABLE 35 - Single Write w/In-Line Data

7	6	5	4	3	2	1	0	
0	1	1	0	TM1	WS2	WS1	AD	Byte 1
0	0	0	0	0	0	0	0	Byte 2
A4	A2	A1	F16	F8	F4	F2	F1	Byte 3
0	0	N16	N8	N4	N2	N1	A8	Byte 4
W8	W7	W6	W5	W4	W3	W2	W1	Byte 5
W16	W15	W14	W13	W12	W11	W10	W9	Byte 6
W24	W23	W22	W21	W20	W19	W18	W17	Byte 7
0	0	0	0	0	0	0	0	Byte 8

Conservative Block Transfer Instruction

This instruction is used for executing a Conservative CAMAC Block Transfer operation. This command is used for executing either CAMAC read or write operations. If a CAMAC control operation is specified, processing terminates and a CHECK CONDITION is returned. The Sense Key is set to 5, the Sense Code is set to 80 hex, and the Sense Code Qualifier is set to one.

Data transfer to/from the CAMAC Dataway during the Block Transfer operation occurs in the same fashion as the non-list processing block transfers. Refer to the Executing Block Transfers section of this manual for additional information.

This instruction requires a transfer byte count specification. This specification differs from the non-list processing transfer count specification. The transfer count for list processing must be in the two's complement form. The following table shows the format for the Conservative Block Transfer Operation instruction.

TABLE 36 - Conservative Block Transfer Operation

7	6	5	4	3	2	1	0	
0	0	1	TM2	TM1	WS2	WS1	AD	Byte 1
0	0	0	0	0	0	0	0	Byte 2
A4	A2	A1	F16	F8	F4	F2	F1	Byte 3
0	0	N16	N8	N4	N2	N1	A8	Byte 4
TC8	TC7	TC6	TC5	TC4	TC3	TC2	TC1	Byte 5
TC16	TC15	TC14	TC13	TC12	TC11	TC10	TC9	Byte 6
TC24	TC23	TC22	TC21	TC20	TC19	TC18	TC17	Byte 7
1	1	1	1	1	1	1	1	Byte 8

Fast Block Transfer Instruction

This instruction is used for executing a Fast CAMAC Block Transfer operation. This command is used for executing either CAMAC read or write operations. If a CAMAC control operation is specified, processing terminates and a CHECK CONDITION is returned. The Sense Key is set to 5, the Sense Code is set to 80 hex, and the Sense Code Qualifier is set to 1. Data transfer to/from the CAMAC Dataway during the Block Transfer operation occurs in the same fashion as the non-list processing block transfers. Refer to the Executing Block Transfers section of this manual for additional information.

This instruction requires a transfer byte count specification. This specification differs from the non-list processing transfer count specification. The transfer count for list processing must be in the two's complement form. The following table shows the format for the Fast Block Transfer Operation instruction:

TABLE 37 - Fast Block Operation

7	6	5	4	3	2	1	0	
0	1	0	TM2	TM1	WS2	WS1	AD	Byte 1
0	0	0	0	0	0	0	0	Byte 2
A4	A2	A1	F16	F8	F4	F2	F1	Byte 3
0	0	N16	N8	N4	N2	N1	A8	Byte 4
TC8	TC7	TC6	TC5	TC4	TC3	TC2	TC1	Byte 5
TC16	TC15	TC14	TC13	TC12	TC11	TC10	TC9	Byte 6
TC24	TC23	TC22	TC21	TC20	TC19	TC18	TC17	Byte 7
1	1	1	1	1	1	1	1	Byte 8

HALT Instruction

The HALT instruction is used to signify the end of a list-processing sequence. This is the only non-CAMAC instruction defined. The following table shows the format of the HALT instruction:

TABLE 38 - HALT Instruction Format

7	6	5	4	3	2	1	0	
1	0	0	0	0	0	0	0	Byte 1
0	0	0	0	0	0	0	0	Byte 2
0	0	0	0	0	0	0	0	Byte 3
0	0	0	0	0	0	0	0	Byte 4

List Processing Operation

The list processing operation begins with the list being sent over to the 3929. The list is transferred to the 3929 by using the LOAD LIST instruction. Refer to the LOAD LIST SCSI Command description for the format of the command.

