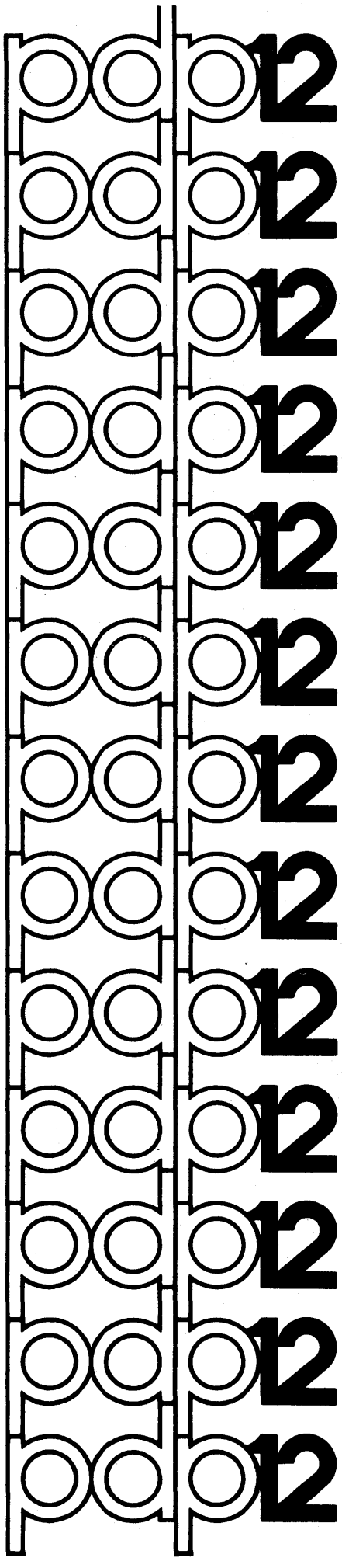
digital



MIDAS

USER'S MANUAL

DEC-12-SQ3A-D

MIDAS USER'S MANUAL

For additional copies, order No. DEC-12-SQ3A-D from
Program Library, Digital Equipment Corporation,
Maynard, Massachusetts, 01754 Price $5.00

JULY 1971

The material in this manual is for information purposes and is subject to change without notice.

PREFACE

The MIDAS User's Manual (DEC-12-SQ2A-D) provides a detailed description of the MIDAS and MSORT programs.

Terminology and concepts specific to this software are defined in the Glossary; however, the manual assumes that ideas unique to the LDP system are familiar to the reader and treats them either superficially or not at all. Those who are already expert in the use of MIDAS and MSORT may refer directly to the Operation Summary sheets in the Appendices B and C of this manual. These summary sheets have also been printed on perforated pages at the end of this document so they may be removed and displayed on the computer.

Complete understanding of the contents of this manual requires the reader to be familiar with the AIPOS User's Guide, particularly Chapter 1 and Appendix A.

CONTENTS

1.0 INTRODUCTION

MIDAS (Multi Instrument Data Acquisition Software) is a general data acquisition program for the LDP system which acquires data from multiple instruments (8) in a synchronous or asynchronous manner and throughputs that data to mass storage (RK8 Disk or LINCtape). The maximum data rate for a given instrument or the maximum across a number of them depends on hardware configuration but can be as high as 15ØØØ Hertz.

MIDAS allows control over experiments via Schmitt Triggers, external syncs, analog inputs, clock, sense lines, relays, and keyboard. The controls may be dependent upon or independent of other on-line instruments in the MIDAS environment. The program recognizes all AIP hardware configurations, extended core, and up to 8 mass storage devices, and also allows setup of instruments while acquiring data from other on-line experiments.

MSORT (MIDAS Sorter) converts the output of MIDAS from a given experiment into a three word floating point format that is interpretable by other LDP software, e.g., DORA.

