national accelerator laboratory

TM-539
2340.000
December, 1974

# CAMAC EXTENDED BRANCH SERIAL DRIVER

A. E. BRENNER and RONALD G. MARTIN

FERMI NATIONAL ACCELERATOR LABORATORY*

BATAVIA, ILLINOIS 60510

## Summary

The Extended Branch Serial Driver (EBSD) has been developed to provide a serial extension of the present CAMAC Multi-Crate Branch Highway System. This serial driver provides an interface to the NIM proposed CAMAC Multi-Crate Serial Highway System using an existing Branch Highway and Driver configuration. The EBSD can be operated in either a bit serial or a byte serial mode.

## Introduction

While the CAMAC Branch Highway System has been used quite extensively in the experimental areas at FERMILAB, the evaluation of the new CAMAC Serial Highway renders new application possibilities in the area of experimental data acquisition and monitoring. The proposed CAMAC Serial System Description (TID-26488/A&E#2) specifies the standards to be followed in the implementation of a Serial Highway and a Type L1 Serial Crate Controller. The computer interface however still remains the responsibility of the individual involved in configuring the Serial System to his particular application. Since BISON computer systems[1] presently being used at FERMILAB in the experimental areas employ "standard" computer interfaces and software packages, it is the intention of FERMILAB to use this existing hardware and software to implement new Serial System applications.

Instead of designing and developing a new special purpose serial computer driver, a CAMAC module has been developed which extends the existing CAMAC Branch Highway System into a Serial Highway configuration. This Extended Branch Serial Driver (EBSD) can be placed in any crate of an existing CAMAC Branch Highway System. The EBSD provides its user the flexibility of performing single operations on the Serial Highway or else may be "microprogrammed" to perform numerous autonomous serial operations. A typical crate arrangement is illustrated in Figure 1.

## Operation Description

Serial messages are initiated by the EBSD in response to CAMAC dataway operations involving command and data registers. The module is set to a predefined state upon power up conditions. Whenever the EBSD is not currently involved in generating serial operations, WAIT bytes are transmitted by the EBSD to "sync" the Serial Highway System and determine if any demands are pending.

Transverse and longitudinal parity are checked on all incoming messages and if enabled, a LAM will be generated for either of these parity error conditions.

The EBSD has been designed to function in four different operational modes (being A, B, C, & D). Mode A is basically a programmed data transfer/command mode. Mode B is defined as a block transfer mode for a serial system having the same module distribution in a multi-crate configuration. Mode C is a block transfer mode where the serial highway commands are loaded into the FIFO memory of the EBSD. The reply messages, along with demand messages, from the serial highway are also loaded into the FIFO memory. In mode D, the operation is the same with the exception that demand messages are not loaded into the FIFO memory. A detailed description of these various operational modes follow:

### Mode A

Seven registers are used to provide control and communication between the Serial Highway and the CAMAC dataway. These registers are the Command, Write Data, Status #1, Demand, LAM, and LAM Mask. The bit identification and the CAMAC function codes for these registers are shown in Figure 2 and Table 1 respectively.

A serial message transmission is initiated by:

1. Writing the Command Register with a serial read or command function, or
2. Writing the Write Data Register while the Command Register contains a serial write function.

It should be noted that loading the Command Register with a serial write command does not initiate a serial transmission.

After a serial message transmission has been initiated, the EBSD will transmit a fixed number of SPACE bytes to the Serial Highway to enable the addressed crate and module to return the required reply message. The receipt of a reply message by the EBSD will be acknowledged by either the generation of a LAM, if the RDY bit of the LAM register has not been masked, or else by reading the status of the LAM register (see Table 2).

Serial read data is retrieved by performing a F(Ø)·A(Ø) operation on the EBSD after a reply message has been received. The serial X and Q signals are translated to the dataway X and Q lines while performing this read operation. The serial X and Q signals for non-read operations are retrieved by performing a F(1)·A(Ø) operation on Status #1 Register of the EBSD. Programmed utilization of the Serial Highway usually involves the transmission of a serial message and a wait for the receipt of a reply message before the transmission of a new serial message is initiated. Any attempts to write the Data or the Command Registers while the transmitter is busy will not be allowed and a Q=Ø response will be returned by the EBSD. If a time of one millisecond elapses between the start of a serial message transmission and the receipt of a reply message, the TIMEOUT signal is asserted which sets the TMO bit of the LAM register. Any reply message received after TIMEOUT has been asserted will be ignored.