The LOAD LIST command contains two bytes for the specification of the list load address. The valid address range for list processing memory is from 0 to BFFF hex. This allows for approximately 48 Kilo-bytes of list storage. Care must be taken when loading the list, that the transfer count specification does not cause the 3929 to deposit list data over the BFFF hex boundary.

The command list sent to the 3929 MUST have data transfers in the same direction. For example, the list MUST contain only CAMAC read operations or only CAMAC write operations. This limitation is only to those operations that require data transfer over the SCSI bus. Since the Single Write With In-Line Data instruction has its write data within the list, this instruction may reside in a CAMAC read list. Also, CAMAC control operations may be embedded anywhere in the list as long as they appear in the correct format.

Once the list is loaded, it is executed by using the EXECUTE LIST SCSI command. In this command, an address specification is used to inform the 3929 as to the initial address of list execution. This allows for multiple sub-lists to reside within the 3929. Again, this address specification must be in the address range of 0 to BFFF hex. The EXECUTE LIST SCSI command format also requires that a transfer count be specified. This count is the total number of bytes that the 3929 will transfer over the SCSI bus during the execution of the list. This count includes only data transferred during the SCSI Data phase. Therefore, all individual block transfer counts and single transfer counts within the list must be totalled to obtain this number.

When list execution starts, the 3929 reads the first four elements starting at the address specified by the EXECUTE LIST instruction. The third and fourth bytes are then loaded into the internal CAMAC Command Register of the 3929. The first byte is then examined to determine the type of instruction. If the instruction specified a HALT instruction, the 3929 terminates list processing and then sends a Status byte and COMMAND COMPLETE message. If the instruction specified is undefined, the 3929 sends a CHECK CONDITION status with the Sense Key set to 5, the Sense Code set to 80 hex, and the Sense Code Qualifier set to 4. If the instruction was a CAMAC operation, the internal word count and Mode Control registers are loaded and the CAMAC operation begins. After the CAMAC operation is complete, the status is examined. List processing continues if no errors were detected during the operation. If an error occurs, the 3929 switches to the Status phase and returns a CHECK CONDITION. The Sense Key is set to 0B hex, the Sense Code is set to 80 hex, and the Sense Code Qualifier is set to either 1 to 2. The qualifier is set to 1 for Single operations that terminated and set to 2 for Block Transfer operations.

List Processing Example

As an example, assume it is desired to read a two channel analog-to-digital converter (ADC) located in slot number 2. 1024 samples of the analog data are to be taken from each channel. The ADC module is accessed by the following commands:

F(2) A(0) --	Reads the digitized data. Q response of 1 indicates data is valid.
F(17) A(0) --	Selects the channel to be read. Data = 1 selects channel 1. Data = 2 selects channel 2.
F(24) A(0) --	Disables ADC conversions.
F(26) A(0) --	Enables ADC conversions.

For this application, an F(17)A(0) with data = 1 is first executed to select the desired channel. An F(26)A(0) is then executed to enable conversions. A Standard CAMAC Block Transfer

operation is then executed in the Q-Repeat mode to read the 1024 converted samples. After the samples are read, an F(24)A(0) command is executed to disable the conversions. These steps are then repeated for channel 2.

