Space Sttenc
UW SSEC Publication
No. 90.09.MI

McIDAS

Man computer Interactive Data Access System

Ingestor Protocol Manual

Issued September 1990

Space Science and Engineering Center
University of Wisconsin - Madison
1225 West Dayton Street
Madison, WI 53706
Telephone 608-262-2455
TWX 608-263-6738

Reader Response

We would like your comments on our manual. Please take a moment to fill out this self-mailing form and return it to SSEC - McIDAS User Services and Documentation. Thank you.

Manual name (from the title page)

_____

Did you find this manual helpful? Did it provide the required information?

_____

_____

_____

Did you find errors or omissions in the manual? Please be very specific.

_____

_____

_____

Please rate this manual on the following items from 1 to 5, with 1 as low and 5 as high.

Text	1	2	3	4	5
Graphics	1	2	3	4	5
Ease-of-use	1	2	3	4	5
Overall	1	2	3	4	5

What is the best feature of this manual?

_____

_____

_____

What is the worst feature of this manual?

_____

_____

_____

Your name and title _____

Date _____

McIDAS site _____

Address _____

_____

Telephone _____

Check here if you would like a written reply. ___

Fold Here

Fold Here

Space Science and Engineering Center
McIDAS User Services and Documentation, Room 611
University of Wisconsin - Madison
1225 West Dayton Street
Madison, WI 53706

Ingestor Protocol Manual

Introduction 1-1
What's in this Manual 1-1
Before Using this Manual 1-1
Related Manuals 1-2

General Protocol 2-1
Control Fields 2-1
Command Chains 2-2
Command Chain Codes 2-3
SYNC Command 2-4
WRITE Command 2-5
CLEAR Chain 2-6
SETUP Chain 2-6
NEWDOC Chain 2-7
DATA Chain 2-8
READ Command 2-9
NOP Command 2-9

GVAR Block 11 Ingestor Protocol 3-1
Control Fields 3-1
Command Chains 3-2
Command Chain Codes 3-3
SYNC Command 3-3
WRITE Command 3-4
CLEAR Chain 3-4
NEWDOC Chain 3-4
DATA Chain 3-6
READ Command 3-8

GVAR Imager Ingestor Protocol 4-1
Control Fields 4-1
Command Chains 4-4
Command Chain Codes 4-5
SYNC Command 4-6
WRITE Command 4-7
CLEAR Chain 4-8
NEWDOC Chain 4-8
DATA Chain 4-10
READ Command 4-12

GMS Ingestor Protocol 5-1
Control Fields 5-1
Command Chains 5-3
Command Chain Codes 5-4
SYNC Command 5-5
WRITE Command 5-6
NEWDOC Chain 5-7
DATA Chain 5-9
READ Command 5-10

AAA Block 1 Ingestor Protocol 6-1
Control Fields 6-1
Command Chains 6-3
Command Chain Codes 6-3
SYNC Command 6-4
WRITE Command 6-5
CLEAR Chain 6-5
NEWDOC Chain 6-5
DATA Chain 6-7
READ Command 6-8
NOP Command 6-8

AAA MARK II Ingestor Protocol 7-1
AAA Raw Data Formats 7-1
Command Chains 7-2
Command Chain Codes 7-2
SYNC Command 7-3
Scan Line Number 7-3
Tracking Discrepancies 7-4
WRITE Command 7-5
DOC Chain 7-6
DATA Chain 7-7
COMDOC Chain 7-10
NOP Command 7-12
READ Command 7-12
IDLE Command 7-12
WAIT Command 7-12
Common Doc 7-13
Section 1 7-13
Section 2 7-13
Section 3 (Mode AA only) 7-14

METEOSAT PDUS Ingestor Protocol 8-1
Control Fields 8-1
METEOSAT Signal Formats 8-3
Subframes 8-3
Header Subframes 8-3
Image Data Subframes 8-5
Annotation Subframes (B Type Format Only) 8-6
Conclusion Subframes 8-6
Command Chains 8-7
Command Chain 1 8-7
Command Chain 2 8-7
Command Chain 3 8-8
IDLE Command 8-9
SYNC Command 8-9
WAIT Command 8-9
LABEL Command 8-9
WRITE Command 8-11
DIRECT WRITE 8-11
REVERSE WRITE 8-12

POES Ingestor Protocol 9-1
Control Fields 9-1
Command Chains 9-2
IDLE Command 9-3
SYNC Command 9-3
READ Command 9-3
WRITE Command 9-3
COMDOC Chain 9-5
COMMAND Chain 9-10
DATA Chain 9-11

Introduction

This manual describes the protocol for communication between the mainframe ingestor handler and the ingestor hardware. It is designed as a reference for both software and hardware technicians responsible for maintaining satellite data ingestor systems.

What's in this Manual

This *Ingestor Protocol Manual* includes protocol information for all SSEC designed ingestors. Each site has a unique combination of one or more ingestors, but no site has all ingestors operational. The information pertaining to ingestor systems not used at your site will be helpful if your site adds that ingestor.

Background information for all modern i.e., new generation SSEC designed satellite systems is found in the General Protocol chapter of this manual. This chapter outlines protocol as the command/response communication between the intelligent hardware and software subsystems of an ingestor system. The remaining chapters discuss individual ingestors and their unique features. A description of the control fields specific to each ingestor is included in these chapters.

Ingestors that existed before the General Protocol format was established, i.e., the AAA GOES, POES and METEOSAT PDUS ingestors, remain in their earlier formats and do not follow the General Protocol format at this time.

Before Using this Manual

In order for this manual to be useful, you should be familiar with the satellite data ingestor system used by McIDAS and have a working knowledge of one or both of these areas:

- ingestor hardware
- the mainframe ingestor handler

Protocol is essentially the "bridge" between hardware and software. If you understand the mainframe ingestor handler, this manual will tell you what to expect from the ingestor hardware, and vice versa.

Related Manuals

For additional information on the satellite data ingestor system used by McIDAS, refer to the manuals listed below.

- *McIDAS Ingestors Manual*
- *McIDAS Operator's Manual*

General Protocol

This chapter describes the General Protocol for communication between the mainframe ingestor handler and the ingestor hardware. All new generation ingestor protocols are a subset of this General Protocol format. Specific format details are described separately for each hardware ingestor card in the chapters that follow. Refer to them by ingestor name.

An ingestor hardware card acts as a number of buffers, each containing blocks of data broadcast specifically for that ingestor. The ingestor hardware informs the mainframe of these blocks as they accumulate in sets. The length of these sets differs for each ingestor and is defined in the following chapters. The mainframe may or may not request data from a particular block. In either event, the card is informed when the block is no longer needed and may be cleared from memory.

The ingestor protocol is a series of command chains executed by these commands:

- SYNC
- WRITE
- READ
- NOP

Control Fields

The software ingestor handler needs to know the type of signal and the quality of the signal an image is composed of before it can proceed to ingest or ignore an image. Each satellite sends control fields containing the satellite status along with various other bytes to describe the signal currently being transmitted. This information usually resides in the Common Documentation section of the signal, though some control status is also sent to the ingestor handler by the ingestor hardware.

The Common Documentation format and content varies from satellite to satellite. Each ingestor handler has a specific section of code which decodes the control fields in order to determine the signal quality or to check against ingestor request parameters.

The definitions of the specific control fields as used by the ingestor software and provided by the satellite or the ingestor hardware, are found in each ingestor specific chapter of this manual.

Command Chains

A command chain is a series of desired actions that a host processor sends to the channel processor for execution. Figure 1 below shows a sample command chain for modern SSEC ingestors. To keep the channel busy and prevent data loss, the host processor always queues two channel command chains to the channel. When a chain is completed, the NEWDOC WRITE-READ command pair provides the mainframe with the information it needs to create and queue the next chain. The commands that acquire the data described in NEWDOC, also a WRITE-READ pair, are sent after the two chains already queued and acquiring data described in the previous NEWDOC are completed. Thus, it is necessary to buffer at least four scans:

- One scan is waiting.
- A second scan is read by the mainframe.
- A third scan is loaded from the signal.
- The fourth scan is held in reserve in case the CPU slows down and the ingestor handler needs to catch up with the hardware.

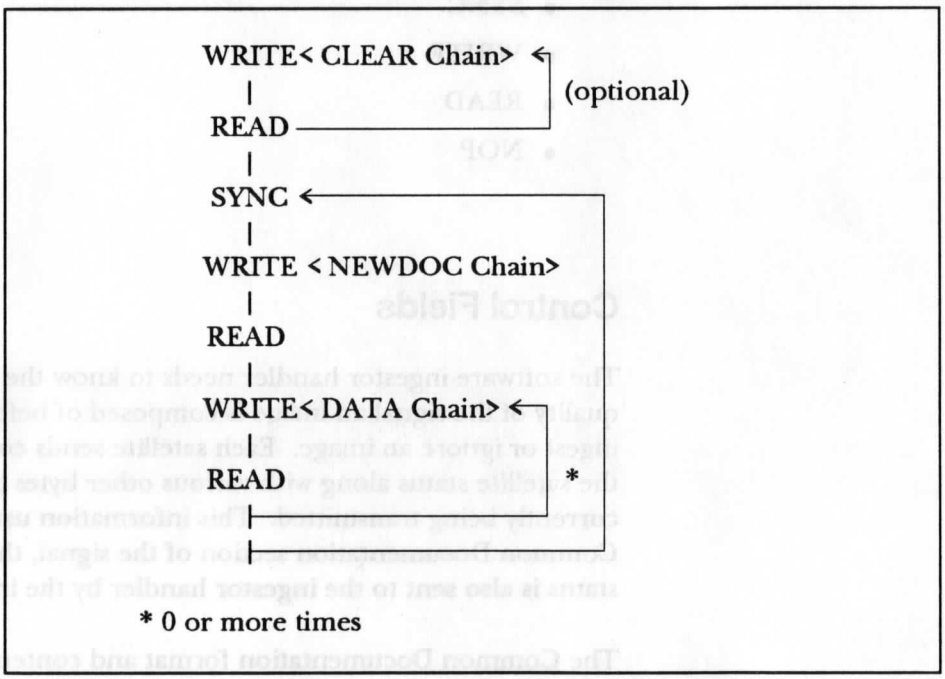


Figure 1. Sample Command Chain

Command Chain Codes

The General Purpose Channel Interface (GPCI) acts as an interface between the channel and the ingestor hardware. The command chain codes are not transmitted from the channel. The GPCI recognizes the various command chains and generates the command chain codes. These codes are transmitted to the ingestor hardware as the command type. Throughout the remainder of this document, the command type is considered the first byte of the command. From an ingestor software or channel perspective, the second byte of a command is the beginning of the command.

SYNC Command

The SYNC command (07) controls the timing. It provides the synchronization that keeps the mainframe and ingestor card programs in agreement. It also reduces mainframe channel traffic when no signal is present. The SYNC command response time varies, depending on the presence of a signal. These response times are explained below. Specific response time information can be found in the individual ingestor chapters.

Response Time	Reason
Immediate	There are unreported scans in the buffers. The ingestor hardware clears the Busy and Unit Exception status flags.
Short	There are no unreported scans, but unread data exists; either the new scan is completed or the scan is completed with old data. The BlkInf sets (the bytes containing block information to guide a data request) that identify old data in the next WRITE-NEWDOC response are set to zero. The ingestor hardware clears the Busy and Unit Exception status flags.
Long	There are no unreported blocks or unread data in the buffers; either new data arrives during the time allotted for the ingestor or a Unit Exception (abort chain) occurs meaning there is no data. The ingestor hardware responds by clearing the Busy flag and either clearing the Unit Exception flag (new data) or setting the Unit Exception flag (no new data).

The GPCI monitors each ingestor's status flags. Upon detecting a cleared Busy flag, the GPCI uses the status of the Unit Exception flag to indicate command success or failure. If the Unit Exception flag is set, a command fail is transmitted. If the Unit Exception flag is cleared, a successful completion is transmitted.

WRITE Command

The WRITE command (01) may consist of 2 to 21 bytes. The first byte (01 - GPCI generated) identifies the command as the WRITE command and the remaining bytes as WRITE command parameters. The WRITE command has four possible chain formats: CLEAR, SETUP, NEWDOC and DATA. In each command, the four high order bits of the second byte signify the command code (chain format). The hexadecimal values of the codes and their meanings are listed below.

Command Chain	Code	Meaning
CLEAR Chain	1*x*	clears all buffers
SETUP Chain	2*x*	performs setup using the parameters that follow
NEWDOC Chain	0*x*	NEWDOC with validity code
NEWDOC Chain	8*x*	NEWDOC without validity code
DATA Chain	4*y*	DATA with validity code
DATA Chain	C*y*	DATA without validity code

For WRITE-CLEAR, WRITE-SETUP and WRITE-NEWDOC commands, the four low order bits of the second byte are represented by an *x*. Recall that *x* represents the don't care condition (i.e., a one or zero has the same effect on the overall code).

For WRITE-NEWDOC and WRITE-DATA commands, an optional validity code may be included as the first 4 bytes of the returned data or common documentation. On a WRITE-NEWDOC command the validity code is always requested, although the option to not use it is available. Channel limitations sometimes necessitate asking for data in more than one WRITE-DATA command. After the first WRITE-DATA command, subsequent WRITE-DATA commands should use the option not to send the validity code.

In an image area line, the validity code comprises only the first 4 bytes of the line. If more than one command is sent, dividing the line into pieces, the validity code will occur in the middle of the image line if it was sent as the first four bytes of each piece of returned data. Therefore, to avoid writing the validity code in the middle of the line, the validity code is requested with the first piece of data, but not requested with subsequent pieces.

For WRITE-DATA commands, the four low order bits of the second byte signify the sample/average data process instructions and are represented by the *y* in the command code. The hexadecimal values for process instructions and their meanings are listed below.

Process Instruction	Meaning
0	Full Resolution
1	Undefined
2	Sample by 2
3	Sample by 3
4	Sample by 4
5	Sample by 6
6	Sample by 8
7	Sample by 12
8	Sample by 16
9	Average by 2
A	Average by 3
B	Average by 4
C	Average by 6
D	Average by 8
E	Average by 12
F	Average by 16

The CLEAR, SETUP, NEWDOC and DATA chains are outlined below. The command, its parameters, response and field definitions are listed.

CLEAR Chain

Command: WRITE, 1*x*H

Response: None. The CLEAR chain clears the hardware buffers. It is optional and has no corresponding READ command.

SETUP Chain

Command: WRITE, 2*x*H, P1, P2, P3, P4 (1 byte each)

Parameter: P# is the parameter number (to be defined).

Response: None. The SETUP chain performs the setup using the four 1-byte parameters given in the command line.

Response Fields: P1 - P4 are the 1-byte setup parameters.

NEWDOC Chain

Command: WRITE, 0*x* or 8*x*H, VC (4 bytes)

Response: VC (4 bytes; optional), GenStat (10 bytes), BlkInf*1*...BlkInf*m* (3 bytes each), Doc (*n*)

Response Fields:

VC	is the Validity Code.
GenStat	is the frame sync and auxiliary status.
BlkInf*1-m*	contains the buffer address of each of the *m* blocks of data.
Doc *(n)*	is the common documentation, or Block Header plus some or all of the common documentation; *n* depends on the size of the common documentation.

DATA Chain

Command: WRITE, 4*x*H or C*x*H, BfId (1 byte), HdOfst (2 bytes)*, LdOfst (2 bytes)*, DtOfst (2 bytes)*, Length (2 bytes), XferTyp (1 byte), TimFlg (1 byte), TimVal (8 bytes)

* lower byte first

Parameters:

BfId — is the Buffer ID of the block.

HdOfst — is the offset into the header if a buffer wrap occurs on the mainframe.

LdOfst — is the offset to the start-of-line documentation in a block.

DtOfst — is the offset into the data field.

Length — is the number of words to transfer.

XferTyp — is the bits per word and how to transfer them.
0= zero-masked 6-bit data
1= shifted 6-bit data
2= unshifted 6- and 8-bit data
3= shifted 10-bit data
4= shifted 12-bit data
5= 16-bit data

TimFlg — is the time flag set when a block has no Block 0 time associated with it. If the Doc Available flag in BlkInf equals zero, the time sent back on a data request is taken from TimVal.

TimVal — is the time generated by the ingestor handler software.

Response: VC (4 bytes; optional), Data (*n*)

Response Fields:

VC — is the Validity Code.

Data (*n*) — is the header and data, or the header and doc-as-data. *n* depends on the signal's documentation size or the amount of data requested.

READ Command

The READ command (02) transfers the results of the WRITE commands. It carries the data from the hardware buffer to the mainframe channel.

NOP Command

The NOP command (03) is the null command. No operation is performed. It is sent from the host processor to the channel processor at the beginning of some command chains.

GVAR Block 11 Ingestor Protocol

This chapter describes the protocol for communication between the mainframe ingestor handler and the GVAR Block 11 ingestor hardware. The Block 11 ingestor card uses the new generation ingestor hardware format.

The Block 11 ingestor card acts as a number of buffers. When filled, a buffer contains an entire Block 11 block. The Block 11 ingestor hardware informs the mainframe of these blocks, five at a time. The mainframe may or may not request data from a particular block.

The Block 11 card contains a 512K byte memory area that is treated like a circular queue. A block's worth of data makes up an arbitrary buffer. The firmware contains pointers that point to the start of these buffers. The firmware allocates space for up to 128 buffer pointers. The length of the buffer is in the block's header information. Blocks which cross the 512K boundary continue at an address of zero. All data is stored as 8-bit bytes with 10-bit data stored in 2 bytes.

The Block 11 protocol is a series of command chains executed by three commands: SYNC, WRITE and READ.

Control Fields

The following list defines the GVAR Block 11 control fields which are presented by the satellite or the ingestor hardware for use by the ingestor software.

Block Type describes each of the 22 available block types. Block type is used in matching ingestor requests.