A Q-Scan mode for read operations can be performed if the MR bit of the Command Register is set to "1". If the MQ bit of this same register has been cleared (equal to "Ø"), an F(Ø)·A(Ø) command to the EBSD will cause another serial message to be transmitted using the same serial Command Register values that were used for the previous serial message. If MQ=1, the new serial message will have the value of SA, and possibly the value of SN, will be changed depending on the status of the SQ signal from the previous serial operation. If SQ=1 for the previous operation, the value of SA will be incremented. Had SQ=Ø for the last serial operation, then SA will be reset and the value of SN will be incremented for the new serial message being generated. If the internal N-counter advances beyond N=32 in this mode, the crate address is cleared and any subsequent serial transmission with CRATE=Ø will cause the ADNR bit of the LAM Register to be set upon receipt of the reply message.

In addition to the seven register used by the EBSD to control and communicate with the Serial Highway, there are four condition flags which can be tested by the computer for Q response to determine the operational status of the Serial Highway System. The identification of these four bits, along with their CAMAC test codes are shown in Table 3.

If an unrecognized command message is received (M1 + M2 = Ø), the unrecognized crate address is stored in the lower 6 bits of Status Register #1 and the upper 6 bits are cleared.

Mode B

The command, Write Data and the Read Data Registers are not used in this mode and the serial operation is now controlled by the use of Parameter #1 and #2 Registers. In this mode of operation either write or read data is stored in the FIFO memory. In addition to these two Parameter Registers, a second Status Register (Status #2) and an ERROR Register are available to the user to indicate the operational status of the Serial Highway while the EBSD is functioning in this mode. These additional registers have been shown in Figure 3.

The initial value of the serial crate address, the station number and the station subaddress, along with the serial function code to be performed, are loaded into Parameter #2 Register. A serial message, as determined by the function code loaded in Parameter #2, will be re-transmitted after a reply has been received for the previous message. The EBSD will sequentially increment the values of the station sub-address, the station number and the serial crate address dependent upon the incrementing information which is loaded into Parameter Register #2. In addition to the incrementing information, a bit (QM) is used to control the format by which the EBSD will change the values of SA, SN, and SC. If QM=1, then the value of SA (sub-address) will be continuously incremented for each new serial message while the last reply message contained SQ=1. Whenever SQ=Ø, the SA is set to the initial starting value (Parameter #2) and the value of SN is incremented. The value of SN will continue to be incremented until the programmed number of increments have occurred.

If QM=Ø, the incrementing operation of SA, SN, and SC is the same as described above with the exception that SA will be continuously incremented for the programmed number of times as determined by the values of Parameter Register #1.

The serial operation is initiated by either performing an F(25)·A(Ø) command to the EBSD or else setting the EX bit of Parameter Register #2.

If the serial function is a write, a sufficient number of data words must have been previously loaded into the FIFO memory by an F(16)·A(7) command. Had the serial function been a read operation, the data information of the reply messages will be stored in the FIFO and can be read out by the computer using either an F(Ø)·A(7) or an F(2) command. Reply status information is not loaded into the FIFO memory in this mode of operation.

If the FIFO becomes empty on a serial read operation prior to all of the programmed transfers being made, further generation of additional serial message will be terminated by the EBSD. A LAM will be generated, if not masked, indication that the FIFO has been emptied on serial writes and has been filled during serial reads.

After the block operation has been completed, the user can read Status Register #2 to get summary information concerning the last block transfer. This register contains fields which indicate the first crate address, the number of serial transfer that have taken place and a summary bit that indicates if any errors have been detected during the block operation.

Mode C

In this mode, neither the Command nor the Parameters Registers are used to specify the type of serial message to be generated. The serial commands are stored in the FIFO memory in the order of their intended execution. If write commands are included, the write data must immediately follow each write command. As serial commands are executed, the replies to these commands are loaded into the FIFO. If the reply message contains read data, this data is loaded in the next word location immediately following the reply information.

Along with the reply message and read data information, any demand messages which are received by the EBSD can also be loaded into the FIFO.

This demand storage option in the FIFO is enabled by an F(26)·A(Ø) command and is disabled by an F(24)·A(Ø) command.

The command, reply and demand message word format for the FIFO memory are shown in Figure 4.