2.0 HARDWARE REQUIREMENTS

Minimum hardware for MIDAS is

PDP-12/40
AIP-12 Analytical Instrument Processor

Minimum hardware for MSORT is

PDP-12/40
(KW12A clock is not required)

Although an RK8 disk and more than 8K of core are not required hardware, they will provide significant improvements in aggregate throughput and sorting. MIDAS supports up to 24K of core, MSORT supports up to 2ØK of core.

3.0 MIDAS

3.1 General Description

For proper run time execution MIDAS requires the hardware cited above (refer to Section 2.0) and at least one LDP volume with JCL (Job Control), the Monitor, the MIDAS binary and a WA (Working Area) sufficiently large for data storage[1]. MIDAS is called from JCL (refer to the AIPOS User's Manual, DEC-12-SØ1A-D) in the usual manner; however, the keyboard input at JCL time has a specific meaning to MIDAS (refer to 3.2 Job Control Time).

MIDAS acquires data in either 12 or 15 bit format from any of the 16 possible AIP analog channels (4Ø-57) or digital data on channels 2Ø-27 in either 18 bit binary or BCD to binary format[2]. The program acquires data in a synchronous or periodic fashion under control of the KW12A clock or in an asynchronous manner via the PDP-12 A/D or AIP-12 external sync and essentially under control of the instrument. Data is stored on mass storage in AIP-12 double word format (refer to AIP-12 Operations Manual) which means that each datum requires two words of mass storage and a block of LINCtape or Disk holds 128 points. If a LINCtape or Disk is virtually free, i.e., almost all working area, it can hold about 100,000 or 400,000 points respectively. The MIDAS data acquired across a number of runs is stored on a MIDAS file in a mixed fashion along with certain control words. The control information is interpreted at sort time and allows MSORT to isolate all points of a given run and convert them into an LDP floating point file.

3.2 Job Control

The first task is to load AIPOS and call MIDAS from Job Control (when Job Control is running it is said to be at JCL TIME). The most general format of the command is:

dev:MIDAS dev:PRIME,dev:FILE1,...,dev:FILE7;LABELØ,...,LABEL 7

---

[1] As a general rule of thumb, a 62_8 block WA is always big enough for MIDAS output. The primary MIDAS file has 19_{10} header blocks and each secondary MIDAS file has 1 header block.

[2] AIP channels 50-57 require the AIP expander, digital channels 22-27 are not yet implemented in the hardware. BCD to Binary format has certain implications for throughput and MSORT (see Sections 3.5.7 and 4.4.3).

MIDAS has no use for input files and ignores any that are specified. At least one output file must be explicitly defined for MIDAS (JCL will not default to one for it); more than one is optional. MIDAS uses the LDP output volumes specified at JCL time to store data at acquisition time. MIDAS addresses them sequentially beginning with the first. The output files specified at JCL time that contain data at EXIT time are saved on their respective volumes under the names specified at JCL time; those that do not contain any data are not saved. In particular, the first output file is always saved, and all saved output files with the possible exception of the last one written on exhaust the JCL time WA and therefore fill their respective volumes, i.e., a DX (Display Index) upon return to JCL does not display any working area for that volume.

The first output file specified to JCL is the primary MIDAS output file and has special properties not found in subsequent output files (the MIDAS secondaries). The LDP volumes and associated output file names specified at JCL time are said to define one logical MIDAS file. Note that a logical MIDAS file is conceptually different from a standard LDP file because it is possible for a logical MIDAS file not to be volume contained. If as many output files are specified as there are devices, a sort problem may be inadvertently created (including the possibility of not being able to sort at all).

The parameter string (LABELØ,...,LABEL7) comprises the volume labels for output files PRIME,...,FILE7 respectively and is optional JCL time input. The labels allow physical or visual identification of a MIDAS volume.

At sort time, a logical MIDAS file may be identified by sort device, MIDAS output file name, and label (refer to 4.4.2 ID Command). A label is delimited by a comma or six characters, whichever comes first in the parameter string. Because the parameter string cannot be more than 32 characters long, six 6-character labels cannot be specified. For example, the following parameter string:

```
;1234567,89
```

would be interpreted by MSORT as 3 labels,

```
LABELØ      123456
LABEL1      7
LABEL2      89
```

where label "123456" is assigned to the primary file; "7" and "89", to the first and second secondaries, if they exist.