For this example, the instruction list would appear as follows: (All data shown is in hex).

```
60 ┐
00 │
11 │  Single Write With In-Line Data / 24-Bit / Q-Stop / Abort Enable
04 │  N(2) F(17) A(0)
01 │  Data High = 0
00 │  Data Middle = 0
00 │  Data Low = 1
00 ┘

00 ┐  Single CAMAC Transfer / 24-Bit / Q-Stop / Abort Enable
00 │  N(2) F(26) A(0)
1A │
04 ┘

30 ┐
00 │  Conservative Block Transfer / 24-Bit / Q-Repeat / Abort Enable
02 │  N(2) F(2) A(0)
04 │  Transfer Byte Count = -(1024 * 4) = -4096
00 │
F0 │
FF │
FF ┘

00 ┐  Single CAMAC Transfer / 24-Bit / Q-stop / Abort Enable
00 │  N(2) F(24) A(0)
18 │
04 ┘

60 ┐
00 │  Single Write With In-Line Data / 24-Bit / Q-Stop / Abort Enable
11 │  N(2) F(17) A(0)
04 │  Data High = 0
02 │  Data Middle = 0
00 │  Data Low = 2
00 │
00 ┘

00 ┐  Single CAMAC Transfer / 24_bit / Q-Stop / Abort Enable
00 │  N(2) F(26) A(0)
1A │
04 ┘

30 ┐
00 │  Conservative Block Transfer / 24-Bit / Q-Repeat / Abort Enable
02 │  N(2) F(2) A(0)
04 │  Transfer Byte Count = -(1024 * 4) = -4096
00 │
F0 │
FF │
FF ┘

00 ┐  Single CAMAC Transfer / 24-Bit / Q-Stop / Abort Enable
00 │  N(2) F(24) A(0)
18 │
04 ┘

80 ┐  HALT Instruction
00 │
00 │
00 ┘
```

Model 3929-Z1B

Auxiliary Controller Configuration

The 3929 may be configured as either a main or auxiliary controller. The 3929 is shipped from the factory configured as a main controller. If the 3929 is to be used as an auxiliary controller, the socketed resistor packs must be removed from the A- and B-boards. The A-board is the left-most board as viewed from the front panel and the B-board is the right-most board. The A-board has 6 resistor packs that need to be removed and the B-board has 9. Figures 9 and 10 show the locations of the resistor packs on each board.

To configure the 3929 as an auxiliary controller, follow the procedure below.

1. Remove the rear mounting brackets (2) by removing the 4 screws.

2. Remove the 2 screws that hold the A-board to the front panel.

3. Separate the A and B-boards by gently lifting up on the rear of the A-board until the inter-board connector disengages.

4. On the A-board, remove the 7 resistor packs that are shown in Figure 9.

5. On the B-board, remove the 10 resistor packs that are shown in Figure 10.

6. Store the resistor packs for future use.

7. Reassemble the 3929, making sure the interconnection pins line up with the mating socket.

Note: When reinstalling the resistor packs for main controller operation, observe the keying indicator on each resistor pack.

To support multiple controllers in a CAMAC crate, a 40-position connector is required on the rear panel of each controller for connection to the Auxiliary Controller Bus (ACB). The interconnection is made via a 40-conductor flat ribbon cable. The ACB includes 3 control signals (ACL, REQUEST and REQUEST INHIBIT), 5 encoded-N signals, and 24 LAM signals. These signals are described in the following text.

All ACB signals are driven with either tri-state or open-collector drivers. Refer to APPENDIX A for the signal allocation on the ACB connector. The three control signals, ACL, REQUEST and REQUEST INHIBIT are used in the arbitration for control of the CAMAC Dataway. Two modes, Request/Grant and Auxiliary Controller Lockout (ACL) exist for the arbitration for control of the CAMAC Dataway. The Request/Grant protocol is used by the 3929 for gaining access to the CAMAC Dataway. The 3929 controller can also be used in a crate containing a controller that uses the Auxiliary Controller Lockout arbitration mode to gain control of the dataway.

The remaining signals on the ACB are the 5 ENCODED-N signals (EN16, EN8, EN4, EN2, EN1) and the 24 LOOK-AT-ME signals (AL24-AL1). An auxiliary controller, when executing an addressed CAMAC operations, transmits the binary coded station number to the main crate

controller via the ACB ENCODED-N lines. The main crate controller decodes these signals and asserts the appropriate CAMAC N signal to the Dataway at the control station. The CAMAC signals L24 through L1 are retransmitted by the main crate controller to all other controllers in the crate via the ACB signals AL24 through AL1.