Buffer IDs indicate where in the ingestor hardware's memory the current data resides. There is one buffer ID per block of data. It is used to construct a channel command that asks for data.

Number of Blocks tells how many blocks of data the hardware has waiting. There can be up to 10 for Block 11. This value controls the chain creation for the loops of data request commands.

Satellite ID is the number used to match a request window. Satellite ID should not change throughout an image.

Word Size is used by the ingestor handler to determine how many bytes of data to request.

Command Chains

A command chain is a series of desired actions that a host processor sends to the channel processor for execution. Figure 2 below shows a sample command chain for the Block 11 ingestor. To keep the channel busy and prevent data loss, the host processor always sends two channel command chains to the channel. When a chain is completed, the NEWDOC WRITE-READ command pair provides the mainframe with the information it needs to create and send the next chain. The commands that acquire the data described in NEWDOC (up to ten WRITE-READ pairs) are sent after the two chains already queued and acquiring data described in the previous NEWDOC are completed. Thus, it is necessary to buffer at least four sets of ten blocks:

- One set is waiting.
- A second set is read by the mainframe.
- A third set is loaded from the signal.
- The fourth set is held in reserve in case the CPU slows down and the ingestor handler needs to catch up with the hardware.

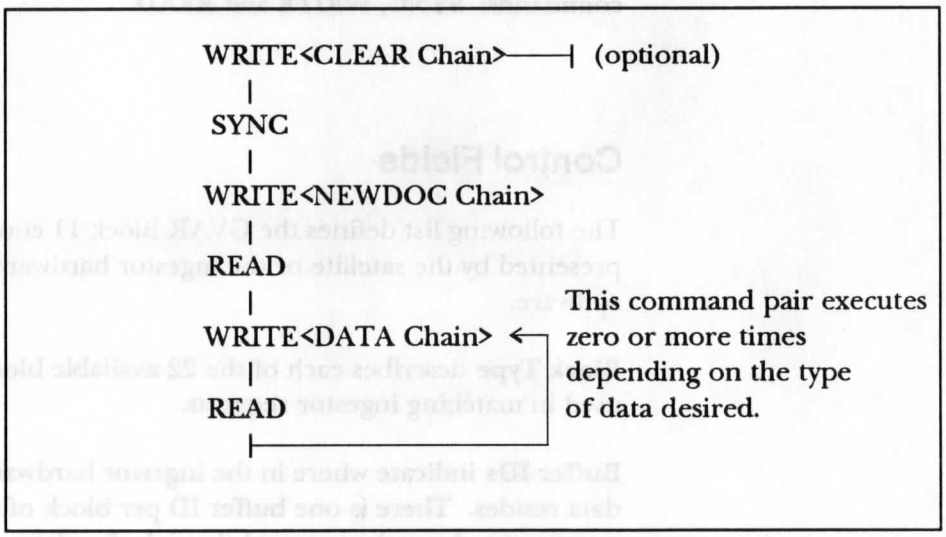


Figure 2. Sample Block 11 Command Chain

Command Chain Codes

The General Purpose Channel Interface (GPCI) acts as an interface between the channel and the ingestor hardware. Command chain codes are not transmitted from the channel. The GPCI recognizes the various command chains and generates the command chain codes. These codes are transmitted to the ingestor hardware as the command type. Throughout the remainder of this chapter, the command type is considered the **first** byte of the command. From an ingestor software or channel perspective, the second byte of a command is the beginning of the command.

SYNC Command

The SYNC command (07) controls the timing. It provides the synchronization that keeps the mainframe and ingestor card programs in agreement. It also reduces mainframe channel traffic when no signal is present. The SYNC command response time varies, depending on the presence of a signal. These response times are explained below.

Response Time	Reason
Immediate	There are 10 or more unreported blocks in the buffers. The GVAR hardware responds by clearing the Busy and Unit Exception status flags.
Short (5 seconds)	There are no unreported blocks, but unread data exists; either 10 block 11s are received or no new data is received. The GVAR hardware responds by clearing the Busy and Unit Exception status flags.
Long (29 seconds)	There are no unreported blocks or unread data in the buffers; either new data arrives in these 29 seconds or a Unit Exception (abort chain) occurs meaning there is no data. The GVAR hardware responds by clearing the Busy flag and either clearing the Unit Exception flag (new data) or setting the Unit Exception flag (no new data).

The GPCI monitors the GVAR Ingestor's status flags. Upon detecting a cleared Busy flag, the GPCI uses the status of the Unit Exception flag to indicate command success or failure. If the Unit Exception flag is set, a command fail is transmitted. If the Unit Exception flag is cleared, a successful completion is transmitted.

WRITE Command

The WRITE command (01) may consist of 2 to 20 bytes. The first byte identifies the command as the WRITE command and remaining bytes as WRITE command parameters. The WRITE command has three chain formats. They are CLEAR, NEWDOC and DATA. In each command, the four high order bits of the second byte signify the command code (chain format). The hexadecimal command codes and their meanings are listed below.

Command Code	Meaning
10	Clears all buffers
00	NEWDOC with validity code
80	NEWDOC without validity code
40	DATA request
C0	DATA without validity code

The CLEAR, NEWDOC and DATA chains are outlined below. The command, its parameters, response and field definitions are listed. The NEWDOC and DATA chains include diagrams depicting the response format.

CLEAR Chain

Command: WRITE, 10

Response: None. The CLEAR chain clears the hardware buffers. It is optional and has no corresponding READ command.

NEWDOC Chain

Command: WRITE, 00/80, VC (4 bytes)

Response: VC (4 bytes; optional), GenStat (2 bytes), [BlkInf*1* + Block Hdr] (40 bytes), . . . [BlkInf *m* + Block Hdr] (40 bytes)

Response Fields:

VC is the Validity Code.

GenStat is the frame sync and auxiliary status.

BlkInf tells you which bytes contain information about a particular block to guide a data request; *m* has a maximum of 10.

Block Hdr is the Block Header for this block.

The response format for the WRITE-NEWDOC chain is further shown in Figure 3 below.

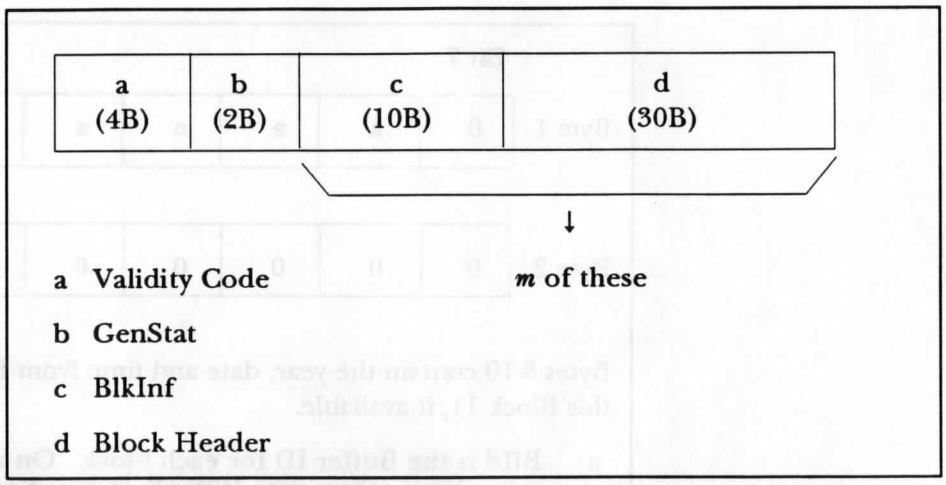


Figure 3. Response Format for the WRITE-NEWDOC Chain

The GenStat field in the NEWDOC chain is shown in Figure 4 below. This figure is an enlargement of box "b" in Figure 3 above.

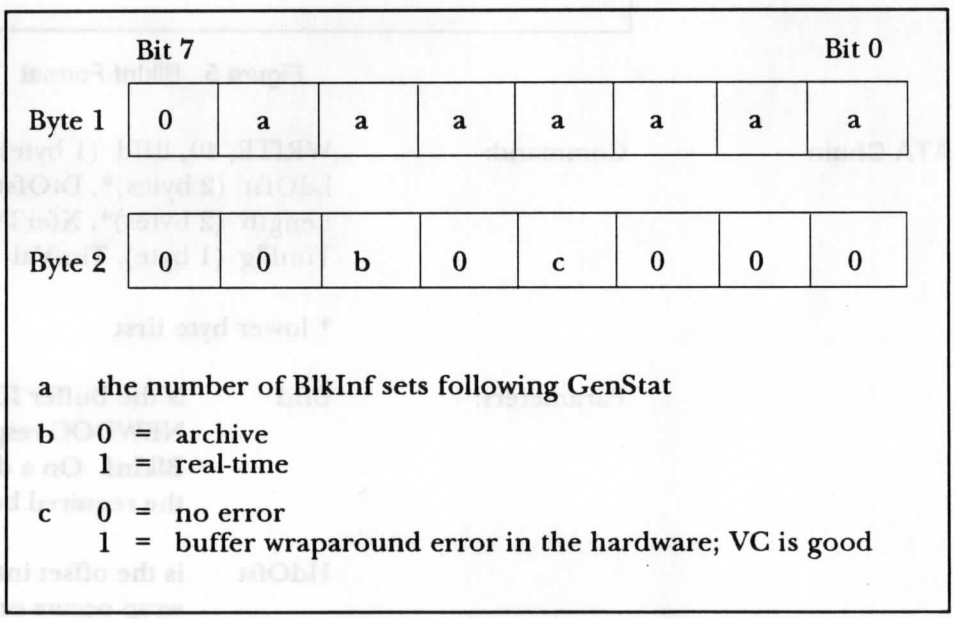
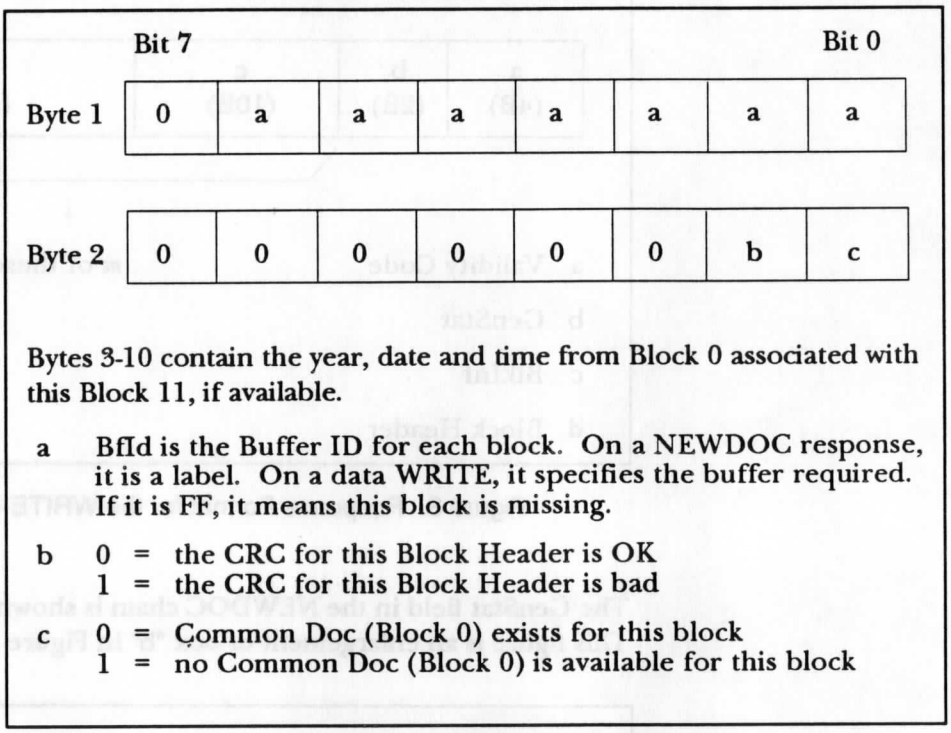


Figure 4. GenStat Format

The BlkInf field in the NEWDOC chain is shown in Figure 5 below. This figure is an enlargement of box "c" in Figure 3 on the previous page.



Bytes 3-10 contain the year, date and time from Block 0 associated with this Block 11, if available.

a BfId is the Buffer ID for each block. On a NEWDOC response, it is a label. On a data WRITE, it specifies the buffer required. If it is FF, it means this block is missing.

b 0 = the CRC for this Block Header is OK
1 = the CRC for this Block Header is bad

c 0 = Common Doc (Block 0) exists for this block
1 = no Common Doc (Block 0) is available for this block

Figure 5. BlkInf Format

DATA Chain

Command: WRITE, 40, BfId (1 byte), HdOfst (2 bytes)*, LdOfst (2 bytes)*, DtOfst (2 bytes)*, Length (2 bytes)*, XferTyp (1 byte), TimFlg (1 byte), TimVal (8 bytes)

* lower byte first

Parameters:

BfId is the Buffer ID of the block. On a NEWDOC response, it is a label in BlkInf. On a data WRITE, it specifies the required buffer.

HdOfst is the offset into the header if a buffer wrap occurs on the mainframe.

LdOfst is the offset to the start-of-line documentation in a block.

DtOfst is the data field offset.

Length is the number of bytes to transfer.

XferTyp — is the bits per word and how to transfer them.
- 0= zero-masked 6-bit data
- 1= shifted 6-bit data
- 2= unshifted 6- and 8-bit data
- 3= shifted 10-bit data
- 4= shifted 12-bit data
- 5= 16-bit data

TimFlg — is the time flag set when Block 11s have no Block 0 time associated with them. If the Doc Available flag in BlkInf equals zero, the time sent back on a Block 11 data request is taken from TimVal.

TimVal — is the time generated by the ingestor handler software.

Response: VC (4 bytes; optional), Data (*n*)

Response Fields:
- VC — is the Validity Code.
- Data (*n*) — is 8040 bytes of Bock 11 data.

Figure 6 below shows the response format for the WRITE-DATA chain.

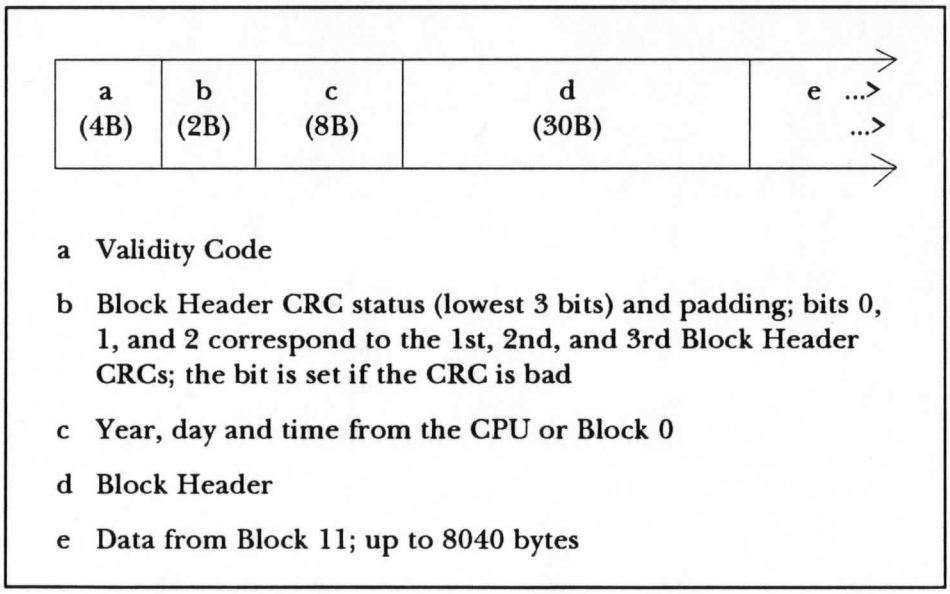


Figure 6. Response Format for the WRITE-DATA Chain

READ Command

The READ command (02) transfers the results of the WRITE commands. It carries the data from the hardware buffer to the mainframe channel.

GVAR Imager Ingestor Protocol

Protocol is the communications command/response link between intelligent hardware and software subsystems of a satellite data ingestor system. This chapter describes the protocol for communication between the mainframe ingestor handler and the GVAR Imager ingestor hardware.

The GVAR Imager ingestor card acts as a number of buffers, each of which contains an entire GVAR data block. Eleven data blocks make up a GVAR scan. The ingestor hardware informs the mainframe of these blocks. The mainframe may or may not request data from a particular data block.

The GVAR Imager card contains a 2M byte memory area that is treated like a circular queue. A block's worth of data makes up an arbitrary buffer. The controlling firmware contains pointers to the start of these buffers. The firmware allocates space for up to 128 buffer pointers. The length of the buffer is in the block's header information. Blocks that cross the 2M boundary continue at an address of zero. All data is stored as 8-bit bytes with 10-bit data stored in 2 bytes.

The Imager protocol consists of a series of command chains executed by three commands: SYNC, WRITE and READ.

Control Fields

The following list defines the GVAR control fields which are presented by the satellite or the ingestor hardware for use by the ingestor software. Before it can proceed to ingest or ignore an image, the ingestor handler needs to know the quality and type of signal present. GVAR transmits these control fields in the common documentation section of the signal.

Absolute Scan Number tells the ingestor software where it is in the image and directs where to write the data in the image area. It is used in matching the ingestor requests and deciding whether or not to ask for data at this point.

Buffer IDs indicate where in the ingestor hardware's memory the current data resides. It is used to construct a channel command that asks for data.

Data Valid Flag tells whether the data on this scan is valid or not and is a signal quality check.

Easternmost Visible Pixel is the eastern boundary of the current image. It determines what percentage of the image being sent overlaps the image area(s) in the ingestor requests and whether the image is, therefore, desired or not.

Frame Count is a value that increments for each image. It checks signal quality and consistency.

Frame End Flag tells the hardware ingestor that this is the end of the frame and/or image being sent.

Mode tells you the type and/or size of the ingested image. There are four mode choices: normal, watch, warning and priority.

Northernmost Image Line is the northern boundary of the current image. It determines what percentage of the image being sent overlaps the image area(s) in the ingestor requests and whether the image is, therefore, desired or not.