A maximum of 255 words can be used for each block transfer. One word is reserved for a delimiter word which is inserted by

hardware upon the receipt of an initiate Mode C operation. This delimiter word will be used by the EBSD to indicate when the last serial message has been generated. Mode C operation is initiated by performing an F(25)·A(1) command on the EBSD.

## Mode D

This mode of operation is identical to Mode C, except that demand messages are not stacked into the FIFO memory during the normal block operation. This allows data from as many as 255 input sources to be stored in the FIFO. Demand messages may be stored in the FIFO after the completion of a block transfer, if demand storage in the FIFO has been enabled and the FIFO is not full.

Mode D serial operation is initiated by performing an F(25)·A(2) command on the EBSD.

## Module Description

The EBSD is configured as a six wide CAMAC module. Three of the CAMAC cards are used to provide the interface and control logic for the Serial Highway, two of the cards provide the CAMAC interfacing and control logic and the sixth card is a 256 word x 24 bit FIFO memory. The interface with the Serial Highway is made via two 25 contact front panel connectors. An auxiliary 19 contact front panel connector is provided to enable the user to monitor special internal logic functions and timing signals. The operational status, error conditions, and demands on the serial highway are indicated by front panel LED's.

The output clock of the serial highway can be controlled by either an internal crystal or variable frequency oscillator or else by an external source.

A rear panel switch enables the selections of bit serial or byte serial mode of operation.

## Conclusion

A prototype of the EBSD[2] and a Serial Crate Controller Type L1[3] have been tested at FERMILAB. These units were operated in both the bit serial and byte serial modes at clock frequencies up to 5MH$_z$. Extensive studies into error rates as a function of operating frequencies and cable length are now in progress. Production units are currently being manufactured and are scheduled for delivery prior to the end of this year.

## References

*Operated by the Universities Research Association, Inc. under contract with the United States Atomic Energy Commission.

BISON is a term applied to the standard supported on-line computing systems at FERMILAB.

[2]EBSD Unit, Designed and Built by Kinetic Systems per FERMILAB specifications.

[3]Serial Crate Controller, SCC-L1, Designed and Built by Jorway Corporation per FERMILAB Specifications and TID-26488