One does not have to specify the JCL command string in its most general format to call MIDAS:

MIDAS A) or MIDAS) [1]

is sufficient; however, the second call will cause MIDAS to execute an error return to JCL because of no output file specification. When JCL input is terminated, MIDAS is loaded and initialization begins. Before striking carriage return, make sure that all on-line devices are write enabled.

3.3 Initialization

When MIDAS comes on the air it must configure itself to its data acquisition time environment, i.e., extended core, types of mass storage devices, etc.; it must also start up certain hardware such as FPP, AIP, etc. The user will see MIDAS writing on various LDP volumes, initializing and positioning output files for subsequent data acquisition. One need not concern himself with this aspect of MIDAS execution except that initialization must be successfully completed before data acquisition can begin. The initialization has a number of error returns to JCL, which fall into two classes: hardware malfunction or improper output file specification. For such cases, MIDAS outputs an error message before returning to JCL (refer to 3.7 Teletype[2] Messages). Do not touch on-line devices, i.e., do not switch to LOCAL, do not press DISK LOCKOUT, etc; such actions will easily cause program failure. During initialization, MIDAS asks the user three questions that pertain to sense switch option, BURST MODE, and PRIMARY LABEL.

The sense switch option enables sense switches Ø-3 by substituting them for sense lines 1Ø-13 respectively. Any response other than Y and a terminator indicates to MIDAS that the sense switch option has not been chosen (refer to 3.5 Instrument Setup).

To enable BURST MODE, respond via the keyboard with Y and a terminator; any other response is non-BURST MODE. BURST MODE means that MIDAS will forego certain options in exchange for higher throughput rates (refer to 3.6, BURST MODE).

---

[1]) = carriage return.

[2] Teletype is a registered trademark of the Teletype Corporation.

Every logical MIDAS file must have an associated PRIMARY LABEL; this label is related to all the output files specified at JCL time, i.e., all the secondaries as well as the primary. The PRIMARY LABEL is the only internal link that unites and is common to all the LDP output files which comprise the logical MIDAS file; it is critical to MSORT (refer to 4.0 MSORT). The response to PRIMARY LABEL may be a string of decimal digits, meant to be input in pairs and separated by spaces or commas. The maximum number of digits allowed is 12, the maximum number of characters allowed, including spaces, commas and terminator, is 18; spaces and commas can be typed at any time; non-numeric input, except for space, comma, and terminator, is illegal and will cause the PRIMARY LABEL question to be reasked. The PRIMARY LABEL allows for but is not restricted to the following numeric identification format:

x_1x_2	x_3x_4	x_5x_6	x_7x_8	x_9x_{10}	$x_{11}x_{12}$
MO	DAY	YR	HR	MIN	SEC

If the full 12 digits are not input, then zero is implied for the unspecified digits; in particular, just a terminator will create a PRIMARY LABEL of zeroes.

NOTE

All LDP files, i.e., non-MIDAS files, have a PRIMARY LABEL which is commonly all zeroes; and logical MIDAS files created under different MIDAS runs but whose PRIMARY LABELS are identical may cause difficulties for MSORT.

A good practice is to create PRIMARY LABELS that are unique to a logical MIDAS file; as this rule is practically unenforceable by MIDAS, and violation of it may create insoluble or highly unpleasant SORT problems, caution is advised.

If initialization is successfully completed, MIDAS will output the binary version number message and wait for user command input (refer to 3.4 KEYBOARD COMMANDS). Prior to the version number message, the AIPOS CTRL/C[1] is active so that a return to JCL may be executed from the keyboard; subsequent to the version number message, CTRL/C will be inhibited, i.e., data acquisition time MIDAS cannot be aborted. The binary version number is primarily for maintenance purposes. Revisions to MIDAS by DEC and subsequent releases to the field will cause an update of the version number. The initial binary version number is ØØ.