Priority Frame is a flag that is set if a priority frame is being sent. It is used in matching requests and determining which areas to write data to.

Priority 1 is a flag that is set if the frame is the outer frame of nested priorities. Currently, priority frames are not nested.

Priority 2 is a flag that is set if the frame being received is the nested priority frame. Currently, priority frames are not nested.

Satellite Day is used to match an image request. A satellite day change indicates that the data just received is from a new image.

Satellite ID is used to match a request window. It does not change throughout an image.

Satellite Time is used to match an image request. A satellite time change greater than the expected per-scan increment indicates that the data just received may be from a new image.

Scan Direction indicates whether or not a legal image is being transmitted. When this value indicates North-to-South, a legal image is being transmitted. When an image is completed, the satellite scans one line or scans South-to-North and at this point is not sending valid, navigable image data.

Scan Number is the number that tells the ingestor software where it is in the image. It directs where to write the data in the image area.

Southernmost Image Line is the southern boundary of the current image. It determines what percentage of the image being sent overlaps the image area(s) in the ingestor requests and whether the image is, therefore, desired or not.

Validity Code is a number created by the ingestor software once for each image. The hardware sends the validity code back at the front of each image line if the data is good. If a line is missed or garbled, the validity code is returned from the hardware as a zero.

Westernmost Visible Pixel is the western boundary of the current image. It determines what percentage of the image being sent overlaps the image area(s) in the ingestor requests and whether the image is, therefore, desired or not.

Command Chains

A command chain is a series of desired actions that a host processor sends to the channel processor for execution. Figure 7 below shows a sample command chain for the GVAR Imager ingestor. To keep the channel busy and prevent data loss, the host processor always sends two channel command chains to the channel. When a chain is completed, the NEWDOC WRITE-READ command pair provides the mainframe with the information it needs to create and send the next chain. The commands that acquire the data described in NEWDOC (also a WRITE-READ pair) are sent after the two chains already queued and acquiring data described in the previous NEWDOC are completed. Thus, it is necessary to buffer at least four scans:

- One scan is waiting.
- A second scan is read by the mainframe.
- A third scan is loaded from the signal.
- The fourth scan is held in reserve in case the CPU slows down and the ingestor handler needs to catch up with the hardware.

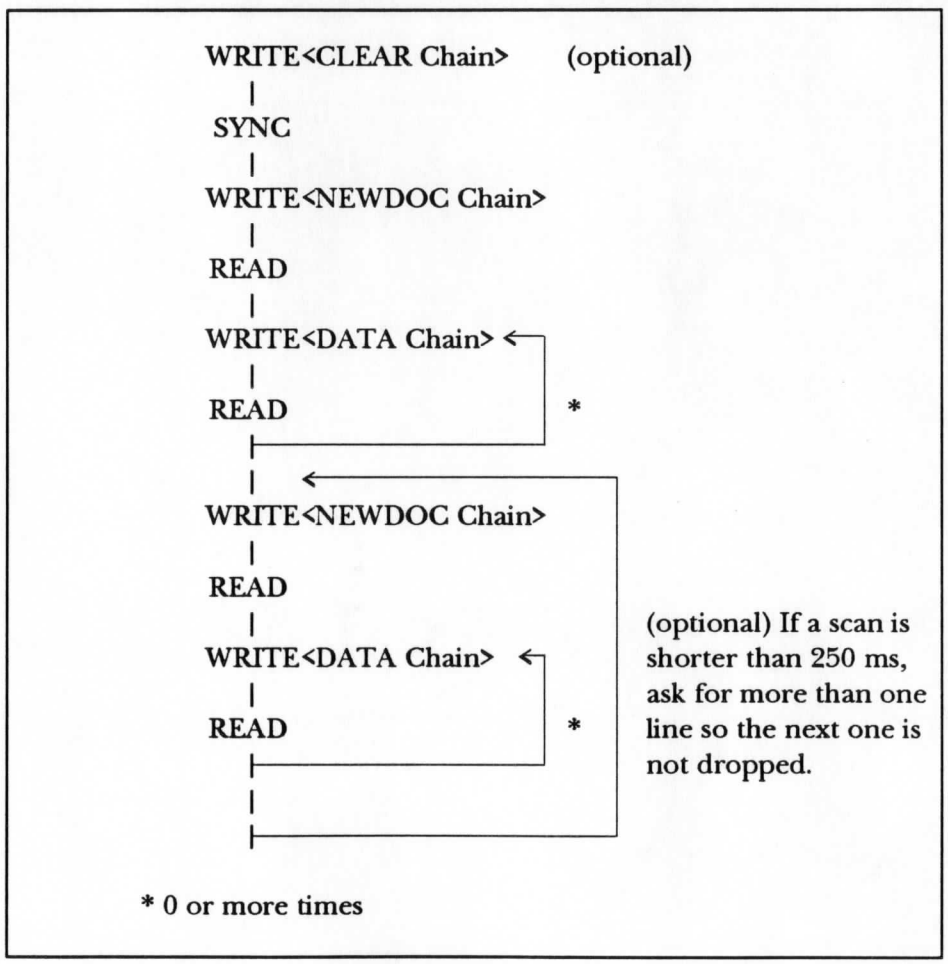


Figure 7. Sample Imager Command Chain

Command Chain Codes

The General Purpose Channel Interface (GPCI) acts as an interface between the channel and the ingestor hardware. Command chain codes are not transmitted from the channel. The GPCI recognizes the various command chains and generates the command chain codes. These codes are transmitted to the ingestor hardware as the command type. Throughout the remainder of this chapter, the command type is considered the **first** byte of the command. From an ingestor software or channel perspective, the second byte of a command is the beginning of the command.

SYNC Command

The SYNC command (07) controls the timing. It provides the synchronization that keeps the mainframe and ingestor card programs in agreement. It also reduces mainframe channel traffic when no signal is present. The SYNC command response time varies, depending on the presence of a signal. These response times are explained below.

Response Time	Reason
Immediate	There are unreported scans in the buffers. The GVAR hardware clears the Busy and Unit Exception status flags.
Short (5 seconds)	There are no unreported scans, but unread data exists; either the new scan is completed or the scan is completed with old data. The BlkInf sets (the bytes containing block information to guide a data request) that identify old data in the next WRITE-NEWDOC response are set to zero. The GVAR hardware clears the Busy and Unit Exception status flags.
Long (29 seconds)	There are no unreported blocks or unread data in the buffers; either new data arrives in these 29 seconds or a Unit Exception (abort chain) occurs meaning there is no data. The GVAR hardware responds by clearing the Busy flag and either clearing the Unit Exception flag (new data) or setting the Unit Exception flag (no new data).

The GPCI monitors the GVAR Ingestor's status flags. Upon detecting a cleared Busy flag, the GPCI uses the status of the Unit Exception flag to indicate command success or failure. If the Unit Exception flag is set, a command fail is transmitted. If the Unit Exception flag is cleared, a successful completion is transmitted.

WRITE Command

The WRITE command (01) may consist of 2 to 21 bytes. The first byte identifies the command as the WRITE command and remaining bytes as WRITE command parameters. The WRITE command has three chain formats. They are CLEAR, NEWDOC and DATA. In each command, the four high order bits of the second byte signify the command code (chain format). The hexadecimal codes and their meanings are listed below.

Command Code	Meaning
10	Clears all buffers
00	NEWDOC with validity code
80	NEWDOC without validity code
4*y*	DATA request
C*y*	DATA without validity code

For WRITE-DATA commands, the four low order bits of the second byte signify the sample/average data process instruction and are represented by the *y* in the command code. The hexadecimal values for the process instructions and their meanings are listed below.

Process Instruction	Meaning
0	Full resolution
1	Undefined
2	Sample by 2
3	Sample by 3
4	Sample by 4
5	Sample by 6
6	Sample by 8
7	Sample by 12
8	Sample by 16
9	Average by 2
A	Average by 3
B	Average by 4
C	Average by 6
D	Average by 8
E	Average by 12
F	Average by 16

The CLEAR, NEWDOC and DATA chains are outlined on the next few pages. The command, its parameters, response and field definitions are listed. The NEWDOC and DATA chains include diagrams depicting the response format.

CLEAR Chain

Command: WRITE, 10

Response: None. The CLEAR chain clears the hardware buffers. It is optional and has no corresponding READ command.

NEWDOC Chain

Command: WRITE, 00/80, VC (4 bytes)

Response: VC (4 bytes; optional), GenStat (10 bytes), BlkInf (2 bytes; 11 of these), Doc

Response Fields:

VC — is the Validity Code.

GenStat — is the frame sync and auxiliary status.

BlkInf — contains the buffer address for each of the blocks of data.

Doc — is the oldest unreported Block 0.

The response format for the WRITE-NEWDOC chain is detailed in Figure 8 below.

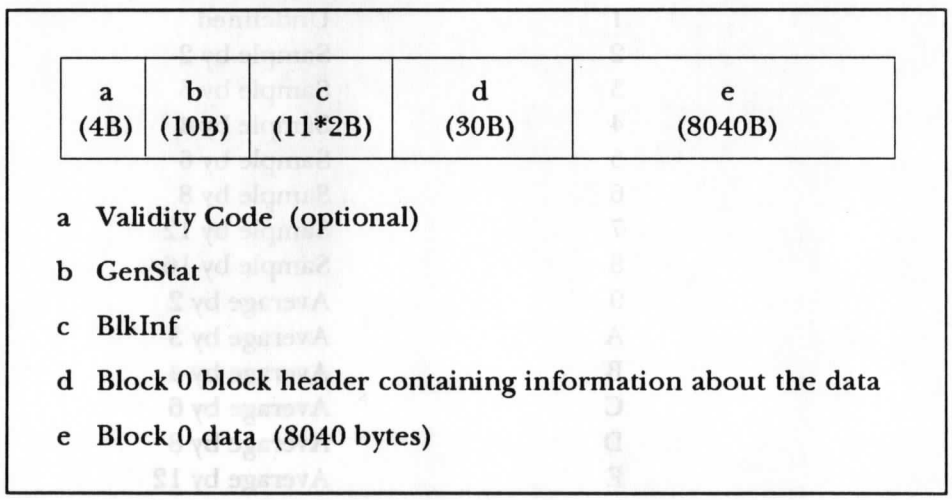
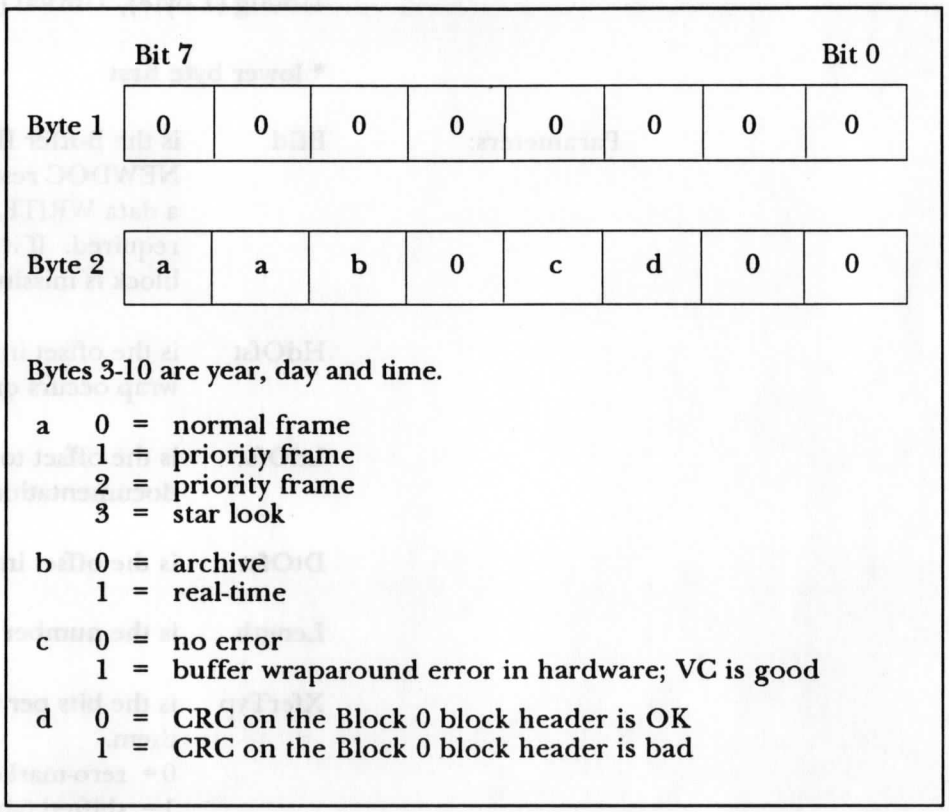


Figure 8. Response Format for the WRITE-NEWDOC Chain

The GenStat field in the NEWDOC chain is shown in Figure 9 below. This figure is an expansion of box "b" in Figure 8 on the previous page.



Bytes 3-10 are year, day and time.

a 0 = normal frame
1 = priority frame
2 = priority frame
3 = star look

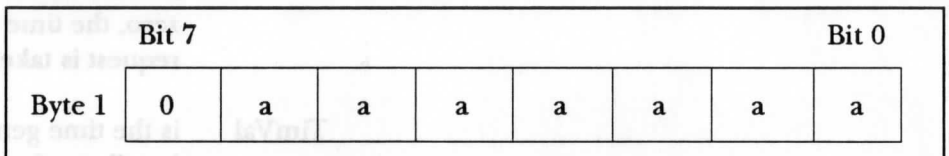
b 0 = archive
1 = real-time

c 0 = no error
1 = buffer wraparound error in hardware; VC is good

d 0 = CRC on the Block 0 block header is OK
1 = CRC on the Block 0 block header is bad

Figure 9. GenStat Format

The format for the BlkInf field in the NEWDOC chain is shown in Figure 10 below. This figure is an expansion of box "c" in Figure 8 on the previous page.



Byte 2 is unused.

a BfId is the Buffer ID for each block. On a NEWDOC response, it is a label. On a data WRITE, it specifies the buffer required. If it is FF, it means this block is missing.

Figure 10. BlkInf Format

DATA Chain

Command: WRITE, 4*y*, BfId (1 byte), HdOfst (2 bytes)*, LdOfst (2 bytes)*, DtOfst (2 bytes)*, Length (2 bytes)*, XferTyp (1 byte), TimFlg (1 byte), TimVal (8 bytes)

* lower byte first

Parameters:

BfId — is the Buffer ID for each block. On a NEWDOC response it is a label. On a data WRITE, it specifies the buffer required. If it is FF, it means the block is missing.

HdOfst — is the offset into the header if a buffer wrap occurs on the mainframe.

LdOfst — is the offset to the start-of-line documentation in a block.

DtOfst — is the offset into the data field.

Length — is the number of bytes to transfer.

XferTyp — is the bits per word and how to transfer them.
0= zero-masked 6-bit data
1= shifted 6-bit data
2= unshifted 6- and 8-bit data
3= shifted 10-bit data
4= shifted 12-bit data
5= 16-bit data

TimFlg — is the time flag set when a block has no Block 0 time associated with it. If the Doc Available flag in BlkInf equals zero, the time sent back on a data request is taken from TimVal.

TimVal — is the time generated by the ingestor handler software.

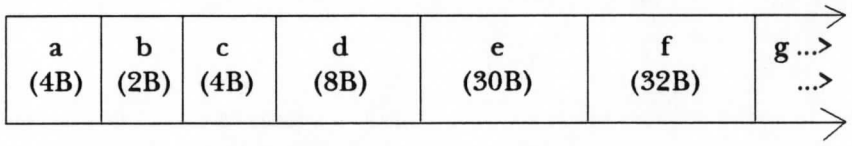
Response: VC (4 bytes; optional), Data (*n*)

Response Fields:

VC — is the Validity Code; it is zero if the requested block is missing.

Data (*n*) — is the header and data or doc-as-data; *n* depends on the documentation size and the amount of data requested.

Figure 11 below shows the response format for the WRITE-DATA chain. Figure 12 shows the response format for a Doc-As-Data WRITE request.



a Validity Code (optional)

b Block Header CRC Status (lowest 3 bits) and padding; bits 0, 1 and 2 correspond to the 1st, 2nd and 3rd Block Header CRCs; the bit is set if the CRC is bad

c Scan Status; the bits describing scan direction, etc. from Block 0

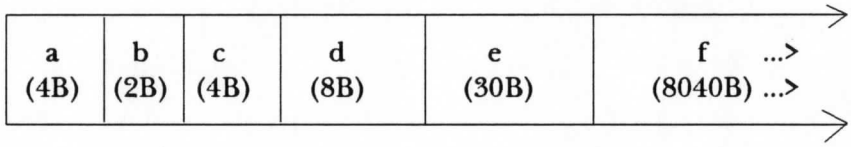
d Year, day and time from Block 0

e Block Header; contains various information about the data

f Line Documentation; first part of the data; uniquely identifies the data; part of the header when HdOfst is used

g The rest of the data; up to 41920 bytes

Figure 11. Response Format for the WRITE-DATA Chain



a Validity Code (optional)

b Block Header CRC Status (lowest 3 bits) and padding; bits 0, 1 and 2 correspond to the 1st, 2nd and 3rd Block Header CRCs; the bit is set if the CRC is bad

c Scan Status; the bits describing scan direction, etc. from Block 0

d Year, day and time from Block 0

e Block Header; contains various information about the block

f Data from Block 0; 8040 bytes

Figure 12. Doc-As-Data WRITE Request Format

READ Command

The READ command (02) transfers the results of the WRITE commands. It carries the data from the hardware buffer to the mainframe channel.

GMS Ingestor Protocol

Protocol is the communications command/response link between intelligent hardware and software subsystems of a satellite data ingestor system. This chapter describes the protocol for communication between the mainframe ingestor handler and the GMS ingestor hardware.

The GMS ingestor card uses the new generation ingestor hardware format. This ingestor card listens to the GMS signal while waiting for commands from the mainframe ingestor handler. It synchronizes the mainframe to the ingestor signal.