Figure 1

CAMAC Crate Configuration

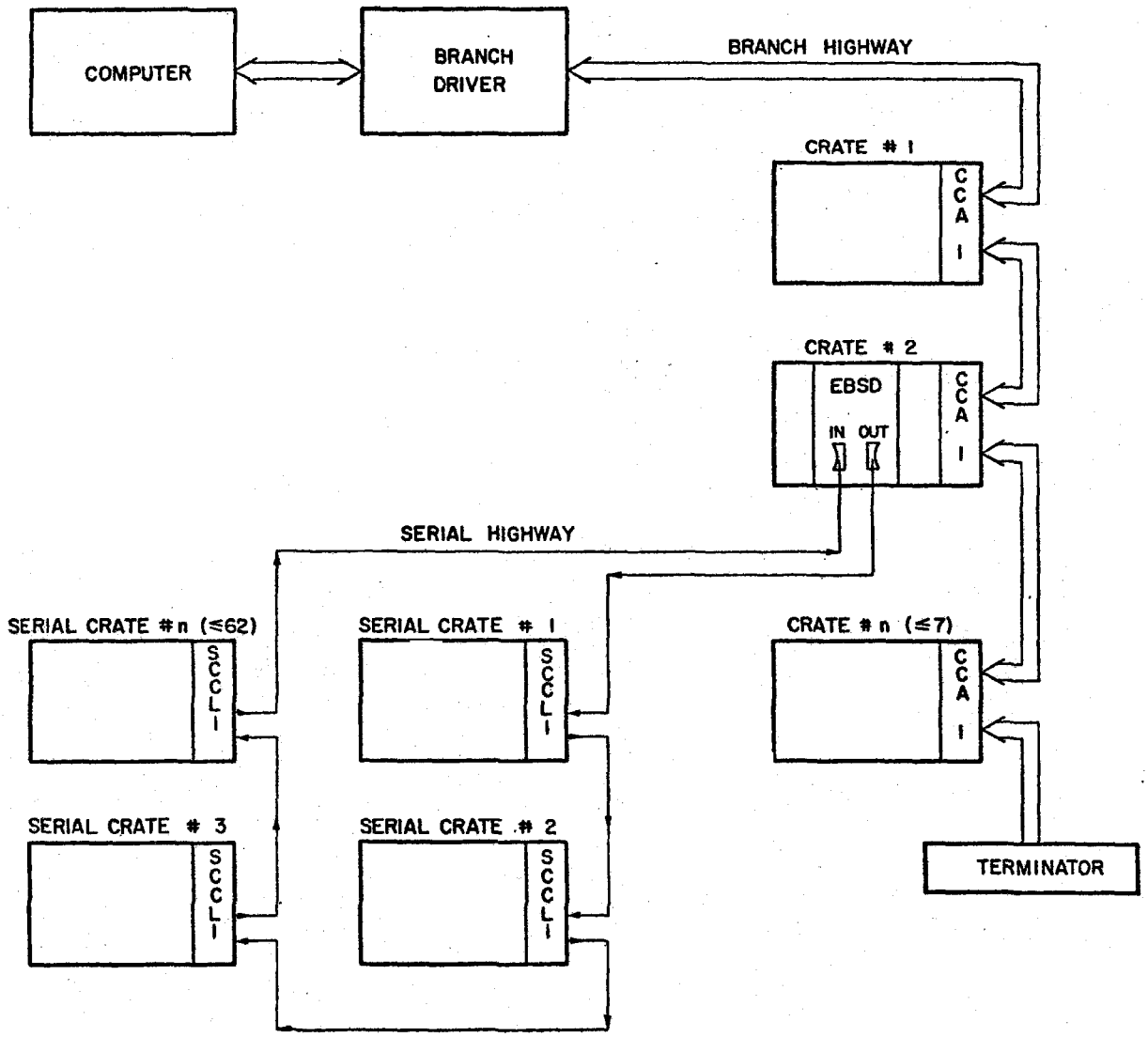


Figure 2