[1]To type CTRL/C, hold down the CTRL key and type the letter C. The character ↑C will be printed on the Teletype.

3.4 Keyboard Commands

MIDAS is always in one of two modes: command input or setup. In command input mode, MIDAS waits for a string of keyboard characters which it interprets as a command. During data acquisition time, all of the following characters terminate a string of keyboard input:

- the set of standard LDP terminators (line feed, carriage return, and ALT MODE or ESCape)
- all control characters, e.g., ↑A, ↑B, ↑C

When MIDAS is in command input mode, all of the above terminators are illegal except carriage return. If an input string runs over 15 characters, it is terminated and treated as illegal. More generally, MIDAS will output an error message and ignore any command that it considers illegal; errors in syntax and nonsense commands fall into this class. There are eight legal keyboard commands: EXIT, PAUSE, RELAYS, RESET, SETUP, START, STOP, and TEST. The general format of a command is:

command;n)

where n is a run number previously assigned by MIDAS (refer to 3.5 INSTRUMENT SETUP). Trailing spaces in a command are legal. More than six characters to the left of the semicolon or to the left of the carriage return when there is no semicolon are always illegal. Not all commands require an argument; however, the semicolon is required as a delimiter for those that do. If an argument is supplied for a command that does not require one, the command is executed and the argument ignored. Similarly, MIDAS may output an error message for the case of a required and missing argument.

3.4.1 EXIT

The format of this command is

EXIT

The EXIT command is the last one given to MIDAS before return to JCL. The command requires no arguments and its function is to save all acquired data and return to JCL. EXIT terminates MIDAS, however, the command is illegal, ignored, and causes an error message if it is requested when any run or instrument is on line. If an instrument

has been set up but has not been terminated, i.e., it is either inputting data or may start inputting data to MIDAS, then that instrument is said to be on line. More simply, MIDAS inhibits aborting a run or a number of runs via EXIT.

3.4.2 PAUSE

The format of the PAUSE command is:

PAUSE;n↵

where n is a run number. PAUSE relates to an instrument or run number input to the right of the semicolon. A run number may be thought of as a name that MIDAS associates to an instrument (e.g., a gas chromatograph) that is external but defined to its environment via setup and user interface. PAUSE to MIDAS means shut down the AIP channel that is connected to the requested instrument but do not take the channel off line, i.e., temporarily stop data acquisition. Any instrument that is on line is either paused or started. Instruments that are paused may be restarted and vice versa; the conditions are antagonistic to one another. PAUSE has special implications for analog controlled instruments (refer to 3.5.7 X;Y AXES QUESTION). For a given instrument, the PAUSE condition does not have to be keyboard defined, e.g., a sense line could control PAUSE.

If n is not a run number, MIDAS outputs an error message and ignores the command. If the run number is not on line or if it is on line but its PAUSE condition is not keyboard defined, then MIDAS ignores the command but does not output an error message. The command is also ignored if the requested run is in PAUSE. For the case that the requested run number is on line, started, and has a keyboard defined PAUSE, MIDAS pauses the requested run and outputs the pause message to the Teletype. A run may be PAUSED as often as desired.

3.4.3 RESET

The format of this command is

RESET↵

Like EXIT, RESET requires no arguments and is illegal when any run number is on line. RESET allows the maximum number of distinct runs

to be expanded from 21 to 336, i.e., with the aid of RESET 336 different experiments can be conducted in one MIDAS run (one call from JCL). The maximum number of setups that may be specified before having to RESET is 21; the maximum number of RESETS is 16, of which 15 may be requested because there must be at least one left for EXIT (16·21=336). MIDAS is responsible to MSORT for retaining certain information about its various runs. This sort control information is saved at EXIT time with the "real" user data at the front of the primary MIDAS file[1]. If 21 runs are defined without RESET, then the area in the program reserved for sort control information is said to be saturated and MIDAS will inhibit setup until a RESET command is given. RESET may be requested before 21 runs but MIDAS will allocate space for the full 21. At a given point in time, the number of runs left can be computed from the formula:

$$NLEFT = 336 - [(NRESET)*(21)+NSETUP]$$

where NRESET = the number of RESET commands that MIDAS executed
NSETUP = the number of setups that the user has completed since the last RESET.