Request/Grant Mode

The Request/Grant arbitration mode for control of the Dataway involves the three signals, REQUEST, REQUEST INHIBIT and ACB GRANT. The signal REQUEST is common to all controllers via the ACB, creating a wired-OR signal. Also, the signal REQUEST is extended to a single-pin LEMO on the front panel of each controller that supports the Request/Grant protocol. The signal REQUEST INHIBIT is common to all controllers via the ACB. The signal ACB GRANT is connected in a daisy-chained fashion between all controllers supporting the Request/Grant protocol. The front-panel single-pin LEMO connectors, ACB GRANT IN and ACB GRANT OUT, are used for connection of the signal ACB GRANT, as described below. If the 3929 is the only controller in the crate, the connection between the front-panel connectors REQUEST to ACB GRANT IN must be made.

FROM	TO
REQUEST of the highest priority controller	ACB GRANT IN of the highest priority controller
ACB GRANT OUT of the highest priority controller	ACB GRANT IN of the second highest priority controller
ACB GRANT OUT of the second highest priority controller	ACB GRANT IN of the third highest priority controller

This pattern continues until the lowest priority controller has been reached. There is no connection to the ACB GRANT OUT LEMO connector on the lowest priority controller.

Both signals, ACL and REQUEST INHIBIT, must be false before a controller may request for control of the CAMAC Dataway. If both signals are false, the controller may then assert REQUEST. The assertion of REQUEST is received by the highest priority controller at its front-panel LEMO connector ACB GRANT IN (since REQUEST is connected to ACB GRANT IN on its front panel). If the highest priority controller did not request control of the Dataway, it asserts the signal ACB GRANT OUT on its front-panel single-pin LEMO. The signal ACB GRANT propagates to the other controllers in the order of priority established during the configuration. When the requesting controller receives the assertion of the signal ACB GRANT, the signal propagation ceases. The requesting controller does not assert the ACB GRANT OUT signal, but instead asserts the ACB signal REQUEST INHIBIT. Once REQUEST INHIBIT is asserted, the controller that gained control of the Dataway executes its cycle(s).

When two or more controllers assert the REQUEST signal at the same time, control of the Dataway is given to the highest priority requesting controller. The propagation of the signal ACB GRANT is as described above. Once the controller has been granted control of the

CAMAC Dataway and asserts REQUEST INHIBIT, the other controllers remove their assertion of REQUEST. One or more CAMAC cycles then occurs. To relinquish control of the Dataway, the controller removes the assertion of the REQUEST INHIBIT signal. The Dataway is then free for further arbitration.

Auxiliary Controller Lockout Mode

The ACL mode for control of the CAMAC Dataway involves the signal Auxiliary Controller Lockout (ACL). The signal ACL is common to all controllers via the ACB. Only one controller per crate is allowed to use the signal ACL to gain control of the crate. All controllers in a crate must respond to the ACL signal as follows. When ACL is asserted by a main controller, any CAMAC operation in progress is aborted if the CAMAC Dataway signal S1 has not been asserted for the current operation. If the assertion of the S1 signal has already occurred, the CAMAC operation is allowed to complete. After the cycle is complete, control is given to the controller asserting ACL. The 3929 may be used in a crate containing a controller that uses the ACL mode of arbitration.

Front Panel

The front panel of the 3929 contains one thumb-wheel switch, two toggle switches, 13 LEDs, three single-pin LEMOs, a 50 position low-density connector and a 50 position high-density SCSI connector. The thumb-wheel switch is used for selecting the SCSI address of the 3929. The toggle switches are used for On-Line and Off-Line CAMAC operations. The LEDs are used to reflect the status of the 3929. The three LEMO connectors are used for configuring the Request/Grant protocol. The two 50-position connectors are used for connection to the SCSI bus.