General Purpose Registers

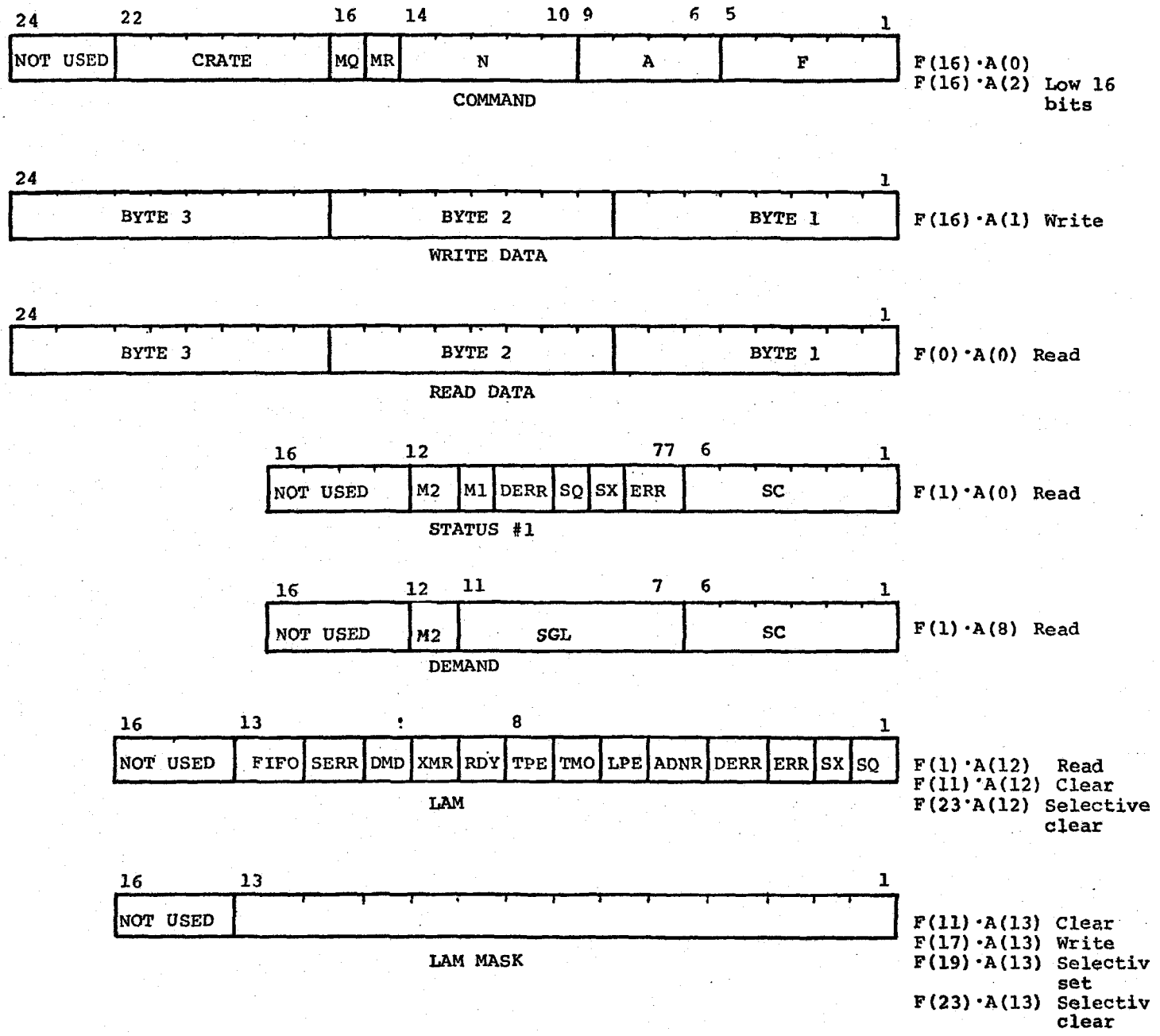
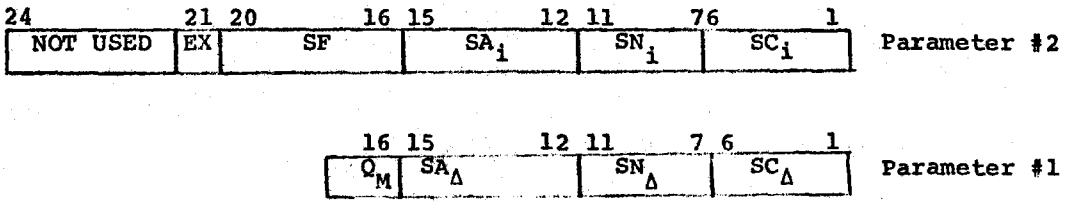