If the RESET command is illegal, MIDAS will output an error message and ignore the command; otherwise, MIDAS executes the command and outputs no message. One may think of RESET as a task that MIDAS must perform for MSORT, so that more than 21 runs can be executed.

3.4.4 RELAY

The format for this command is

RELAY ⏎

The RELAY command directs MIDAS to open all relays. It requires no arguments and is illegal if any instrument is on line, in which case MIDAS outputs INPUTERR<?> (Refer to Teletype Messages).

3.4.5 SETUP

The format for this command is:

---

[1]Sort control information is the reason why primary MIDAS files always have so many header blocks.

SETUP↵

SETUP also requires no arguments and is used to define a new run to MIDAS. SETUP is illegal if eight instruments are on line, if RESET is required, or if data acquisition has been terminated for any reason (refer to 3.7 Teletype Messages). In order for an instrument to get on line its parameters must be specified via a SETUP; the SETUP command switches MIDAS from command input mode to SETUP mode for the purpose of defining a new run. SETUP is legal while other instruments are on line, i.e., data acquisition may be ongoing during a setup.

3.4.6 START

START requires a run number argument, i.e.,

START;n ↵

START is analogous to PAUSE in all respects and the action that MIDAS takes upon a START command is parallel to its behavior under the PAUSE command, e.g., a START request for a run that is started is ignored (refer to 3.4.2, PAUSE). START means to enable the AIP channel associated to run number n, i.e., start acquiring data on it. Note that MIDAS makes a strong distinction between acquired data and throughput data (refer to 3.4.8 TEST).

3.4.7 STOP

The format of this command is

STOP;n ↵

STOP requires a run number argument and it is a request for MIDAS to take the given instrument off line. Once an instrument is STOPped it cannot be restarted; that particular run is finished. In order to rerun a STOPped instrument, it must be setup all over again (in which case MIDAS will assign the instrument a new run number). STOP only applies to run numbers that are on line. Unlike START and PAUSE, STOP is always keyboard defined, e.g., if

STOP;27

is typed and run number 27 is on line, then run 27 is terminated, regardless of the STOP condition specified at setup time.

3.4.8 TEST

The format of the command is

TEST;n↵ or TEST↵

TEST without an argument means terminate test mode for the current instrument. TEST with a run number argument means terminate test mode for the current instrument and initiate test mode for the requested instrument. If MIDAS detects an illegal run number, it ignores the command and outputs an error message. If no instrument is in test mode, then MIDAS ignored TEST↵.

TEST gives a scope image of data coming in from the instrument, i.e., TEST mode activates the VR12 display and is the only visual aid that MIDAS offers its users. TEST mode can be requested for any one-line instrument at any time but only one run can be viewed at a time, i.e. concurrent display of a number of runs is not allowed. START and PAUSE are transparent to TEST mode so that an instrument which is PAUSEd prior to entering TEST mode remains PAUSEd. An instrument may be STARTed and PAUSEd during TEST just as during non-TEST or throughput mode. Data that is sent out from an instrument and received by MIDAS through the AIP channel tied to the instrument is said to have been acquired by MIDAS. Acquired data that MIDAS buffers out to some LDP mass storage volume for permanent record is said to have been throughput. MIDAS never acquires data from an instrument which is PAUSEd; data is acquired only from a started instrument according to conventions laid down for that instrument at SETUP time. MIDAS throughputs the data only if the instrument is not in TEST mode, i.e. MIDAS will not display and throughput data from a given instrument but it will display data from one instrument and throughput data for others. If an instrument enters TEST mode when it is PAUSEd, MIDAS displays the data from the previous test. Once the instrument is started, the incoming data is displayed as a moving waveform across the scope. The motion may be either forward or backward, depending upon the data rate of the instrument; and for those cases where the data rate is harmonic to the display frequency, the waveform will be stationary. SETUP may be requested (command input is active) during TEST mode; however, MIDAS automatically aborts TEST mode. If TEST

mode is requested for an off line run, the display is activated but that is all.