Front Panel Switches

The front-panel mounted thumb-wheel switch is used for selecting the SCSI identification of the 3929. This selection ranges from zero to seven. This selection is used to identify the 3929s SCSI address during the Selection phase of a SCSI sequence. Note that this switch is only read at power-up. Changing this switch while the 3929 is powered has no effect on changing the SCSI address.

The On-Line/Off-Line switch enables or disables the 3929 from executing CAMAC operations. When the On-Line/Off-Line switch is in the Off-Line position, the 3929 is prevented from executing CAMAC operations initiated from the users software via the SCSI bus. When the 3929 is Off-Line, any SCSI command which requires a CAMAC Dataway operation, terminates with a CHECK CONDITION. The Sense Key is set to 2, the Sense Code is set to 4, and the Sense Code Qualifier is set to 3. The TEST UNIT READY command can be used to test the state of the On-Line/Off-Line switch. Refer to the TEST UNIT READY SCSI command description for further information. Also, when the switch is in the Off-Line position, the C/Z momentary front panel switch is enabled. CAMAC operations initiated by software are executed if the On-Line/Off-Line switch is in the On-Line position. The Off-Line bit of the CAMAC Control/Status Register, which is read with an N(30)F(1)A(0) command, reflects the position of the switch.

Model 3929-Z1B

The Off-Line C/Z switch is a double-throw, return-to-center, momentary switch. This switch is enabled when the On-Line/Off-Line switch is in the Off-Line position. When the Off-Line C/Z switch is activated to the C position, the 3929 executes a CAMAC Clear (C) operation. A CAMAC Initialize (Z) operation is executed when the switch is activated to the Z position. If the On-Line/Off-Line switch is in the On-Line position, the Off-Line C/Z switch is inhibited.

Front Panel LEDs

The BUSY LED flashes each time the CAMAC BUSY signal is asserted by the 3929 (i.e., during all CAMAC operations executed by the 3929).
The NO-X LED is lit when a CAMAC operation results in a CAMAC X-response of zero. This LED is illuminated on power-up as a result of the initial CAMAC Z operation. Thus, LED is not updated with the X-response obtained from executing any N(30) commands.

The NO-Q LED is lit when a CAMAC operation results in a CAMAC Q-response of zero. This LED is illuminated on power-up as a result of the initial CAMAC Z operation. This LED is not updated with the Q-response obtained from executing any N(30) commands.

The INHIBIT LED reflects the state of the CAMAC Dataway INHIBIT signal. When the CAMAC signal INHIBIT is asserted on the Dataway, this LED is illuminated. The INHIBIT LED is off when the INHIBIT line is false.

The SELECTED-LAM-PRESENT (SLP) LED is illuminated as long as a selected LAM is present. When no selected LAMs are present, the LED is off.

After application of power to the CAMAC crate, the 3929 executes a CAMAC Initialize (Z) operation. After the "Z" operation, the NO-Q, NO-X and INHIBIT LEDs are illuminated.
The STA1 LED array reflects the state of the 3929 firmware. This array is used primarily for diagnostic purposes. This array should display a pattern of 5 when the 3929 is powered up. The least significant bit of the display is on the bottom.

Front Panel Connectors

The single-pin LEMO connector, ACB REQ, contains the CAMAC Auxiliary Controller Bus (ACB) signal REQUEST. To gain control of the CAMAC Dataway, a crate controller asserts the signal ACB REQUEST and waits for the assertion of the ACB GRANT signal. A single-pin LEMO cable must be connected from the front-panel LEMO connector ACB REQ of the highest priority controller to the front-panel ACB GRANT IN of the highest priority controller. Even if the crate contains only one controller, this connection must be made. Refer to the Auxiliary Controller Configuration section of this manual for further information.