The GMS protocol consists of a series of command chains executed by these three commands: SYNC, WRITE, and READ

Control Fields

The following list defines the GMS control fields which are presented by the satellite or the ingestor hardware for use by the ingestor software. Before it can proceed to ingest or ignore an image, the ingestor handler needs to know the quality and type of signal present. GMS transmits these control fields in the common documentation section of the signal.

Beta Count is the value used for navigating images.

Calibration Table is the number used to determine in which table the calibration for the current image resides. The calibration field changes infrequently and can be used by the ingestor handler to check signal quality.

Data Source determines whether the signal is from a satellite or is a test pattern. The ingestor handler is programmed not to ingest a test pattern.

Earth Radius is a constant used by the ingestor handler to check signal quality.

Frame Flag is set to a high value to tell the ingestor that the satellite has begun scanning for an image.

Picture Flag is set to a high value to tell the ingestor that legal transmission has begun. The ingestor handler can begin to ingest data as soon as there are matching requests.

Picture Reset gives the scan number where the valid image ends.

Picture Set gives the scan number where the valid image begins.

Satellite Day is used to match an image request. A satellite day change indicates that the data just received is from a new image.

Satellite ID is used to match a request window. It does not change throughout an image.

Satellite Time is used to match an image request. A satellite time change greater than the expected per-scan increment indicates that the data just received may be from a new image.

Scan Direction indicates whether or not a legal image is being transmitted. When this value indicates North-to-South, a legal image is being transmitted. When an image is completed, the satellite scans one line or scans South-to-North and at this point is not sending valid, navigable image data.

Scan Number is the number that tells the ingestor software where it is in the image. It directs where in the image area to write the data.

Signal Type designates the signal as one of two GMS signal types: A (full globe) and B (northern hemisphere). This value is needed to match a request and to determine whether the same, continuing image or a new image is being transmitted.

Subcommutation ID gives the two counter values (from 0-7 and from 0-24) that the ingestor handler examines to determine which pieces of Common Documentation to gather for the needed navigation/O&A information. Common Documentation is sent from the satellite in 25 pieces: eight repetitions of the first piece are sent (one per scan), then eight repetitions of the second piece, and so on.

Validity Code is a number created by the ingestor software once for each image. The hardware sends the validity code back at the front of each image line when the data is good. If a line is missed or garbled, the validity code is returned from the hardware as a zero.

Command Chains

To keep the channel busy and prevent data loss, the host processor always sends two channel command chains to the channel. Figure 13 below shows a sample command chain for the GMS ingestor. When a chain is completed, the NEWDOC WRITE-READ command pair provides the mainframe with the information it needs to create and send the next chain. The commands that acquire the data described in NEWDOC, also a WRITE-READ pair, are sent after the two chains already queued and acquiring data described in the previous NEWDOC are completed.

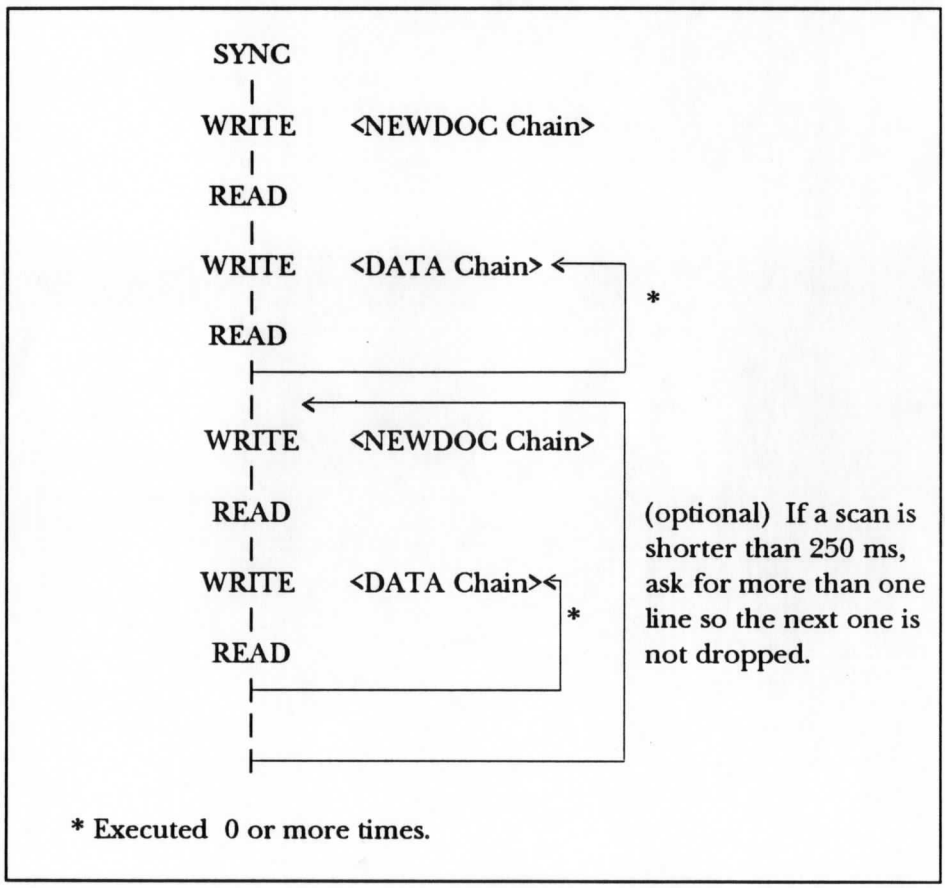


Figure 13. Sample GMS Command Chain

Command Chain Codes

The General Purpose Channel Interface (GPCI) acts as an interface between the channel and the ingestor hardware. Command chain codes are not transmitted from the channel. The GPCI recognizes the various command chains and generates the command chain codes. These codes are transmitted to the ingestor hardware as the command type. Throughout the remainder of this chapter, the command type is considered the **first** byte of the command. From an ingestor software or channel perspective, the second byte of a command is the beginning of the command.

SYNC Command

The SYNC command (07) controls the timing. It provides the synchronization that keeps the mainframe and ingestor card programs in agreement. It also reduces the mainframe channel traffic when no signal is present. The SYNC command response time varies, depending on the presence of a signal. These response times are explained below.

Response Time	Reason
Immediate	There are unreported scans in the buffers. The GMS hardware clears the Busy and Unit Exception status flags.
Short (5 seconds)	There are no unreported scans, but unread data exists; either the new scan is completed or the scan is completed with old data. The BlkInf sets (bytes containing block information to guide a data request) that identify old data in the next WRITE NEWDOC response are set to zero. The GMS hardware clears the Busy and Unit Exception status flags.
Long (29 seconds)	There are no unreported blocks or unread data in the buffers; either new data arrives before the 29-second time window has elapsed or a Unit Exception (abort chain) occurs, meaning there is no data. The GMS hardware responds by clearing the Busy flag and either clearing the Unit Exception flag (if there is new data) or setting the Unit Exception flag (if there is no new data).

The GPCI monitors the GMS Ingestor's status flags. Upon detecting a cleared Busy flag, the GPCI uses the status of the Unit Exception flag to indicate command success or failure. If the Unit Exception flag is set, a command fail is transmitted. If the Unit Exception flag is cleared, a successful completion is transmitted.

WRITE Command

The WRITE command (01) may consist of 2 to 21 bytes. The first byte is generated by the GPCI. It identifies the command as a WRITE command. The remaining bytes are interpreted as WRITE command parameters. The WRITE command has two chain formats, NEWDOC and DATA. In each command, the four high order bits of the second byte always signify the command code (chain format). The hexadecimal code values and their meanings are listed below.

Command Chain	Code	Meaning
NEWDOC Chain	00	NEWDOC with validity code
NEWDOC Chain	80	NEWDOC without validity code
DATA Chain	4*y*	DATA request with validity code
DATA Chain	C*y*	DATA without validity code

For WRITE DATA commands, the four low order bits of the second byte signify the sample/average data process instruction and are represented by the *y* in the command code. The hexadecimal value for the process instructions and their meanings are listed below.

Process Instruction	Meaning
0	Full Resolution
1	Undefined
2	Sample by 2
3	Sample by 3
4	Sample by 4
5	Sample by 6
6	Sample by 8
7	Sample by 12
9	Average by 2
A	Average by 3
B	Average by 4
C	Average by 6
D	Average by 8
E	Average by 12

The NEWDOC and DATA chains are outlined beginning on the next page. The command, its parameters, response and field definitions are listed. Diagrams depicting response formats are shown for GenStat and BlkInf.

NEWDOC Chain

Command: WRITE, 00/80, VC (4 bytes)

Response: VC (4 bytes; optional), GenStat (10 bytes), BlkInf (2 bytes; 8 of these), Doc

Response Fields:

VC	is the Validity Code; it is zero if the requested block is missing.
GenStat	is the frame sync and auxiliary status.
BlkInf	contains the buffer address for each of the blocks of data.
Doc	is the common documentation.

The response format for the WRITE-NEWDOC chain is detailed in Figure 14 below.

a (4B)	b (4B)	c (8*2B)	d (128B)	e (832B)

a Validity Code

b GenStat

c BlkInf

d Block Header containing information about the data

e Block 0 data (0 - 832 bytes)

Figure 14. Response Format for the WRITE-NEWDOC Chain

The GenStat field in the NEWDOC chain is shown in Figure 15 below. This figure shows an expanded view of box "b" in Figure 14 on the previous page.

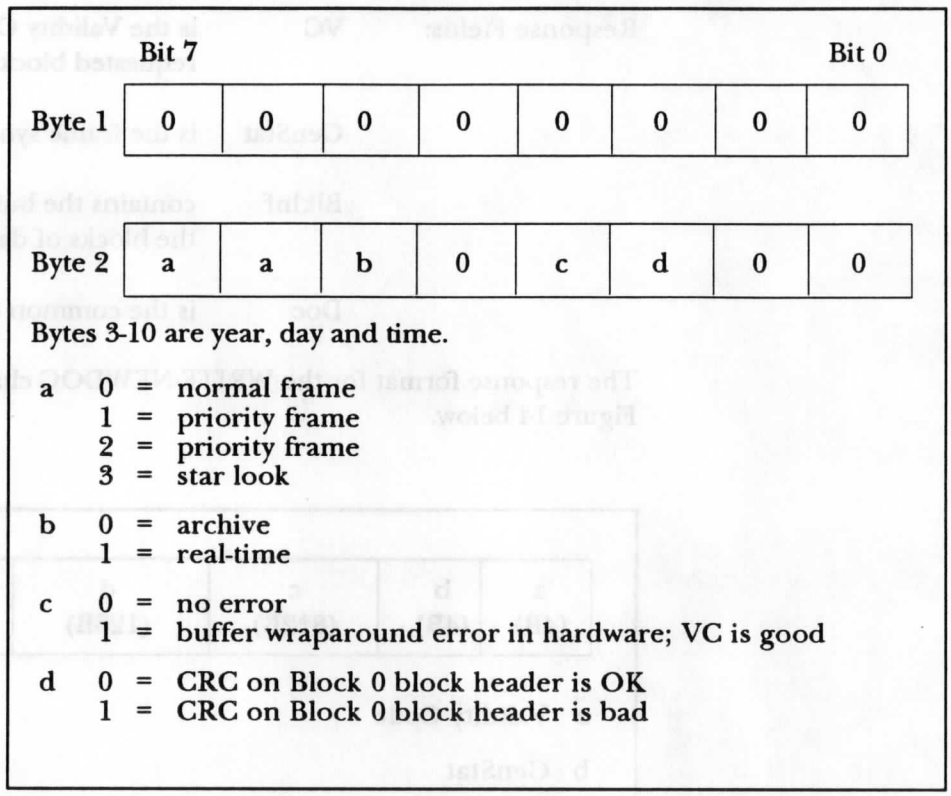


Figure 15. GenStat Format

The format for the BlkInf field in the NEWDOC chain is shown in Figure 16 below. This figure shows an expanded view of box "c" in Figure 14 on the previous page.

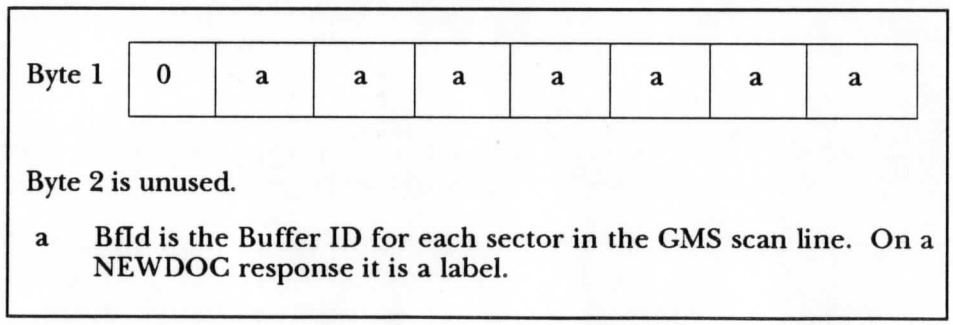


Figure 16. BlkInf Format

DATA Chain

Command: WRITE, 4*y*, BfId (1 byte), HdOfst (2 bytes)*, LdOfst (2 bytes)*, DtOfst (2 bytes)*, Length (2 bytes)*, XferTyp (1 byte), TimFlg (1 byte), TimVal (8 bytes)

* lower byte first

Parameters:

BfId	is the Buffer ID for each sector in a scan line. On a WRITE-DATA chain, it specifies the buffer required. If it is FF, it means the block is missing.
HdOfst	is the offset into the header if a buffer wrap occurs on the mainframe.
LdOfst	is always zero for GMS.
DtOfst	is the offset into the data field.
Length	is the number of bytes to transfer.
XferTyp	is the bits per word and transfer type. 0 = zero-masked 6-bit data 1 = shifted 6-bit data 2 = unshifted 6-bit and 8-bit data 3 = shifted 10-bit data 4 = shifted 12-bit data 5 = 16-bit data
TimFlg	is always 0 for GMS.
TimVal	is always zero for GMS.

Response: VC (4 bytes; optional), Data (*n*)

Response Fields:

VC	is the Validity Code; it is zero if the requested block is missing.
Data (*n*)	is the header and data, or header and doc-as-data. *n* depends on the signal's documentation size and the amount of data requested.

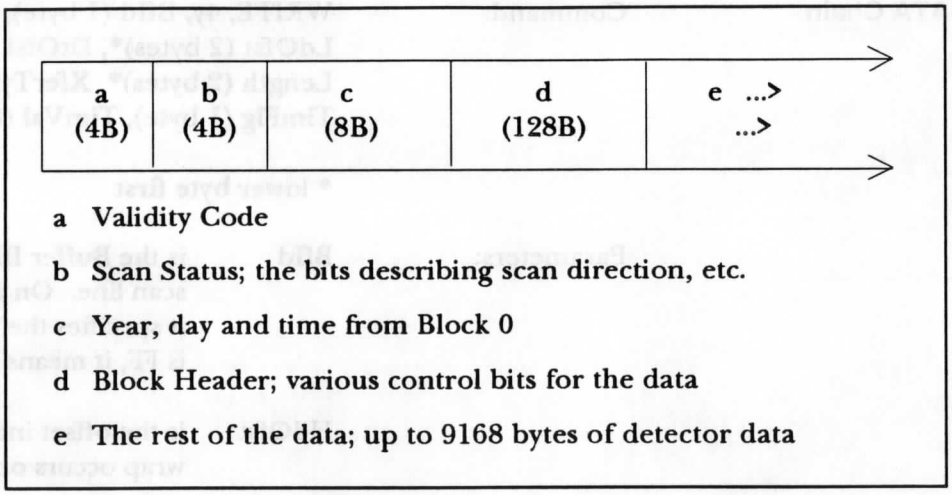


Figure 17. Response Format for the WRITE-DATA Chain

Figure 17 above shows the response format for the WRITE-DATA chain. Figure 18 shows the response format for a Doc-as-Data WRITE request.

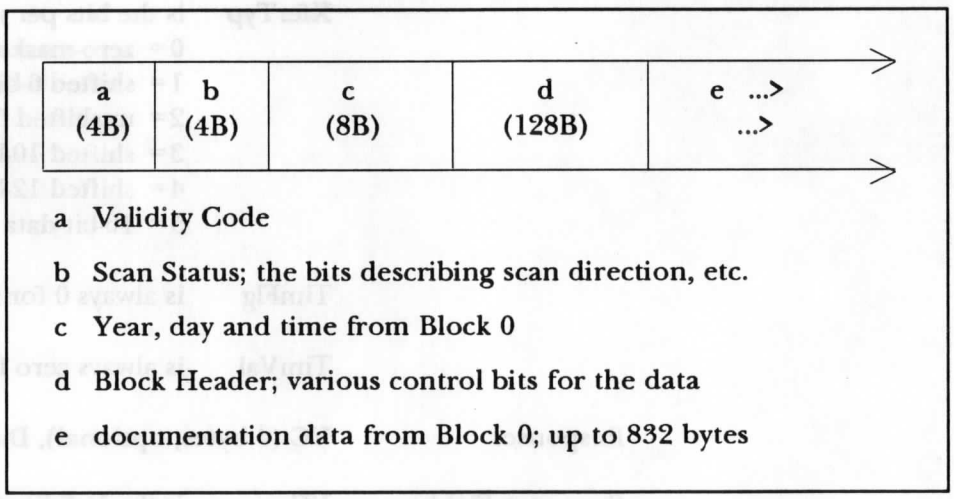


Figure 18. Doc-as-Data WRITE Request Format

READ Command

The READ command (02) transfers the results of the WRITE commands. It carries the data from the hardware buffer to the mainframe channel.