3.5 Instrument Setup

Instruments that are to come on line must go through a SETUP mode which is a series of questions and answers via the keyboard and the scope describing use of relays, data rate, etc. During SETUP MIDAS asks seven questions; successful completion of all seven puts an instrument on line. If a response is terminated with carriage return, MIDAS interprets and executes that response, but if a response is terminated by any other character, including the input of too many characters, MIDAS ignores the response and reasks the previous question (line feed is suggested for back up). If one tries to back up on the first SETUP question, MIDAS terminates SETUP and enters command input mode. If a response to a question is invalid, MIDAS outputs a message and reasks the question; if the response is valid, MIDAS proceeds to the next question. Upon successful completion of the last question, MIDAS re-enters the command input mode. Note that command input is inactive during SETUP mode and that the input string

RESET)

for example, would be interpreted as a response to a SETUP question.

3.5.1 RUN NUMBER Question

The run number message is the only SETUP question that is output to the Teletype; the other six are output to the scope. User run numbers are assigned by MIDAS; they start at 20 and run sequentially up to 336. Run numbers Ø-19 have been reserved for internal use. The format of the message is:

RUN NUMBER n :

where $20 \leq N \leq 336$.

MIDAS never assigns the same run number twice in one given call from JCL. The number is very important because communication to either MIDAS or MSORT about a particular experiment or instrument subsequent to SETUP must be done via the run number assigned by MIDAS, e.g.,

TEST;n)

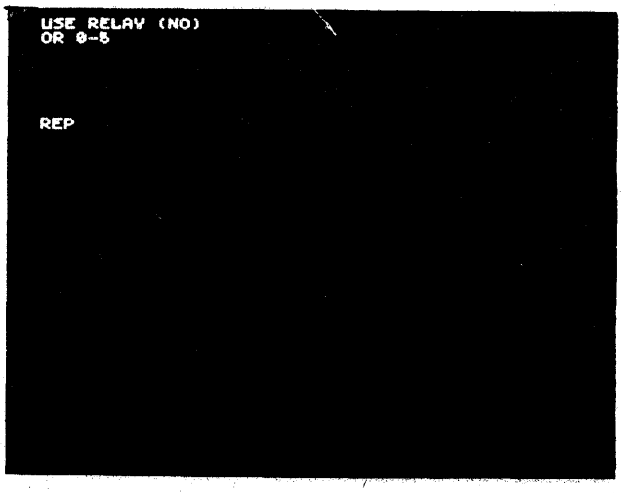
MIDAS outputs the run number to the Teletype so that a printed record is available.

Following the printout of the run number, type in a name under which the run will be saved. Since one will eventually want to break a MIDAS file down by run numbers, i.e., sort and save gas chromatograph data as one file, UV data as another etc., the sorted data can be named now. The name can be any string of characters that is less than 7 followed by a carriage return. If more than six but less than 15 characters are given for a name, MIDAS outputs an error message and reasks for name. Neither MIDAS nor MSORT checks the name for legality (where legal means conformable to JCL ground rules). If, during SETUP, an illegal name or none at all is typed, i.e., just a carriage return, JCL does not accept the name and requests that the file be renamed. If duplicate file names are assigned to different run numbers and are subsequently sorted onto the same LDP volume by MSORT, then the SORT user will get an error message from JCL. The important point of the first SETUP message is to be reasonable and consistent in naming conventions and not to be forgetful of run number assignment.

3.5.2 RELAY Question

The PDP-12 has six relays numbered Ø-5 and clearly visible on the front panel.