The single-pin LEMO connector ACB GRANT IN contains the CAMAC ACB signal GRANT. This signal is connected among all crate controllers within the crate in a daisy-chained fashion. To gain control of the CAMAC Dataway using the Request/Grant protocol, a crate controller must assert ACB REQUEST and then wait for the ACB GRANT. This LEMO is used for patching the GRANT into the 3929. Refer to the Auxiliary Controller Configuration section of this manual for additional information.

The single-pin LEMO connector ACB GRANT OUT contains the ACB GRANT signal. This signal is connected among all crate controllers within a crate in a daisy-chained fashion. If a crate controller receives the assertion of the ACB GRANT signal at its front panel, and did not issue a request for the Dataway, it passes the GRANT via the ACB GRANT OUT LEMO connector. Refer to the Auxiliary Controller Configuration in this manual for additional information.

The remaining two front panel connectors are used for connection to the SCSI bus. One connector is a 50 position low-density connector and the other is a 50-position high-density connector. APPENDIX A shows the signal allocation on these two connectors.

Strap Options

The 3929 contains three strap options. One strap option is used for enabling or disabling the termination of a Q-Repeat Block Transfer operation when a CAMAC Q-response of one is not obtained for a data word within the 200 millisecond timeout period. A second strap option is used for enabling or disabling the P2 Hold option. The third strap option is used for determining the order in which CAMAC data is used by the 3929.

Q-Repeat Timeout Strap

The Q-Repeat Timeout strap is located on the 3929 B-board. Refer to the Auxiliary Controller Configuration section of this manual for information on separating the boards. Figure 2 shows the location of this strap.

If the Q-Repeat Timeout strap is in the QENA position, the Q-Repeat Timeout is enabled. When the timeout is enabled, each data word transferred must receive a Q-response of one within 200 milliseconds. If Q=1 is not received before the timeout period expires, the Block Transfer is terminated. This timeout feature is disabled by placing the strap in the QDIS position. When the timeout is disabled, the 3929 may enter an infinite loop if a Q=1 response is never received for a data word. When this occurs, the 3929 must be issued a RESET command.

P2 Hold Strap

The CAMAC Dataway connector, at each normal station, includes two free bused signals, P1 and P2. The 3929 responds to the assertion of P2 as follows. If a module asserts the P2 Hold signal before the 3929 asserts the CAMAC signal S1, the completion of the Dataway cycle is delayed. The CAMAC signal BUSY and the addressed station number N remain asserted. Once the P2 Hold signal is negated, the 3929 completes its delayed operation. The P2 Hold option allows asynchronous CAMAC operations to slow modules that assert P2. When these slow modules are accessed, they assert P2 before S1 time of the CAMAC operation. Thus, the slow module "holds" up the Dataway cycle until it is ready for data transfer.

A strap is used to enable of disable this feature. The strap is located on the 3929 B-board. Refer to the Auxiliary Controller Configuration section of this manual for information on separating the boards. Also, refer to Figure 2 for the location of the strap on the B-board. The

P2 Hold option is enabled by placing the strap in the P2 position, and disabled by placing the strap in the /P2 position.

Byte Order Strap

The Byte Order Strap is located on the 3929 A-board. This strap is used for setting the 3929 to accept either high-byte first or low-byte first CAMAC data. Please refer to the Data Transfer Format section of this manual for additional information.

To select low-byte first transfers, place the LO-FIRST/HI-FIRST strap into the LO-FIRST position. For high-byte first transfers, place the strap into the HI-FIRST position.

SCSI Termination

SCSI termination resistors are provided on the 3929 A-board. These resistor packs must be installed if the 3929 is the last device on the SCSI bus. If the 3929 is not the last SCSI device on the bus, these two resistor packs must be removed. Refer to Figure 9 for the location of the termination resistors on the 3929 A-board.

A 1 amp fuse is located on the 3929 B-board. The fuse is used for the terminator power source. If the SCSI termination is not functioning properly, ensure that this fuse is intact. Figure 10 shows the location of the terminator power fuse on the 3929 B-board.