AAA Block 1 Ingestor Protocol

This chapter describes the protocol for communication between the mainframe ingestor handler and the AAA Block 1 (AUXBLK) ingestor hardware. The Block 1 ingestor card uses the new generation ingestor hardware format. The ingestor card acts as a number of buffers, each of which contains an entire Block 1 sector. The ingestor hardware informs the mainframe of these blocks, one at a time. The mainframe may or may not request data from a particular block. The card is informed when a block is no longer needed and may be cleared.

The Block 1 protocol consists of a series of command chains executed by three commands: SYNC, WRITE and READ. The NOP command is also discussed at the end of the chapter.

Control Fields

The following list defines the AAA Block 1 control fields which are presented by the satellite or the ingestor hardware for use by the ingestor software or the user. Before it can proceed to ingest or ignore an image, the ingestor handler needs to know the quality and type of signal present. AAA Block 1 transmits these control fields in the common documentation section of the signal.

ASCII Flag tells you the type of data sent in the AUXBLK transmission.

Buffer ID indicates where in the ingestor hardware's memory the current data resides. It is used to construct a channel command that asks for data.

CRC Error Flag tells you if the hardware detects a CRC (Cyclic Redundancy Checker) error. This is a signal quality check.

Data Valid Flag tells you if the data on this scan is valid; it is a signal quality check used primarily by applications programs.

Number of Blocks is how many blocks of data the hardware has waiting. It should equal 1.

Number of Data Words is how many words of data are in the hardware buffers.

Reset Flag is set at the start of an image. If the Reset Flag is not set, data is from the middle of an image.

Satellite ID is a transmission from GOES-7. It is not part of the signal but can be used to match a request window. It does not change throughout an image.

Scan Number is the number that tells the ingestor software where it is in the image. It is compared to request parameters to see if a particular line is needed. It directs where in the image area to write the data. The scan number increments by 1 or, if the Reset flag is high, is set to 1.

Signal Type is used for matching ingestor requests.

Word Size is used by the ingestor handler to determine how many bytes of data to request.

Wrap Error Flag is a signal and/or line quality check for the ingestor.

Command Chains

To keep the channel busy and ensure no loss of data, the host processor always sends two channel command chains to the channel. Figure 19 below shows a sample command chain for the AAA Block 1 ingestor. When a chain is completed, the NEWDOC WRITE-READ command pair provides the mainframe with the information it needs to create and send the next chain. The commands that acquire the data described in NEWDOC (also a WRITE-READ pair) are sent after the two chains already queued and acquiring data described in the previous NEWDOC are completed. Thus, it is necessary to buffer at least four blocks:

- One block is waiting.
- A second block is read by the mainframe.
- A third block is loaded from the signal.
- The fourth block is held in reserve in case the signal is noisy and the ingestor handler needs to decide what to do with an unexpected block, or when the CPU slows down and the ingestor handler needs to catch up with the hardware.

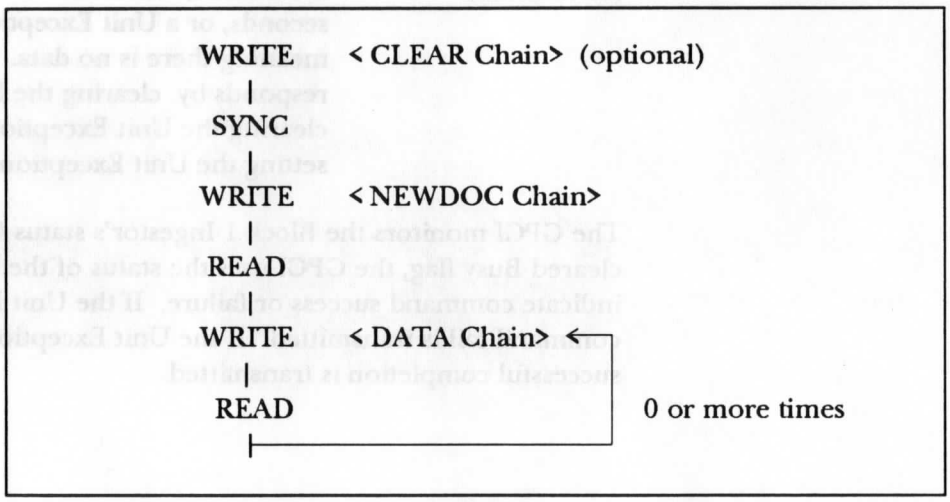


Figure 19. Sample AAA Block 1 Command Chain

Command Chain Codes

The General Purpose Channel Interface (GPCI) acts as an interface between the channel and the ingestor hardware. Throughout the remainder of this chapter, the command type is considered the **first** byte of the command. From an ingestor software or channel perspective, the second byte of a command is the beginning of the command.

SYNC Command

The SYNC command (07) controls the timing and provides the synchronization that keeps the mainframe and ingestor card programs in agreement. It also reduces mainframe channel traffic when no signal is present. The SYNC command response time varies, depending on the presence of a signal. These response times are explained below.

Response Time	Reason
Immediate	There are unreported blocks in the buffers. The Block 1 ingestor responds by clearing the Busy and Unit Exception status flags.
Short (5 seconds)	There are no unreported blocks, but unread data exists. The Block 1 hardware either returns the new data, or sends the old data and sets the count for the number of BlkInf sets to zero (no new data).
Long (29 seconds)	There are no unreported blocks or unread data in the buffers; either new data arrives in these 29 seconds, or a Unit Exception (abort chain) occurs, meaning there is no data. The Block 1 hardware responds by clearing the Busy flag and either clearing the Unit Exception flag (new data) or setting the Unit Exception flag (no new data).

The GPCI monitors the Block 1 Ingestor's status flags. Upon detecting a cleared Busy flag, the GPCI uses the status of the Unit Exception flag to indicate command success or failure. If the Unit Exception flag is set, a command fail is transmitted. If the Unit Exception flag is cleared, a successful completion is transmitted.

WRITE Command

The WRITE command (01) may consist of 2 to 40 bytes. The first byte identifies the command as the WRITE command and the remaining bytes as WRITE command parameters. The WRITE command has three chain formats. They are CLEAR, NEWDOC and DATA. In each command, the four high order bits of the second byte signify the command code (chain format). The hexadecimal command codes and their meanings are listed below.

Command Code	Meaning
10	Clears all buffers
00	NEWDOC with validity code
80	NEWDOC without validity code
40	DATA request

The CLEAR, NEWDOC and DATA chains are outlined below. The command, its parameters, response and field definitions are listed. The NEWDOC and DATA chains include diagrams depicting the response format.

CLEAR Chain

Command: WRITE, 10

Response: None. The CLEAR chain clears the hardware buffers. It is optional and has no corresponding READ command.

NEWDOC Chain

Command: WRITE, 00/80, VC (4 bytes)

Response: VC (4 bytes; optional), GenStat (4 bytes), BlkInf (2 bytes), Doc (*n*)

Response Fields:

- VC is the Validity Code.
- GenStat is the frame sync and auxiliary status.
- BlkInf contains the buffer address for each of the blocks of data.
- Doc (*n*) is the documentation; *n* depends on the block's data size.

The response format for the WRITE-NEWDOC chain is detailed in Figure 20 on the following page.

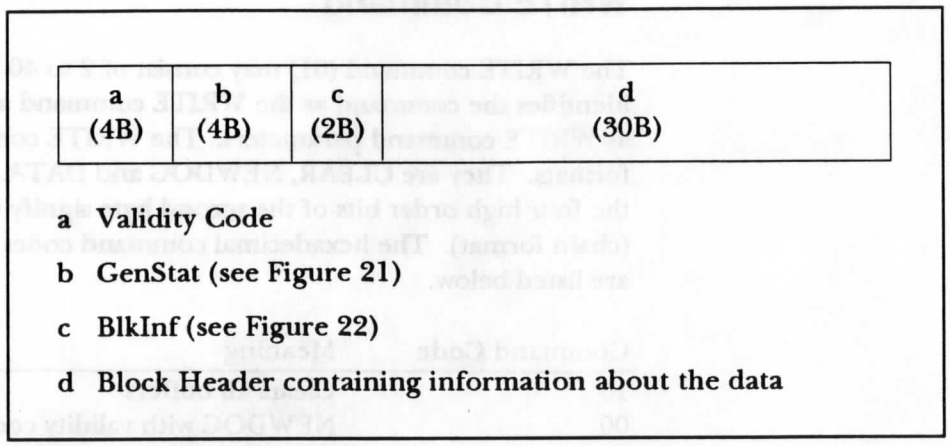


Figure 20. Response Format for the WRITE-NEWDOC Chain

The GenStat field in the NEWDOC chain is shown in Figure 21 below. This figure depicts the details of box "b" in Figure 20 above.

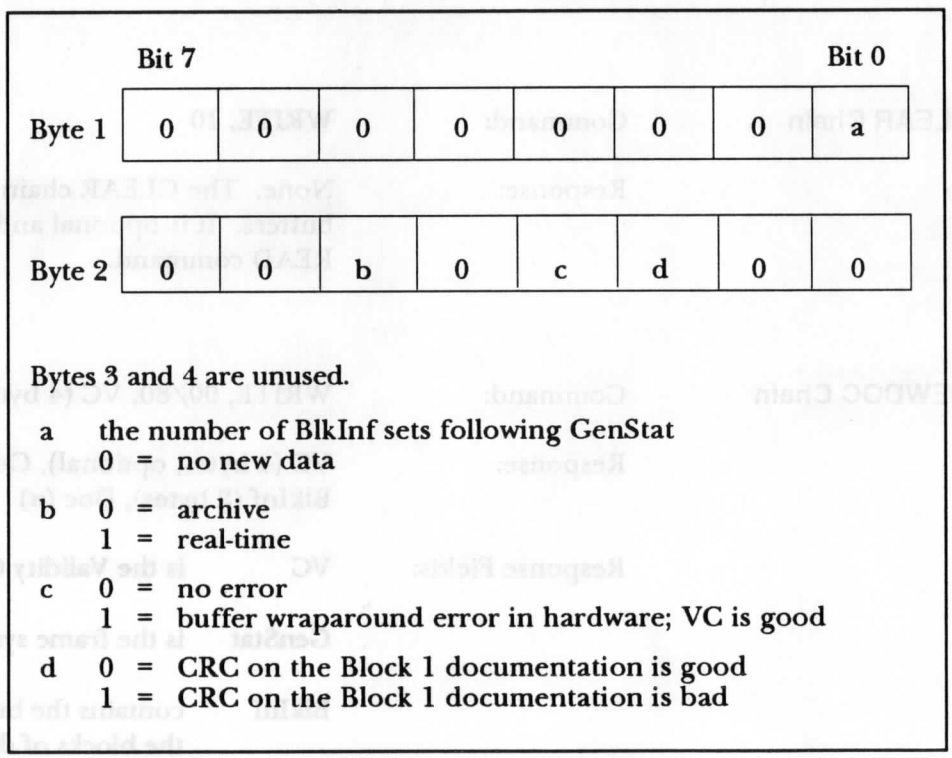


Figure 21. GenStat Format

The BlkInf field in the NEWDOC chain is shown in Figure 22 below. This figure depicts the details of box "c" in Figure 20 on the previous page.

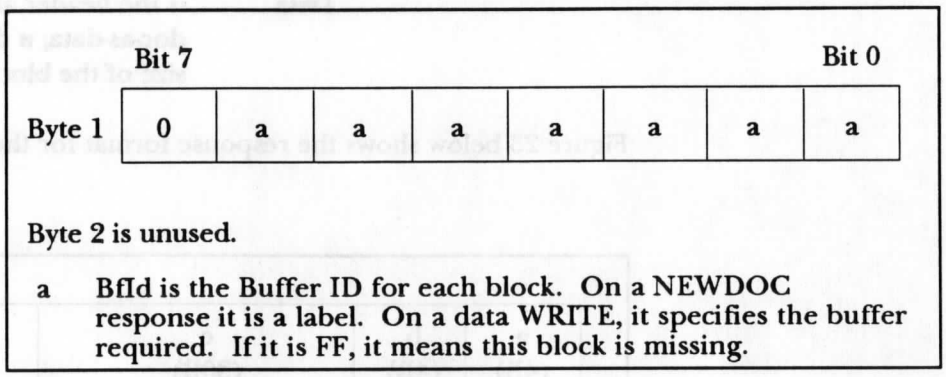


Figure 22. BlkInf Format

DATA Chain

Command: WRITE, 40, BfId (1 byte), HdOfst (2 bytes)*, DtOfst (2 bytes)*, Length (2 bytes)*, XferTyp (1 byte), TimVal (8 bytes)

* low order byte first

Parameters:

BfId is the Buffer ID. It specifies the buffer required for a WRITE-DATA chain. If it is FF, it means the block is missing.

HdOfst is the byte offset into the header if a buffer wrap occurs on the mainframe.

DtOfst is the byte offset into the data field.

Length is the number of bytes to transfer.

XferTyp is the bits per word and type of transfer.
0= zero-masked 6-bit data
1= shifted 6-bit data
2= unshifted 6-bit and 8-bit data
3= shifted 10-bit data
4= shifted 12-bit data
5= 16-bit data

TimVal is the CPU generated day and time value accurate to the millisecond.

Response:	VC (4 bytes; optional), Data (*n*)	
Response Fields:	VC	is the Validity Code.
	Data	is the header and data, or header and doc-as-data; *n* depends on the data size of the block.

Figure 23 below shows the response format for the WRITE-DATA chain.

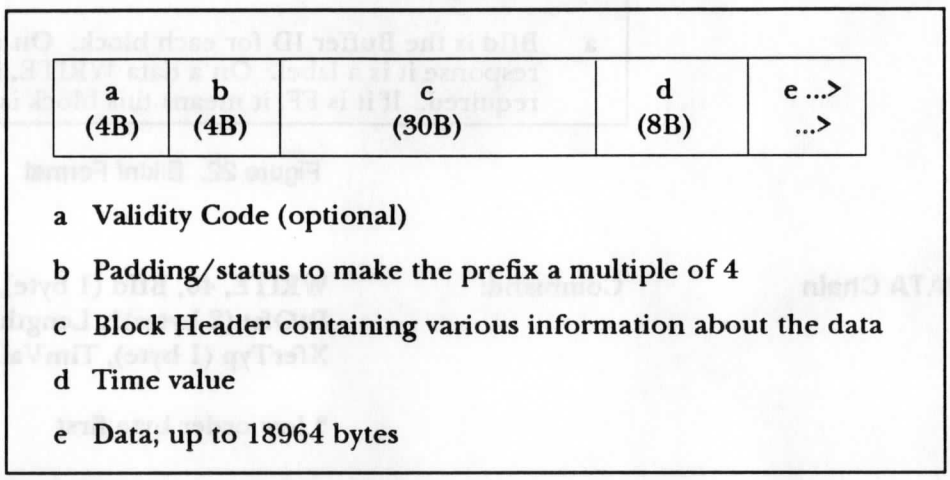


Figure 23. Response Format for the WRITE-DATA Chain

READ Command

The READ command (02) transfers the results of the WRITE commands. It carries the data from the hardware buffer to the mainframe channel.

NOP Command

The NOP command (03) is the null command. No operation is performed. It is sent from the host processor to the channel processor at the beginning of some command chains.

AAA MARK II Ingestor Protocol

This protocol is used for the AAA MARK II ingestor card. The AAA Auxiliary Block (Block 1 or AUXBLK) protocol is described in the previous chapter.

Data can be ingested with the MARK II card alone or with the Library card which is driven by the MARK II card. It is used to average and sample multiband outputs from MSI (Multispectral Imaging) and Dwell Sound signals.

The AAA ingestor protocol consists of the following commands that are sent to the ingestor from the mainframe: NOP, SYNC, WRITE, READ and WAIT.

AAA Raw Data Formats

The chart below shows what products can be obtained from AAA raw data. The abbreviations used in the chart are:

- DS for Dwell Sounding
- MSI for Multispectral Imaging
- IR for Infra Red
- VIS for Visible

Products: Signal	VIS (Single Band)	IR8 (Single Band)	IR16 (Multiband)	IR16 Bits	Grid
A	yes	yes	no	no	no
AA-DS	yes	no	step mode, band 8 only	yes	no
AA-MSI	yes	no	small detector, 3 stage only	yes	no
AAA-DS	yes	no	step mode, band 8 only	yes	yes
AAA-MSI	yes	no	all bands (select bands)	yes	yes
Hardware Mode	1 card	1 card	2 cards	1 card	1 card

Command Chains

A command chain is a series of desired actions that a host processor sends to the channel processor for execution. Figure 24 below shows a sample command chain for the AAA MARK II Ingestor.

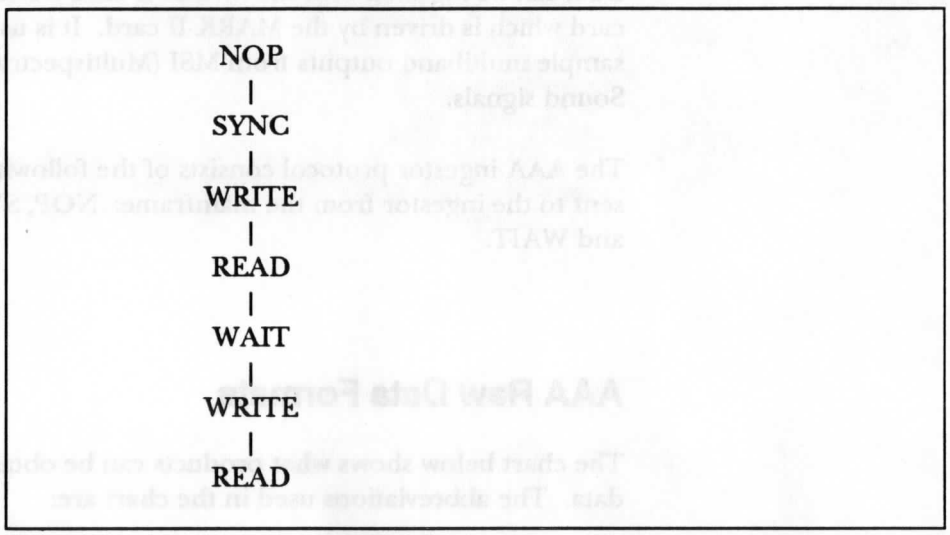


Figure 24. Sample AAA MARK II Command Chain

Command Chain Codes

The General Purpose Channel Interface (GPCI) acts as an interface between the channel and the ingestor hardware. The command chain codes are not transmitted from the channel. The GPCI recognizes the various command chains and generates the command chain codes. These codes are transmitted to the ingestor hardware as the command type. Throughout the remainder of this document, the command type is considered the **first** byte of the command. From an ingestor software or channel perspective, the second byte of a command is the beginning of the command.