Figure 3

Special Block Operation Registers




$Q_M = \emptyset$ $\quad SA_{total} = SA_i + SA_\Delta$

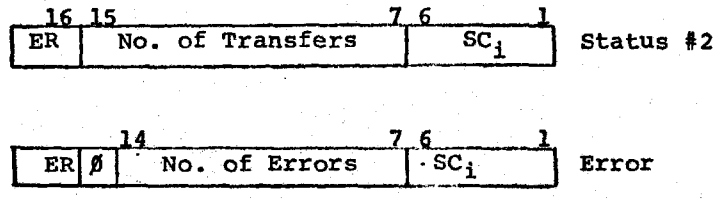
$SN_{total} = SN_i + SN_\Delta$

$SC_{total} = SC_i + SC_\Delta$

$Q_M = 1$ $\quad SA_{total} = SA_i + \Delta(SQ=1)$

$SN_{total} = SN_i + SN_\Delta$

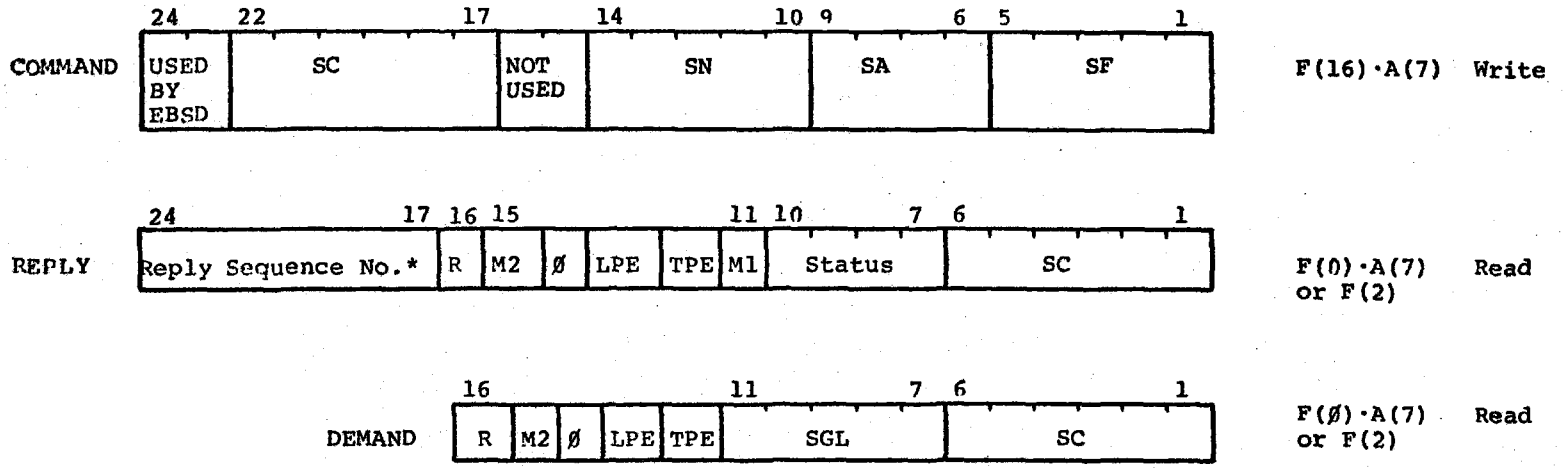
$SC_{total} = SC_i + SC_\Delta$




ER = ERR + TPE + LPE + ADNR + TMO

Figure 4

FIFO Memory Word Format




*Valid only for Reply messages, where M2 = Ø, M1 = 1.

| R | M2 | |
|---|---|---|
| 0 | 0 | Non-read reply |
| 0 | 1 | Demand |
| 1 | 0 | Read-reply, data follow |

| FUNCTION CODE | Q | X | ACTION |
|---|---|---|---|
| F(0)·A(0) | SQ | SX | Gates serial read data onto Dataway. |
| F(0)·A(7)+F(2) | OR | 1 | Dumps FIFO memory output word onto Dataway. |
| F(1)·A(0) | $\overline{CMD}$·SHR | 1 | Gates Status register #1 onto Dataway. |
| F(1)·A(1) | RPL | 1 | Gates Status register #2 onto Dataway. |
| F(1)·A(5) | RPL | 1 | Gates Error Status register onto Dataway. |
| F(1)·A(8) | 1 | 1 | Gates Demand register onto Dataway. |
| F(1)·A(12) | $\overline{CMD}$·SHR | 1 | Gates LAM register onto Dataway. |
| F(8)·A(15) | L | 1 | Returns Q=1 if L signal is present. |
| F(9)·A(0)·S1 | 0 | 1 | Clears (aborts) block-mode operation. |
| F(9)·A(7)·S1 | 0 | 1 | Clears FIFO memory. |
| F(11)·A(12)·S1 | 0 | 1 | Clears LAM register. |
| F(11)·A(13)·S1 | 0 | 1 | Clears LAM mask register. |
| F(16)·A(0)·S1 | $\overline{CMD}$ | 1 | Writes Serial Command buffer. |
| F(16)·A(1)·S1 | $\overline{CMD}$ | 1 | Writes Serial Write-data buffer. |
| F(16)·A(2)·S1 | $\overline{CMD}$ | 1 | Writes lower 16 bits of Serial Command buffer. |
| F(16)·A(7)·S1+F(18) | IR | 1 | Loads word into FIFO memory. |
| F(16)·A(2)·S1 | $\overline{CMD}$·SHR | 1 | Writes Mode B parameter word #1. |
| F(17)·A(3)·S1 | $\overline{CMD}$·SHR | 1 | Writes Mode B parameter word #2. |
| F(17)·A(13)·S1 | 1 | 1 | Writes LAM mask. |
| F(19)·A(13)·S1 | 1 | 1 | Selectively sets LAM mask. |
| F(23)·A(12)·S1 | 1 | 1 | Selectively clears LAM register. |
| F(23)·A(13)·S1 | 1 | 1 | Selectively clears LAM mask. |
| F(24)·A(0) | 0 | 1 | Disables FIFO storage of demand messages. |
| F(25)·A(0)·S1 | $\overline{CMD}$·SHR | 1 | Initiates Mode B operation. |
| F(25)·A(1)·S1 | $\overline{CMD}$·SHR | 1 | Initiates Mode C operation. |
| F(25)·A(2)·S1 | $\overline{CMD}$·SHR | 1 | Initiates Mode D operation. |
| F(26)·A(0) | 0 | 1 | Enables FIFO storage of demand messages. |
| F(27)·A(0) | $\overline{CMD}$ | 1 | Returns Q=1 if transmission of output message has been completed. |
| F(27)·A(1) | $\overline{CMD}$·SHR | 1 | Returns Q if $\overline{CMD}$ is true and if a reply has been received or a timeout has occurred since initiation of previous output message. |
| F(27)·A(2) | SYNC | 1 | Returns Q if receiver sync signal SYNC is true. |
| F(27)·A(4) | DMD | 1 | Returns Q=1 if demand LAM is set. |
| F(27)·A(5) | IR | | Test FIFO status. |
| Z·S2 | - | - | Clears LAM & LAM Mask registers and FIFO buffer memory; halts any block-mode operations. |