The termination resistor packs may be powered by either the TERMPWR signal from the SCSI bus or by the internal supply for the 3929. The strap, labeled CTERM/ITERM, is located on the 3929 A-board. Refer to Figure 9 for the location of the strap. The strap is installed in the CTERM position to terminate the resistor packs with the SCSI TERMPWR signal. If the SCSI cable does not supply the TERMPWR signal, install the strap in the ITERM position.

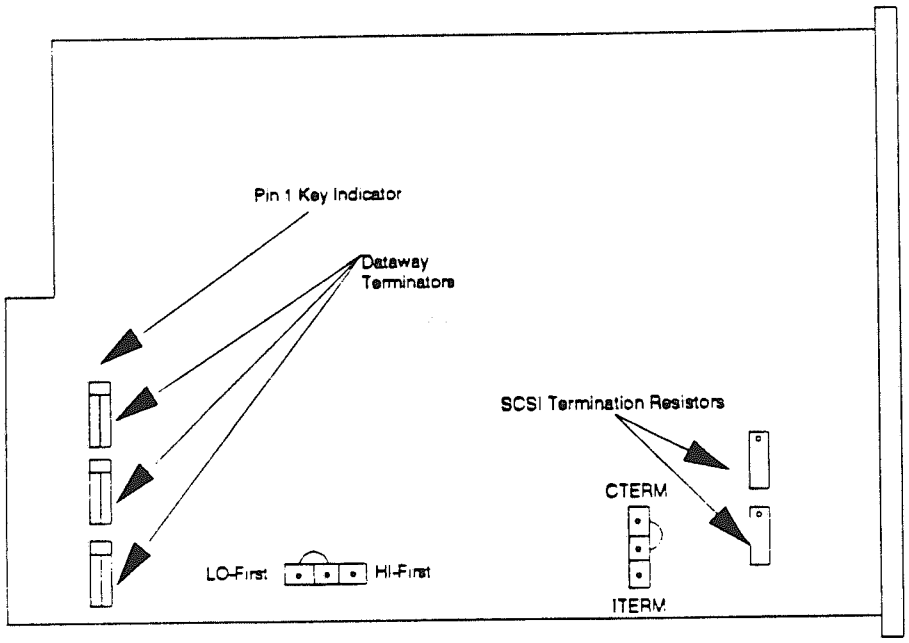


Figure 9 - 3929 A-Board Resistor Pack and Strap Locations

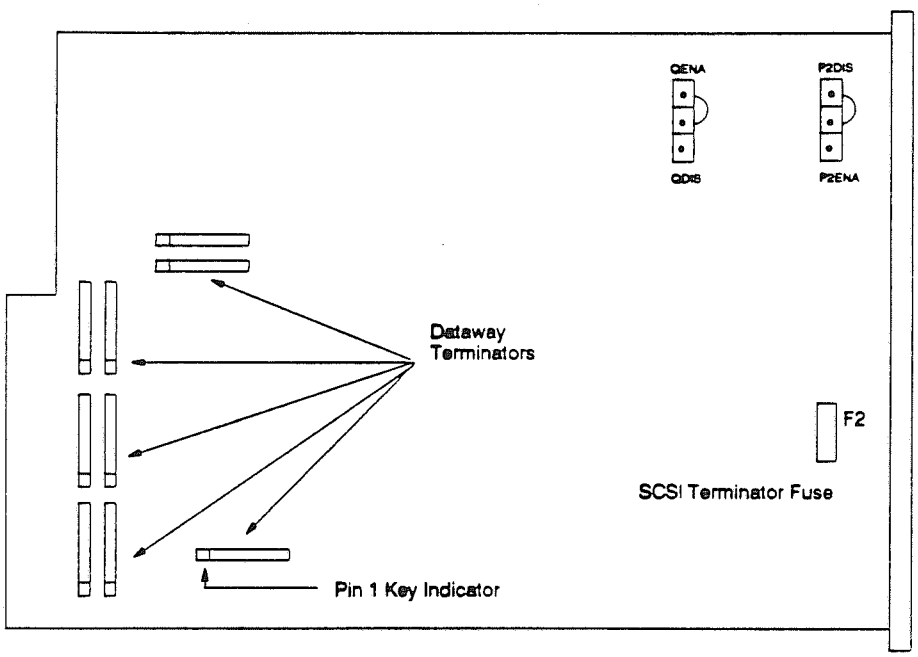


Figure 10 - 3929 B-Board Resistor Pack and Strap Locations

APPENDIX A

Connectors

The following charts show the signal allocation on the various connectors of the 3929.