The AAA commands are listed beginning on the next page. The value that follows the command name in parenthesis (e.g., SYNC command chain, (07)) is the hexadecimal code for that specific command.

SYNC Command

The SYNC command chain (07) ensures that a data transfer starts at the first sector in the anticipated scan line.

Scan Line Number

The scan line number is important for locating the data in an image. The scan line number for the current data must be inferred using flywheeling if the common doc is bad or the number in the doc is unreasonable. Flywheeling predicts a scan line number for the present spin by using information from preceding spins and an algorithm of how the scan line number should change. Spin information used for the flywheel algorithm is listed below.

- Mode A: Scan numbers increment by one on succeeding spins.
- Modes AA and AAA: Scan numbers in MSI mode increment by one. In DS mode there is an alternating increment/no-increment pattern based on the Program Data Load (PDL). Submodes 0, 1, 3 and 4 always increment the scan on succeeding spins while submode 2 does not. To predict the submode, the band of the next averaged spin must be calculated from the information in the PDL.

The A, AA and AAA modes are further discussed in the *Ingestors Manual*. See Appendix A of the AAA MARK II Primary Ingestor Chapter.

Tracking Discrepancies

When the mainframe scan number does not agree with the Ingestor hardware scan number, contingent activities are carried out by the hardware to bring the mainframe ingestor into sync with the hardware. For example, if the mainframe is tracking scan number n, while the ingestor hardware is tracking scan number n+1, the hardware will answer all commands by declaring that VC=0 until there is a new match. This is carried out seven times and if there is no match after the seventh try, the hardware issues a Unit Exception flag.

If the discrepancy is that the mainframe is tracking scan n+1 while the ingestor is tracking scan n, the ingestor hardware gets new data while overwriting the old. If no match is found after repeating this activity seven times, the ingestor hardware issues a Unit Exception flag.

A summary of these activities appears in the table below.

	Scan Number	Action by ingestor hardware
Mainframe Ingestor	n n+1	Answers all commands with VC=0. If no match occurs by the 7th repetition, the hardware issues a Unit Exception flag.
Mainframe Ingestor	n n	No special action is needed.
Mainframe Ingestor	n+1 n	Overwrites data in the buffer until a match occurs. If no match occurs by the 7th repetition, the hardware issues a Unit Exception flag.

The GPCI monitors each ingestor's status flags. Upon detecting a cleared Busy flag, the GPCI uses the status of the Unit Exception flag to indicate the terminated command's success or failure. The Unit Exception flag is cleared when the command has completed successfully; it is set if the command fails to transfer properly.

WRITE Command

The WRITE command (01) writes common documentation or data files to the mainframe. It is coded using the first byte to designate the WRITE command and the second byte (divided into two nibbles) to designate command parameters.

The second byte of the command is called the Command Code. The WRITE Command Code upper nibble MSB shown in the table below is the Validity Code flag. Next is the Prefix flag followed by the Comdoc flag. The LSB is the Card Count flag. In the table the hexadecimal Command Code values are broken down to show each binary digit, or bit, that represents a flag.

When the prefix bit equals one, the prefix part of an IR sector is sent to the mainframe. If the prefix bit equals zero, the data part of the IR sector is sent. The other flag bits shown in the table below signal:

- whether or not the validity code is used
- whether or not the Comdoc from the first card is sent
- when the Library Card is included in the hardware configuration

Command Code	Validity code flag bit	Prefix flag bit	Comdoc flag bit	Card Count flag bit
0*x*	0	0	0	0
8*x*	1	0	0	0
20	0	0	1	0
A0	1	0	1	0
1*x*	0	0	0	1
5*x*	0	1	0	1
9*x*	1	0	0	1
D*x*	1	1	0	1

Table 1. Command Code Flags

The hexadecimal command code values and associated meanings are shown in the list below.

Command Code	Meaning
0*x*	Data without validity code
8*x*	Data with validity code
20	Comdoc without validity code
A0	Comdoc with validity code
1*x*	Data without validity code, no prefix, 2 cards
5*x*	Data without validity code, prefix, 2 cards
9*x*	Data with validity code, no prefix, 2 cards
D*x*	Data with validity code, prefix, 2 cards

The four low order bits (i.e., the lower nibble) of the second byte of the Command Code give the Output Resolution code.

DOC Chain

Command: DOC (1byte), #Param, VC (4 bytes), Flags, Bands (2 bytes), VIS Scan (2 bytes), Ravscan (2 bytes)

Parameters:

DOC — is the request code.

#Param — is the number of parameter bytes which follow.

VC — is the Validity Code.

Flags — are three bits dedicated to signal the following information:

Bit 0; data origin bit
0= real-time data
1= archive data

Bit 1; data type bit
0= step; non-dwell sub_1 and sub_3 only
1= raw data

Bit 2; raveling bit
0= data not raveled
1= raveled if card 2 is present

	Bands	are the filter bands; they are presented low byte first, high byte second. Each set bit signals that a corresponding band is requested. Only one raveled request is allowed per scan line. Bit 0; band 1 request 0= band 1 is not requested 1= band 1 is requested Bit 1; band 2 request 0= band 2 is not requested 1= band 2 is requested Bit 2; band 3 request Bit 3; band 4 request, etc.
	VIS Scan	is the scan line requested low byte first, high byte second.
	Ravscan	is the scan line requested low byte first, high byte second.

DATA Chain

Command: DATA (1 byte), #Param, Offset (2 bytes), Length (2 bytes), Clip (1 byte)

Parameters:

DATA	is the request code.
#Param	is the number of parameter bytes which follow.
Offset	is the offset from the start of the data buffer; low byte first.
Length	is the length of the data transfer; low byte first.
Clip	is the clip byte count for library data requests.

The responses depicted below are examples of formats that the hardware ingestor uses to send data to the mainframe. The captions under each response explain what type of data is being sent. Figure 25, for example, shows the format in which Visible data is sent to the mainframe. The numbers in parentheses followed by a "B" are byte counts for each field.

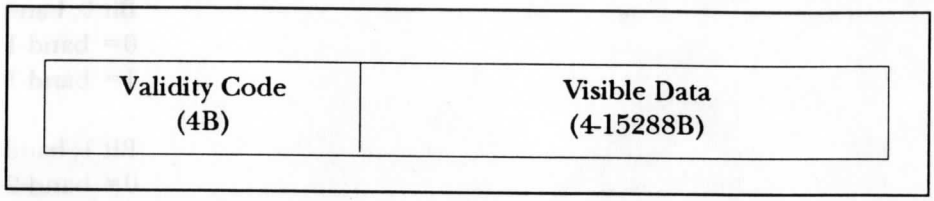


Figure 25. VIS Line Response Format

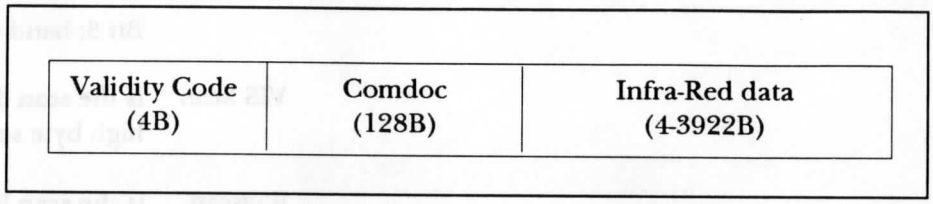


Figure 26. Mode A: IR Line Response Format

In some cases it is necessary to explain on a byte level what is contained in a large and complicated field. Prefix, shown in Figure 27 below, is an example. Figure 28 on the following page shows the Prefix field in more detail.

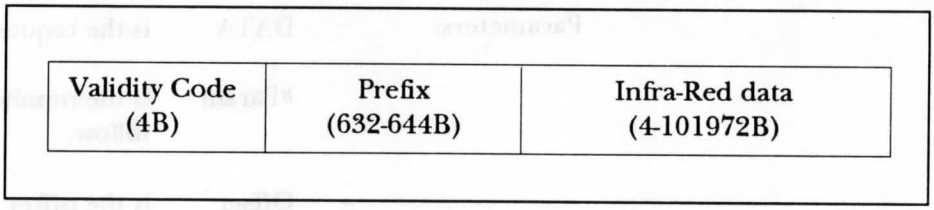


Figure 27. Mode AAA: IR Line Response Format

The Prefix field contains 632 to 644 bytes depending on the number of bands of data that occur in the line. The contents of the Prefix bytes are depicted in Figure 28 below.

Comdoc (512B)	Calibration (116B)	Levels (4, 8, 12 or 16B)

Levels uses one byte for each band in the order that they occur in this line of data.

Figure 28. Prefix Format

The 116-byte Calibration field is broken down into the components shown in Figure 29 below.

a (4B)	b (4B)	c (4B)	d (2B)	e (2B)	f (2B)	g (2B)

a Day — YYDDD, a 4-byte binary integer.

b Time — HHMMSS, a 4-byte binary integer.

c Scan — is the scan number (i.e., image coordinates, line number), a 4-byte binary integer.

d Chan* — is the channel number, a 2-byte binary number.

e nSpin* — is the number of spins, a 2-byte binary number.

f DeltaF* — is the raw line scale factor, a 2-byte nonnegative binary integer.

g Yz* — is the N most significant bits, not including the sign bit, of the space view value; it is a 2-byte nonnegative binary integer.

* Chan, nSpin, DeltaF and Yz are repeated in order 13 times.

Figure 29. Calibration Format

COMDOC Chain

Command: COMDOC (1 byte), VC (4 bytes), Image (1 byte), IR Ravel (scan, bands, other)

Parameters:

VC is the Validity Code.

Image is the image starting flag.

IR Ravel: Scan is the geographic scan to collect for.

IR Ravel: Bands is a bit map of bands desired.

IR Ravel: Other is not yet defined.

The response shown in Figure 30 depicts the format that the ingestor uses to send Comdoc to the mainframe.

Validity Code (4B)	Status (8B)	Comdoc (128, 512B)	Gridbits (0-??)

Figure 30. COMDOC Response Format

The distribution of the 8 Status bytes is shown below in Figure 31.

a (1B)	b (2B)	c (1B)	d (1B)	e (3B)

a Mode is either A, AA or AAA.

b Scan is the microprocessor scan number.

c Flags are shown in detail in Figure 32.

d Dcount is the number of empty DMAs in the previous spin.

e Spares are spare bytes.

Figure 31. Status Format

The assignments for the flag bits used in the Status field are shown in Figure 32.

a	a	a	b	c	d	d	d

a Unused bits

b Sig is the Signal Present Flag.

c SD is the Sector Detector bit when no doc is transmitted.

d Chk is a Check sum bit. One bit is used for each common Doc parity result.

Figure 32. Flag Bits Used in the Status Field

NOP Command

The NOP command (03) is a null command. No operation is performed. It is sent from the host processor to the channel processor at the beginning of some command chains.

READ Command

The READ command (02) transfers the results of the WRITE commands. It carries the data from the hardware buffer to the mainframe channel.

IDLE Command

The IDLE command (0F) responds to the presence or absence of signal. If no signal is present 90 seconds after the command is issued, a Unit Exception flag is set.

WAIT Command

The WAIT command (0B) ensures that new data is accessed beginning in the next sector of the scan line after no additional data from the present scan line is needed.

Common Doc

To properly ingest satellite data, the ingestor hardware needs certain facts which are packaged in the common doc signal. The 8085 ingestor hardware CPU bases decisions on the common doc data. The *McIDAS Ingestors Manual* provides more information about the ingestor hardware.

The Common Doc is transferred in three sections. Section 3 is used only by mode AA ingestors. The numbers found in section 3 are constants for mode AAA ingestors. The common doc sections and their contents are listed below.

Section 1

Word Number	Content
1	retrace
3	satellite ID used by the hardware ingestor for mode determination; word 2 for mode A
4	frame code
6	step code
25 and 26	scan count
28 thru 30	year and day given in YYDDD format
31 thru 33	hour, minute and second given in HHMMSS format
128	longitudinal parity done using special hardware

Section 2

Word Number	Content
20	actual IR header 1
21	actual IR header 2
23-52	Program Data Load (PDL) for 6 scans; the current submode is 23.
68	number of spins on the current band in dwell submode
128	longitudinal parity done using special hardware

Section 3 (Mode AA only)

Word Number	Content
60 and 61	YsubZ IR 1
62 and 63	YsubZ IR 2
64	DeltaF IR 1
65	DeltaF IR 2
128	longitudinal parity done using special hardware

METEOSAT PDUS Ingestor Protocol

This chapter describes the protocol for communication between the mainframe ingestor handler and the METEOSAT PDUS ingestor hardware. The hardware collects and temporarily stores entire scan lines of data. The mainframe host can request all or part of a scan line to be transferred through the Multibus using the General Purpose Channel Interface (GPCI) and the Channel Drivers and Receivers (CDR) boards. The METEOSAT PDUS ingestor does not follow the General Protocol used in the previous chapters of this manual.

Control Fields

The following list defines the METEOSAT PDUS control fields which are presented by the satellite or the ingestor hardware for use by the ingestor software. Before it can proceed to ingest or ignore an image, the ingestor handler needs to know the quality and type of signal present. METEOSAT PDUS transmits many of these control fields in the Label of each subframe. A few of the control fields come directly from the hardware for the ingestor software to use.

Current subframe number determines which subframe (header, data, annotation or conclusion) of the signal is currently being encountered by the ingestor.

Data present field indicates whether or not data is present. The ingestor software must check for requests as soon as the hardware informs it that data is present.

Data type fields determine if Vis, IR or WV data is present and if the data is Res 1 or Res 2. A match is done on these fields, requests and resolutions to determine what data is to be ingested.

Format indicator indicates if the signal is in A type format or B type format to determine the image size and/or resolution, etc. The format indicator also matches the requests for data to signals present in order to ingest the appropriate data.

Frame-per-subframe has a value of either 4 (A type format) or 8 (B type format) throughout a METEOSAT image. It is used as a consistency check.

Frame Sync lock field is a field from the hardware which indicates the status of the frame sync. If it is locked to a signal, the ingestor handler should start looking for data. If it is not locked, the ingestor handler will keep idling.

Image number from start of mission is a unique number given to each image. The number should remain the same throughout the image and can be used for signal quality checks.

Satellite Day, Time field determines which calibration tables to use for the satellite bird and whether or not data is real-time. Because it is consistent throughout a signal, it is used to match requests and check signal quality.

Satellite name determines the satellite number used to match a request window. It should remain constant throughout an image. A satellite name change indicates poor signal quality or a different image from a new satellite.

Scan direction determines whether the scan direction is S-N or N-S and E-W or W-E. The ingestor handler uses the scan direction field to determine which counting and disk writing algorithms to use to ingest the signal.

Scanline number field contains the information to determine where to position subsequent subframes in the ingestor areas. Since this number is always increasing (or decreasing) it is also used as a quality check.

Subframe-per-image field contains the information to determine if an image has full globe Res 2 Vis and IR data.

Validity Code is a number created by the ingestor software once for each image. The hardware sends the validity code back at the front of each image line if the data is good. If a line is missed or garbled, the validity code is returned from the hardware as a zero.

METEOSAT Signal Formats

METEOSAT PDUS data ingested by SSEC hardware is formatted in either A type format or B type format. A type format images cover more territory geographically. A B type format image is a subset of an A type format image. The basic unit of data in either format is called a subframe. Table 2 on page 8-6 summarizes these formats.

Subframes

An image can contain header subframes, image data subframes, annotation subframes (found in B type formats only) and conclusion subframes. The total number of subframes depends on the format and the bands being transmitted. If an image line contains IR and WV data, there will be more subframes than if it contains only IR or WV data. Table 2 on page 8-6 illustrates this data formatting in more detail.

The number and type of subframes in the A type format and B type format images are shown below.

Subframe type	A type format images	B type format images
Header subframes	42	84
Image Data subframes	variable*	variable*
Annotation subframes	0	30
Conclusion subframes	42	84

* Table 2 on page 8-6 shows the number of data subframes for specific formats.

Header Subframes

Each frame in a header subframe is 364 bytes long. A frame begins with a 3-byte synchronization code followed by a 1-byte ID. 360 bytes remain for data. Four of these 360 byte packages are transferred for B type format data while 8 are transferred for A type format data. Byte designations for some of the major fields follow:

- The label uses 24 bytes (described on the following page).
- The Identification Section uses 32 bytes and contains the satellite name, time and date of image.
- Interpretation data uses 1360 bytes and contains the calibration data.
- A type format header subframes have 1440 spare bytes available.

The content of each byte of the 24-byte label is shown below. The fields found in the label are described in the Control Fields glossary found on pages 8-1 and 8-2.