Notes: For Modes B, C, and D, SHR is true following receipt of the last reply.
IR = 1 FIFO Input ready
= Ø FIFO full

Table 1

CAMAC Function Code

| Bit | Label | Type* | Description |
|---|---|---|---|
| 13 | FIFO | A | Is set when FIFO is empty on write or when FIFO is full or read. Is cleared when block operation is initiated. |
| 12 | SERR | A | Indicates that either bytes are not being received or that the receiver has lost byte sync. Set by the asertion of SYNC. |
| 11 | DMD | A | Indicates that a demand message was received. |
| 10 | XMR | A | Indicates that the serial transmitter is ready. It is set by the assertion of $\overline{CMD}$. |
| 9 | RDY | A | Indicates that a serial command has been transmitted and that either the reply message has been received (or command message if the crate address was unrecognized) or that the timeout period has elapsed. It is set by the assertion of $\overline{CMD} \cdot SHR$ |
| 8 | TPE | A | Indicates that a transverse parity error was detected in a reply message. It is not set if a transverse parity error occurs on a wait byte. |
| 7 | TMO | A | Indicates that a preset time has elapsed, normally 1 msec, from the start of a serial message transmission without receipt of a reply. It is set on assertion of TIMEOUT. |
| 6 | LPE | S | Indicates that a longitudinal parity error was detected in the reply message. |
| 5 | ADNR | S | Indicates that the crate address was not recognized by any crate on the serial highway. This condition is recognized by M1 and M2 both being false in the reply message ($\overline{M1} \cdot \overline{M2}$ = TRUE). |
| 4 | DERR | S | Indicates that an error was detected by the addressed SCC in the previous message addressed to it. |
| 3 | ERR | S | Indicates that an error was detected in the command message by the addressed SCC: $\overline{SQ}$ and $\overline{SX}$ will normally be set along with ERR since a detected error inhibits the Dataway cycle. |
| 2 | $\overline{SX}$ | S | Indicates X = Ø response to the serial message. This bit also is not set if $\overline{M1} \cdot \overline{M2}$ is true in the reply. |
| 1 | $\overline{SQ}$ | S | Indicates Q = Ø response to the serial message. The bit is not set if $\overline{M1} \cdot \overline{M2}$ is true in the reply. |

* The 13 LAM bits may be classed as synchronous (S) and asynchronous (A). The synchronous bits are set or not set depending upon the status of signals at the receipt of a reply message. Asynchronous bits are set as events occur.

Table 2

LAM Register

| Binary Flag | Condition | Test Command |
|---|---|---|
| CMD | Indicates that a serial command message is in the process of being transmitted. | F(27)·A(0)<br>Q = $\overline{\text{CMD}}$ |
| SHR | SHR = RPL + TIMEOUT, where<br><br>RPL indicates that a message other than a demand message has been received since the last serial transmission.<br>TIMEOUT indicates that 1 msec (nominal) has elapsed since the last serial transmission without receipt of a reply or unrecognized command message. | F(27)·A(1)<br>Q = SHR·$\overline{\text{CMD}}$ |
| SYNC | Indicates that the receiver is receiving bytes and has acquired byte sync (if in bit serial mode). If more than four bytes are transmitted (including wait bytes) without receipt of a byte, SYNC becomes false. | F(27)·A(2)<br>Q = SYNC |
| DMD | Indicates that a demand message has been received and that the demand LAM source bit is set. | F(27)·A(4)<br>Q = DMD |

Table 3

Condition Flags