Auxiliary Controller Bus (ACB) Connector

PIN 1 -- GROUND
PIN 2 -- ENCODED N1
PIN 3 -- ENCODED N2
PIN 4 -- ENCODED N4
PIN 5 -- ENCODED N8
PIN 6 -- ENCODED N16
PIN 7 -- GROUND
PIN 8 -- AUXILIARY CONTROLLER LOCKOUT (ACL)
PIN 9 -- GROUND
PIN 10 -- CONDITIONALLY FREE
PIN 11 -- GROUND
PIN 12 -- REQUEST
PIN 13 -- GROUND
PIN 14 -- REQUEST INHIBIT
PIN 15 -- GROUND
PIN 16 -- AL01
PIN 17 -- AL02
PIN 18 -- AL03
PIN 19 -- AL04
PIN 20 -- AL05
PIN 21 -- AL06
PIN 22 -- AL07
PIN 23 -- AL08
PIN 24 -- AL09
PIN 25 -- AL10
PIN 26 -- AL11
PIN 27 -- AL12
PIN 28 -- AL13
PIN 29 -- AL14
PIN 30 -- AL15
PIN 31 -- AL16
PIN 32 -- AL17
PIN 33 -- AL18
PIN 34 -- AL19
PIN 35 -- AL20
PIN 36 -- AL21
PIN 37 -- AL22
PIN 38 -- AL23
PIN 38 -- AL24
PIN 40 -- GROUND

APPENDIX A Cont'd.

SCSI Connectors

The signal allocation of the SCSI connectors is the same for both, so only one allocation is shown. The only difference is in the way the connector pins are numbered.

PIN 1 -- GROUND
PIN 2 -- GROUND
PIN 3 -- GROUND
PIN 4 -- GROUND
PIN 5 -- GROUND
PIN 6 -- GROUND
PIN 7 -- GROUND
PIN 8 -- GROUND
PIN 9 -- GROUND
PIN 10 -- GROUND
PIN 11 -- GROUND
PIN 12 -- RESERVED
PIN 13 -- OPEN
PIN 14 -- RESERVED
PIN 15 -- GROUND
PIN 16 -- GROUND
PIN 17 -- GROUND
PIN 18 -- GROUND
PIN 19 -- GROUND
PIN 20 -- GROUND
PIN 21 -- GROUND
PIN 22 -- GROUND
PIN 23 -- GROUND
PIN 24 -- GROUND
PIN 25 -- GROUND
PIN 26 -- DATA(0)
PIN 27 -- DATA(1)
PIN 28 -- DATA(2)
PIN 29 -- DATA(3)
PIN 30 -- DATA(4)
PIN 31 -- DATA(5)
PIN 32 -- DATA(6)
PIN 33 -- DATA(7)
PIN 34 -- DATA(PARITY)
PIN 35 -- GROUND
PIN 36 -- GROUND
PIN 37 -- RESERVED
PIN 38 -- TERMINATOR POWER
PIN 39 -- RESERVED
PIN 40 -- GROUND
PIN 41 -- ATTENTION
PIN 42 -- GROUND
PIN 43 -- BUSY
PIN 44 -- ACKNOWLEDGE
PIN 45 -- RESET
PIN 46 -- MESSAGE
PIN 47 -- SELECT
PIN 48 -- COMMAND/DATA
PIN 49 -- REQUEST
PIN 50 -- INPUT/OUTPUT