Label Bytes	Content
1, 2	number of frames in the subframe; 8 for A type format, 4 for B type format
3, 4	total number of subframes in the image; all header subframes count as 1, all conclusion subframes count as 1
5, 6	current subframe number
7, 8	scan line number
9-12	image number
13	Format indicator: 00 = A type format FF = B type format
14	spectral indicator byte: F0 = first half of Vis1 0F = second half of Vis1
15	spectral indicator byte: F0 = first half of Vis2 0F = second half of Vis2
14, 15	used together as AIVH Vis format flag
16	Infra-Red data flags: 00 = IR data is not present FF = IR data is present
17	Water Vapor data flags: 00 = WV data is not present FF = WV data is present
18	Reference grid flags: 00 = reference grid is not present 0F = reference grid is present
19	Annotation text flags: 00 = annotation text is not present FF = annotation text is present
20	Scan direction: 00 = scan is S-N and E-W F0 = scan is N-S and E-W 0F = scan is S-N and W-E FF = scan is N-S and W-E
21, 22	Spares, not read
23, 24	Set to zero, not read

Image Data Subframes

Except for the AIVH format, an image data subframe contains either:

- one complete scan line of an IR image
- one complete scan line of a WV image,
- one complete scan line of a Vis image at Res 2, or
- one quarter of a scan line of a Vis image at Res 1

An A type format subframe contains 2500 bytes of data; a B type format subframe contains 1250 bytes of data.

Figures 33 and 34 below illustrate how the same geographical area is covered by either a Res 1 or Res 2 scan line. Using Res 1 takes four times as many pixels to cover the same area using Res 2. As a result, the Res 1 scan line contains 4 subframes while the Res 2 scan line contains one subframe.

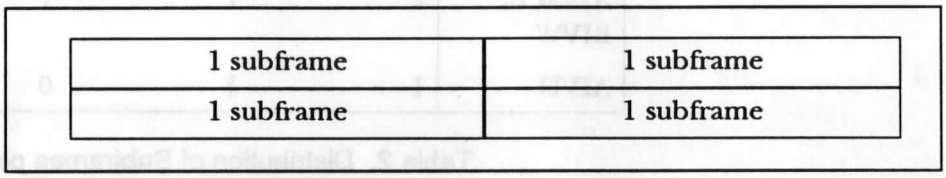


Figure 33. Scan Line at Res 1

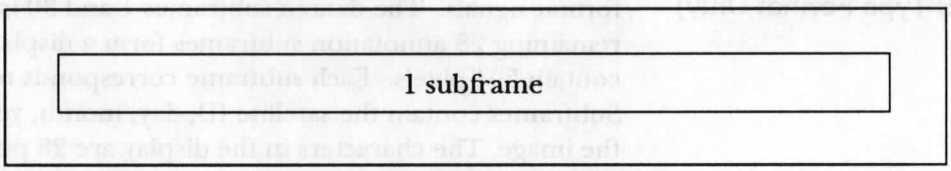


Figure 34. Scan Line at Res 2

AIVH (see Table 2 on the following page) is the only data configuration in which visible data is received at Res 2.

An intermediate format (see AIVW or BIVW in Table 2) uses Res 1 but includes data from only one of the 2 Vis lines shown above in Figure 33. This format is shown in Figure 35 below.

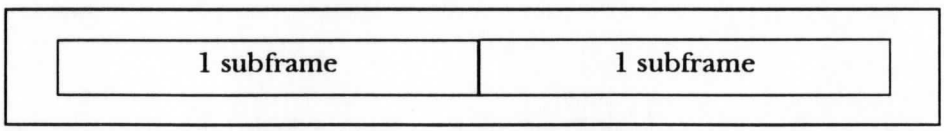


Figure 35. A Two-Subframe Scan Line at Res 1

Table 2 below details the subframes per scan line of various transmission formats.

Transmission Format	Vis Subframes	IR Subframes	WV Subframes	Total Subframes per scan line
AV and BV	4	0	0	4
AI and BI	0	1	0	1
AW and BW	0	0	1	1
AIV and BIV	4	1	0	5
AIW and BIW	0	1	1	2
AIVW or BIVW	2	1	1	4
AIVH	1	1	0	2

Table 2. Distribution of Subframes per Scan Line for Various Transmission Formats

Annotation Subframes (B Type Format Only)

There are 30 annotation subframes per image found only in B type format signals. The data in subframes 1 and 30 is set to zero. The remaining 28 annotation subframes form a display matrix of 28 lines that contain 504 pixels. Each subframe corresponds to one image line. Subframes contain the satellite ID, day, month, year, hour and minute of the image. The characters in the display are 28 pixels high and 20 pixels wide; any remaining pixels are set to zero. This data is not requested by the mainframe handler in formatted mode.

Conclusion Subframes

The Conclusion subframe is a copy of the Header subframe.

Command Chains

The METEOSAT PDUS ingestor runs in a formatted mode, where only data is read and sent to the host, or in an unformatted mode, where the ingestor reads everything that is transmitted by the satellite. During normal operation the ingestor operates in a formatted mode. The unformatted mode is used when the signal is questionable and/or during development.

The METEOSAT PDUS ingestor uses three command chains to acquire data. Commands are explained beginning on page 8-9.

Command Chain 1 This command chain waits for the first good header subframe in a formatted mode.

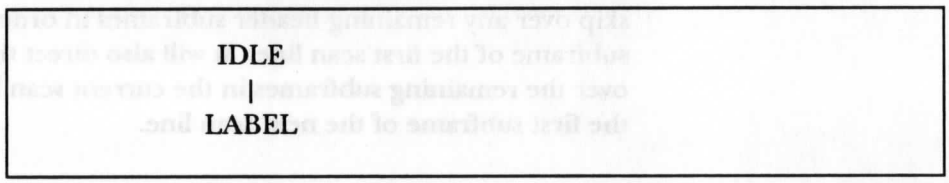


Figure 36. METEOSAT PDUS Command Chain 1

Command Chain 2 This command chain waits for the first occurrence of any subframe in an unformatted mode and reads data immediately.

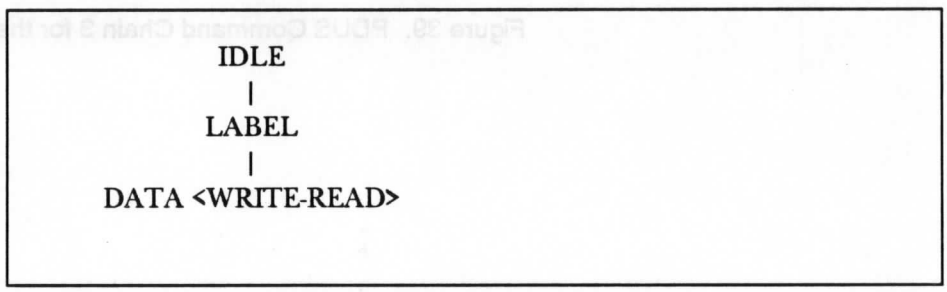


Figure 37. METEOSAT PDUS Command Chain 2

Command Chain 3

Command Chain 3 uses the WAIT command to wait for the next appropriate subframe. It then reads a label and data.

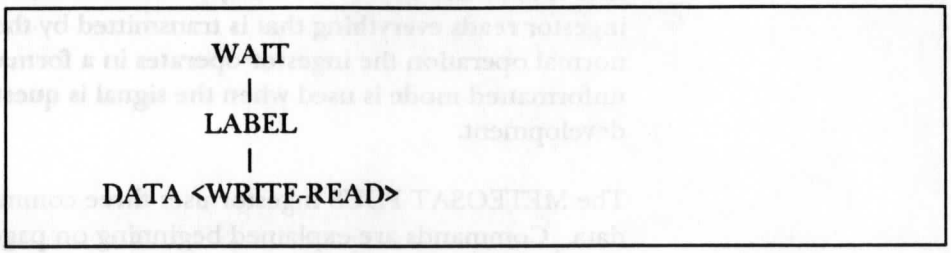


Figure 38. PDUS Command Chain 3 for the Unformatted Mode

In the formatted mode, Command Chain 3 uses the SYNC command to skip over any remaining header subframes in order to reach the first subframe of the first scan line. It will also direct the hardware to skip over the remaining subframes in the current scan line in order to reach the first subframe of the next scan line.

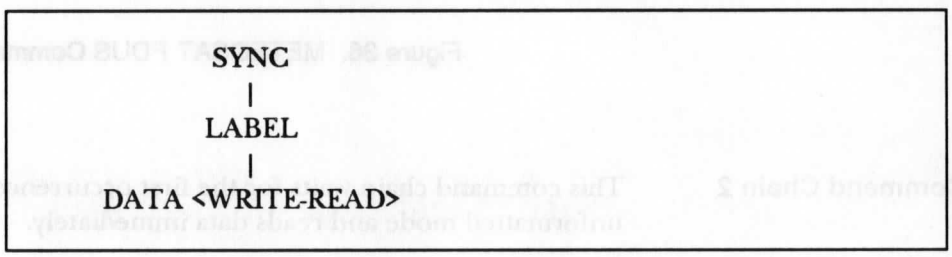


Figure 39. PDUS Command Chain 3 for the Formatted Mode

IDLE Command

The IDLE command (0F) tells the hardware to wait for data. When data becomes available, control passes to the next command in the chain. (0F) is the hexadecimal code for the IDLE command.

SYNC Command

The SYNC command (07) tells the hardware to wait for the first subframe of the next scan line while in a header or data section.

In the formatted mode, after a header subframe is interpreted and the requests are set up, there is no longer a need to read the remaining header subframes. SYNC skips over any remaining header subframes in the image in order to reach the first data subframe.

WAIT Command

The WAIT command (0B) tells the hardware to wait for the next data subframe. This is in contrast to the SYNC command which waits for the first subframe of the next scan line.

LABEL Command

The LABEL command (20 or A0) tells the hardware to send the signal label for control characters. The LABEL command is coded as 20 hexadecimal when no validity code is used. If a validity code is used with the signal being ingested, LABEL is coded as A0 hexadecimal. The response to a LABEL command is depicted below in Figure 40. The Status bytes are shown in detail in Figure 41.

Command: LABEL, 20/A0, VC (4 bytes; optional)

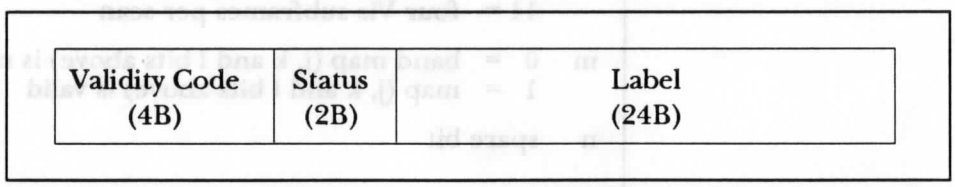


Figure 40. Response Format for the LABEL Command

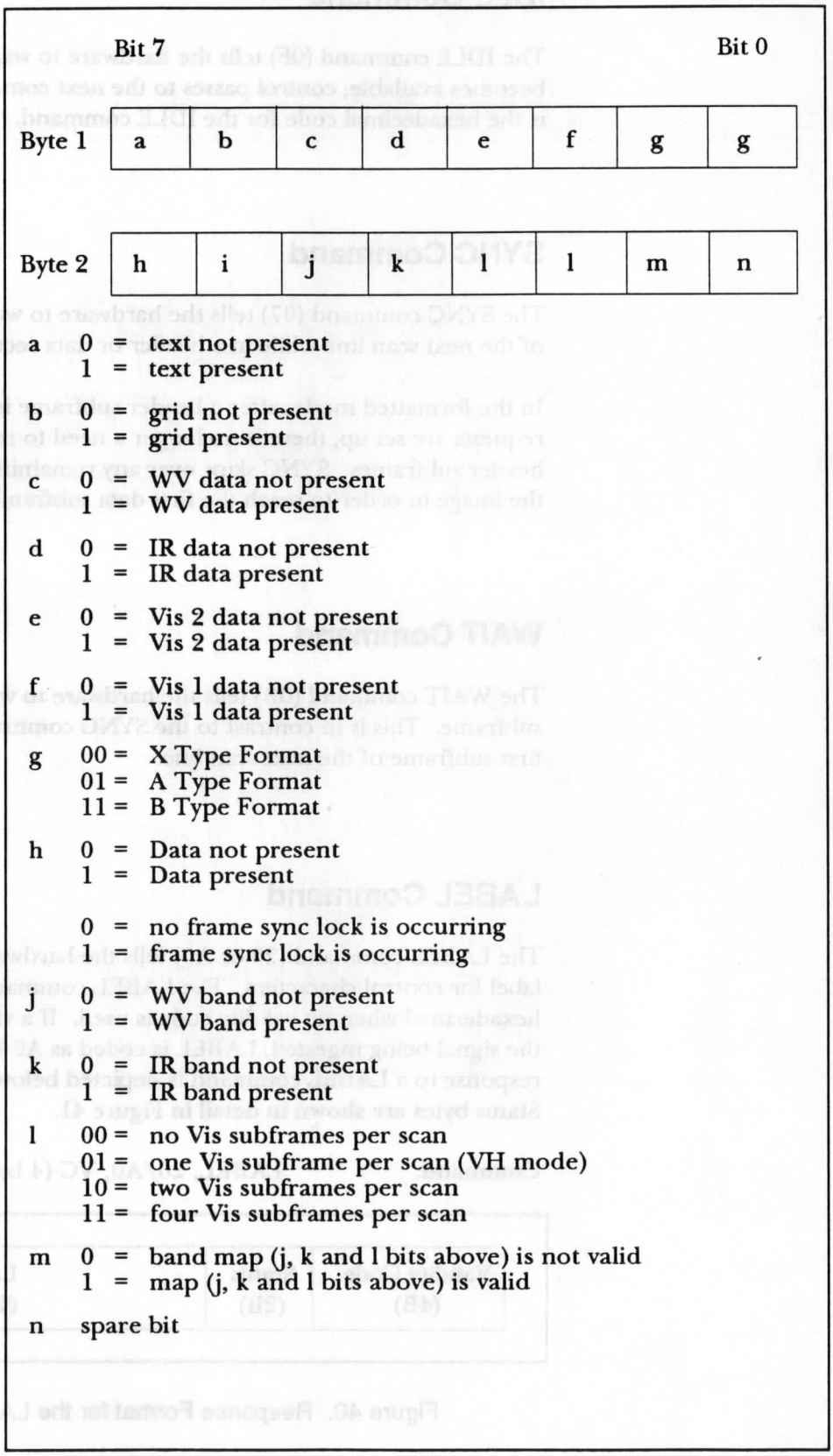


Figure 41. Status Bytes' Format

WRITE Command

The WRITE command tells the hardware to send information to the mainframe. There are two forms of the WRITE command, DIRECT WRITE and REVERSE WRITE. The WRITE response shown below applies to both forms. Details of the Label parameter are shown on page 8-4.

Command: WRITE (00/01, 80/81, 40/41 or C0/C1), offset (2 bytes)*, #bytes (2 bytes)*

* lower order byte first

Parameters: offset is the offset into the data field.

#bytes is the number of bytes requested.

Validity Code (4B)	Label (24B)	Data (*n*B)

Figure 42. Response Format for both WRITE Commands

DIRECT WRITE

The DIRECT WRITE command (00, 80, 01 or 81) sends data in the same direction that it was stored, beginning at the first byte in the buffer. In this manner, data that was scanned from west to east by the satellite is sent beginning with the westmost data and finishing with the eastmost data. REVERSE WRITE, described on the following page, sends the data in reverse order.

The first nibble of the command code depends on the use of a validity code with the ingested signal. If the validity code is used, nibble 1 is eight; if no validity code is used, nibble 1 is zero.

The second nibble of the command code is the output resolution code. A zero in the second nibble position specifies no signal sampling; every byte is sent to the ingestor handler so the signal is ingested exactly at the transmitted resolution. A one specifies signal sampling; every other byte is transmitted. Using sampling, a Res 1 signal can be ingested as Res 2; a Res 2 signal can be ingested as Res 4 because only half the bytes are transmitted.

REVERSE WRITE

The REVERSE WRITE command (40, C0, 41 or C1) sends data beginning at the last byte in the buffer; it sends data in a direction opposite to that in which it was stored. In this manner, data that was scanned from east to west can be sent beginning with the westmost data and finishing with the eastmost data.

The first nibble of the command depends on the use of a validity code with the ingested signal. If the validity code is used, nibble 1 is C (hexadecimal twelve); if no validity code is used, nibble 1 is four. The second nibble of the command code is the output resolution code. A zero in the second nibble position specifies no signal sampling; every byte is sent to the ingestor handler so the signal is ingested exactly at the transmitted resolution. A one specifies signal sampling; every other byte is transmitted. Using sampling, a Res 1 signal can be ingested as Res 2 and a Res 2 signal can be ingested as Res 4 because only half the bytes are transmitted.

POES Ingestor Protocol

This chapter describes the protocol for communication between the mainframe ingestor handler and the POES ingestor hardware. The POES scan line consists of sectors. Common documentation is broken into several parts. The POES ingestor card listens to the POES signal, waiting for commands from the mainframe ingestor handler. It synchronizes the mainframe to the ingestor signal. The commands sent from the mainframe ingestor handler to the hardware are shown on the following page.

Control Fields

The following list defines the POES control fields which are presented by the satellite or the ingestor hardware for use by the ingestor software. Before it can proceed to ingest or ignore an image, the ingestor handler needs to know the quality and type of signal present. POES transmits many of these control fields in the common documentation section of the signal.

Frame Sync Power is an on or off warning. When it is off, the ingestor handler will not ask for data.

Frame Sync Type is used as a consistency check for hardware and is either EMR, Dornier, Generic Serial or Generic Parallel.

Satellite Day is used to match an image request. A satellite day change indicates that the data just received is from a new image.

Satellite ID is a number used in matching a request window. It does not change throughout an image.

Satellite Time is used to match an image request. A satellite time change, which varies from normal time increments, indicates that the data just received is from a new image.

Scan Number is the number that tells the ingestor software where it is in the image. It directs where to write the data in the image area.

Signal Source applies to the EMR frame sync only. It is used as a consistency check and is displayed in the ingestor status display. The signal source is either Wallops or Gilmore.

Signal Type is used to match a request and determine whether the data just received is from a new image.

Stability refers to satellite stability. It is a variable used for checking signal consistency and/or quality.

Validity Code is a number created by the ingestor software once for each image. The hardware sends the validity code back at the front of each image line if the data is good. If a line was missed or garbled, the validity code is returned as a zero.

Command Chains

There are three types of command chains from the mainframe host to the POES ingestor card. Each command chain is shown below and is discussed further in the WRITE command section.

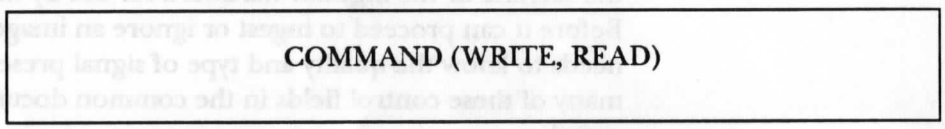


Figure 43. Command Chain Type 1: COMMAND

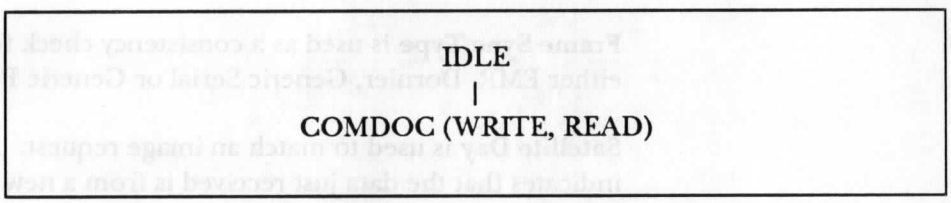


Figure 44. Command Chain Type 2: COMDOC

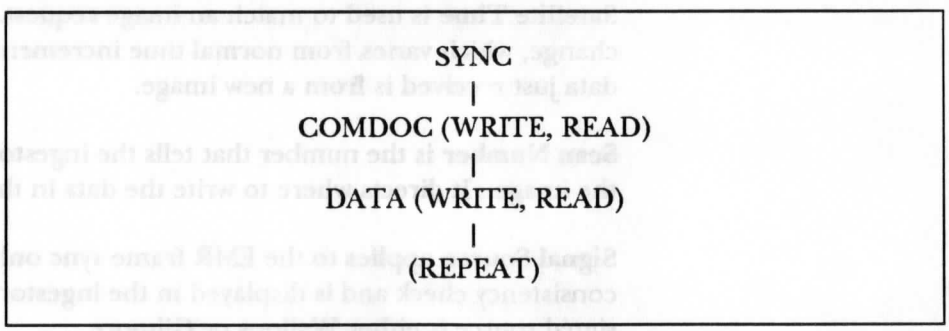


Figure 45. Command Chain Type 3: DATA

IDLE Command

The IDLE command (0F) waits until the frame synchronizer indicates that the signal is present. The hardware will timeout after 150 seconds if no signal is found and a Unit Exception flag is issued via the status byte.

SYNC Command

The SYNC command (07) waits until the beginning of the next line. A Unit Exception in this command means that the frame sync has lost lock, i.e., a signal is no longer present. The number of minor frames that elapse before issuing a Unit Exception flag is determined empirically.

READ Command

The READ command (02) transfers the data from the buffer to the channel as directed by the previous WRITE command. This command structure comprises a WRITE-READ pair.

WRITE Command

The WRITE command (01) may consist of 2 to 16 bytes. The first byte is generated by the GPCI. It identifies the command as a WRITE command. The remaining bytes are interpreted as WRITE command parameters and have the following format: **WR** *pi p1 p2 p3 . . . px*, where *p1 . . . px* represent command parameters.

The WRITE command has three chain formats (see Figures 43 through 45), COMMAND, COMDOC and DATA. In each command, the four high order bits of the second byte signify the command code (chain format). The hexadecimal code values and their meanings are listed on the following page.

Command Code	Meaning
10	COMMAND to the preprocessor (e.g., change the signal source)
00	COMDOC with validity code
80	COMDOC without validity code
4*y*	DATA request with validity code
C*y*	DATA request without validity code

For WRITE-DATA commands, the four low order bits of the second byte signify the sample/average data process instruction and are represented by the *y* in the command code. The hexadecimal values for the process instructions and their meanings are listed below. The 8- and 10-bit output packing formats are shown in Figures 46 and 47.

Process Instruction	Meaning
40	Data from all sensors using the 8-bit format
41	Data from all sensors using the 10-bit format
42	Data from one sensor using the 8-bit format
43	Data from one sensor using the 10-bit format
C0	Data from all sensors using the 8-bit format with no VC
C1	Data from all sensors using the 10-bit format with no VC
C2	Data from one sensor using the 8-bit format with no VC
C3	Data from one sensor using the 10-bit format with no VC

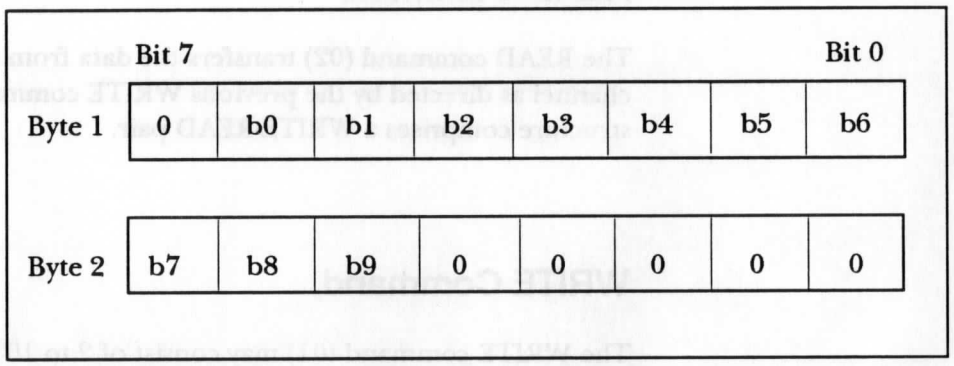


Figure 46. 10-bit Output Bit-Packing Format

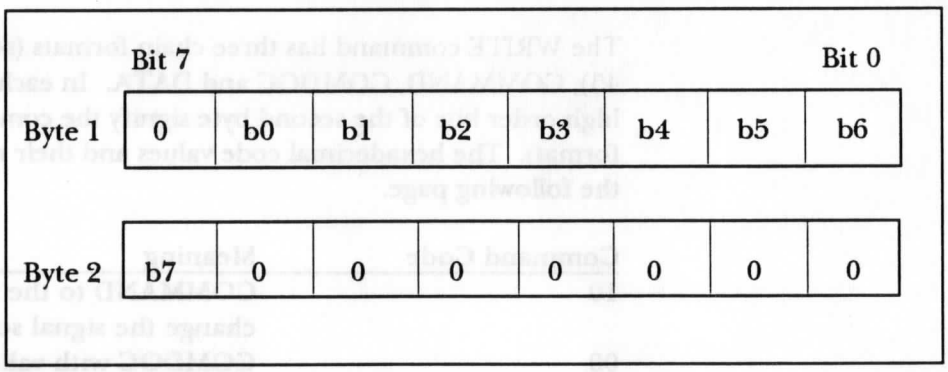


Figure 47. 8-Bit Output Bit-Packing Format

COMDOC Chain

Command: WR (COMDOC), 2/A0 (1 byte), VC (4 bytes)

Response: VC (4 bytes; optional), Status (12 bytes), Comdoc (192 bytes)

Response Fields:

VC — is the Validity Code.

Status — is defined depending on the type of Frame Sync used with the ingestor system. See the Status byte figures that follow.

Comdoc — is the common documentation containing facts needed by the ingestor hardware to properly ingest satellite data. The first 96 words (192 bytes) are sent automatically following a LOCK signal from the Frame Sync.

Details of the COMDOC Response format are shown below. The Status byte format figures begin on the following page.

a (4B)	b (12B)	c (192B)

a Validity code (when enabled)

b Status is a function of the type of Frame Sync used in the system. See pages 9-6 through 9-9.

c COMDOC; the first 96 words are automatically sent by the hardware following a LOCK.

Figure 48. COMDOC 10-Bit Response Format

	Bit 7							Bit 0
Byte 1	s2	s1	m2	m1	f3	f2	f1	bc

s2, s1 **00 = EMR Comsat** (In use for this example)
01 = EMR direct
10 = generic serial
11 = generic parallel (Dornier)
See Frame Sync Status 1 MODE bits for more detail.

m2, m1 **00 = other** (In use for this example)
01 = LAC
10 = GAC
11 = HRPT (2x)

f3 0 = Frame Sync in local
1 = Frame Sync in remote

f2 0 = Frame Sync power off
1 = Frame Sync power on

f1 0 = Frame Sync no LOCK
1 = Frame Sync in LOCK

bc 0 = no error on command
1 = error on command
Only asserted following a command to the preprocessor.

Byte 2 Spare

Byte 3 Frame Sync Status 1 (Operational)

Byte 4 Frame Sync Status 2 (Test Generation)

Byte 5 Quality 1 (Normal/Pseudorandom Noise)

Byte 6 Quality 2 (Normal/Pseudorandom Noise)

Bytes 7 - 12 Spares (All 00s)

Figure 49. Status Bytes for EMR-Comsat Frame Sync

	Bit 7							Bit 0
Byte 1	s2	s1	m2	m1	f3	f2	f1	bc

s2, s1 00 = EMR Comsat
01 = EMR direct (In use for this example)
10 = generic serial
11 = generic parallel (Dornier)
See Frame Sync Status 1 MODE bits for more detail.

m2, m1 **00 = other** (In use for this example)
01 = other
10 = other
11 = HRPT

f3 0 = Frame Sync in local
1 = Frame Sync in remote

f2 0 = Frame Sync power off
1 = Frame Sync power on

f1 0 = Frame Sync no LOCK
1 = Frame Sync in LOCK

bc 0 = no error on command
1 = error on command
Only asserted following a command to the preprocessor.

Byte 2 Spare

Byte 3 Frame Sync Status 1 (Operational)

Byte 4 Frame Sync Status 2 (Test Generation)

Byte 5 Quality 1 (Normal/Pseudorandom Noise)

Byte 6 Quality 2 (Normal/Pseudorandom Noise)

Bytes 7 - 12 Spares (All 00s)

Figure 50. Status Bytes for Frame Sync Case 1: EMR-Direct

	Bit 7							Bit 0
Byte 1	s2	s1	m2	m1	f3	f2	f1	bc

s2, s1 00 = EMR Comsat
01 = EMR direct
10 = generic serial (In use for this example)
11 = generic parallel (Dornier)
See Frame Sync Status 1 MODE bits for more detail.

m2, m1 00 = illegal
01 = illegal
10 = illegal
11 = HRPT (In use for this example)

f3 0 = Frame Sync in local
1 = Frame Sync in remote

f2 0 = Frame Sync power off
1 = Frame Sync power on

f1 0 = Frame Sync no LOCK
1 = Frame Sync in LOCK

bc 0 = no error on command
1 = error on command
Only asserted following command to preprocessor.

Byte 2 Spare

Byte 3 Spare

Byte 4 Spare

Byte 5 Spare

Byte 6 Spare

Bytes 7 - 12 Spares (All 00s)

Figure 51. Status Bytes for Generic Serial Frame Sync

	Bit 7							Bit 0
Byte 1	s2	s1	m2	m1	f3	f2	f1	bc

s2, s1 00 = EMR Comsat
01 = EMR direct
10 = generic serial
11 = generic parallel (Dornier) (In use for this example)
See Frame Sync Status 1 MODE bits for more detail.

m2, m1 00 = other
01 = LAC
10 = GAC
11 = HRPT (In use for this example)

f3 0 = Frame Sync in local
1 = Frame Sync in remote

f2 0 = Frame Sync power off
1 = Frame Sync power on

f1 0 = Frame Sync no LOCK
1 = Frame Sync in LOCK

bc 0 = no error on command
1 = error on command
Only asserted following command to preprocessor.

Byte 2 Spare

Byte 3 Spare

Byte 4 Spare

Byte 5 Spare

Byte 6 Spare

Bytes 7 - 12 Spares (All 00s)

Figure 52. Status Bytes for Generic Parallel (Dornier) Frame Sync

COMMAND Chain

Command: WR (COMMAND), 10 (1 byte), Parameters (3 bytes)

Parameters:

P1 changes the signal source or sets the Frame Sync as indicated:
00 = Wallops
01 = Gilmore
02 = Spare
03 = Test Generator
10 = Set the Frame Sync (see P2 and P3 below)

P2 and P3 are 00 for all cases except when P1 = 10, then the first byte represents bits 1-8 of Group Operational of the Remote Programming Format; the second byte is bits 1-8 of the Group Test.

The Legal Modes when listening to the Comsat are GAC, LAC, and HRPT(2x).

SOURCE SELECT is defined as shown:
Bit 3 must be 1
Bit 4 must be 0
Bit 5 must be 1
Bit 9 is set to a 0 automatically by the preprocessor.

Response: Status (12 bytes)

Response Field: Status is defined depending on the type of Frame Sync used with the ingestor system. See status byte figures 49 through 52.

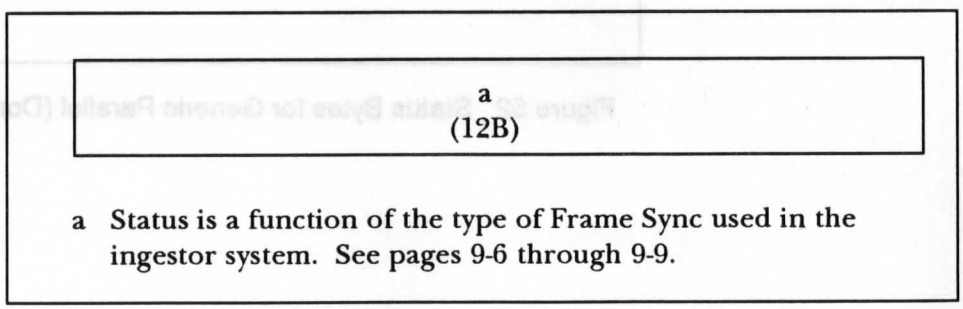


Figure 53. COMMAND Response Format

DATA Chain

Command: WR (DATA), 0/80 or 0/83 (1 byte), Offset (2 bytes), Length (2 bytes)

Parameters:

Offset is the offset from the start of the data buffer, low byte first.

Length is the length of the data transfer, low byte first.

The DATA Response examples shown below are for data from all five sensors.

a (4B)	b (2 - 20480B)

a Validity code (when enabled)

b DATA will need two requests because length is larger than 16K.

Figure 54. DATA 10-Bit Response Format; Mode HRPT, LAC

a (4B)	b (1 - 10240B)

a Validity code (when enabled)

b DATA

Figure 55. DATA 8-Bit Response Format; Mode HRPT, LAC

a (4B)	b (2 - 6454B)

a Validation code (when enabled)

b DATA

Figure 56. DATA 10-Bit Response Format; Mode GAC (Raw)

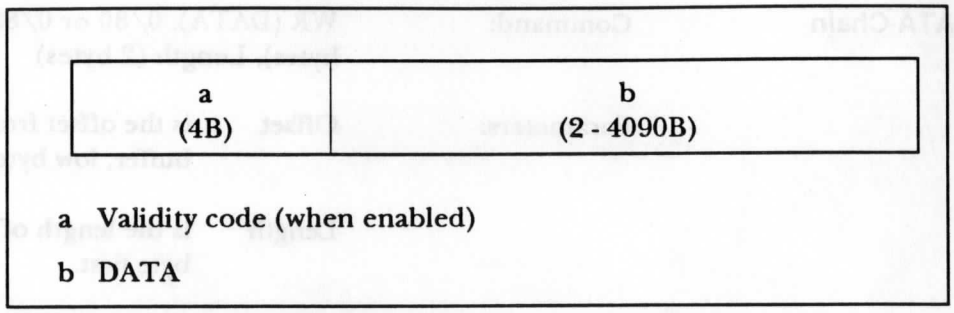


Figure 57. DATA 10-Bit Response Format; Mode GAC

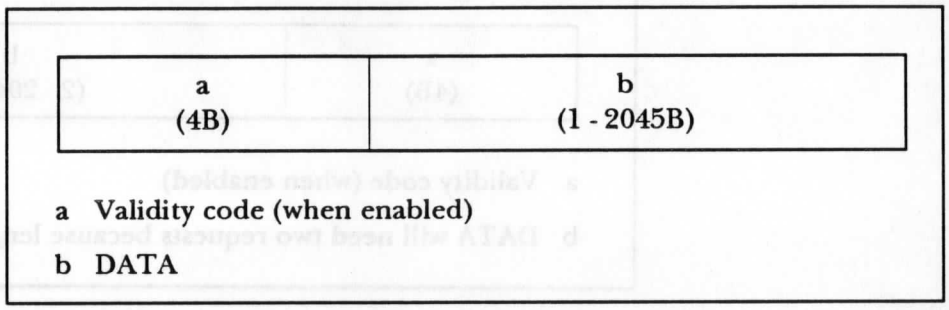


Figure 58. DATA 8-Bit Response Format, Mode GAC

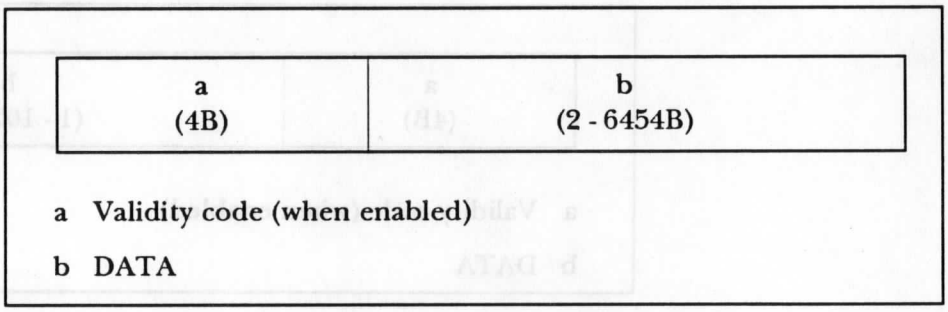


Figure 59. DATA 10-Bit Response Format, Mode GAC (Raw)

89108788530
b89108788530a