These relays can be used to open and close circuits in the MIDAS environment. The RELAY question allows relays to be defined for a run. The format of the question is:



where REP is an abbreviation for REPLY. All input for the remainder of SETUP is echoed on the Teletype and displayed opposite the word REP on the scope. All questions are displayed in the top left of the scope face. The format of the reply to RELAY is

$$\underbrace{x_1 \cdots x_n}_{\text{(OPEN)}} ; \underbrace{y_1 \cdots y_m}_{\text{(CLOSE)}}$$

where $x_1 y_1$ are digits in the range Ø-5 and $Ø \leq n+m \leq 12$. Simply striking carriage return selects the default option in parentheses, i.e., NO means do not use the relays.

Whenever MIDAS encounters the START condition for the run, it opens relays $x_1 \ldots x_n$ and closes relays $y_1 \ldots y_m$. For a STOP or a PAUSE condition, MIDAS reverses the state of the relays, i.e., closes relays $x_1 \ldots x_n$ and opens relays $y_1 \ldots y_m$. In order to have a combination of relays opened on START and closed only on STOP or PAUSE, simply omit the y_1 by terminating after the x_1, e.g.,

Ø4

means open relays Ø and 4 on a START condition and close them on a STOP or PAUSE condition. Similarly, to enable only the y_1 class, omit the x_1 by inputting the semicolon first, e.g.,

;3

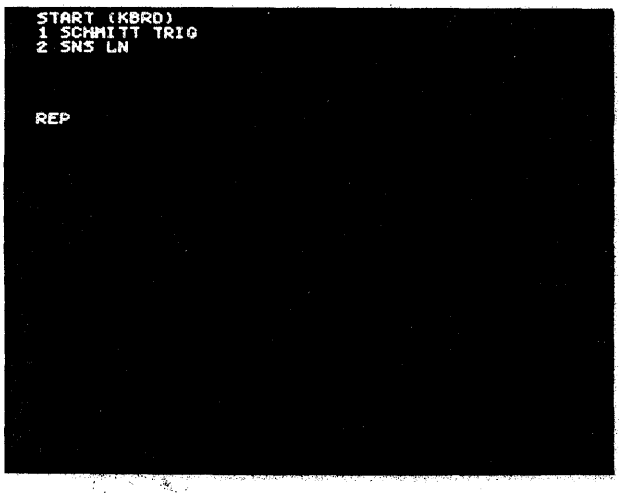
means close relay 3 on a START condition and open it on a PAUSE or STOP condition. A relay specified both to the right and to the left of the semicolon is closed upon START and remains closed throughout the run. The relays which are closed on a STOP condition remain closed but there is a keyboard command to open all the relays (refer to 3.4.4 RELAYS).

MIDAS allows instruments to share relays; therefore, it is possible for a START condition on instrument A to close a relay that is a associated to both instruments A and B. The idea of sharing opens the door for instrument synchronization and interaction and avoids the problem of having to divide three Schmitt Triggers among eight instruments. However, the burden of maintaining instrument independence is upon the user.

3.5.3 START Question

MIDAS asks the START question to determine what event will trigger the START of data acquisition; the program must know when to begin sampling a given AIP channel.

MIDAS considers START as a necessary but not a sufficient condition for through putting data, e.g., if an instrument is "started" when it is under TEST, MIDAS will not throughput any data. Consider the notion of START for externally synched instruments (refer to 3.5.7 X;Y AXES Message). MIDAS does not tell the AIP hardware to acknowledge the instrument to which it is connected until the start event occurs; in this sense the START condition is necessary. Prior to start the lines of communication between the instrument and the AIP are closed logically by MIDAS and one may consider this effect equivalent to the instrument physically not interfaced to the AIP. Once MIDAS receives a START pulse from an externally synched instrument it opens lines of communication; however, it may not receive any data simply because the instrument may not send any signals into its associated AIP channel. The format of the START question is:



The format of the reply is

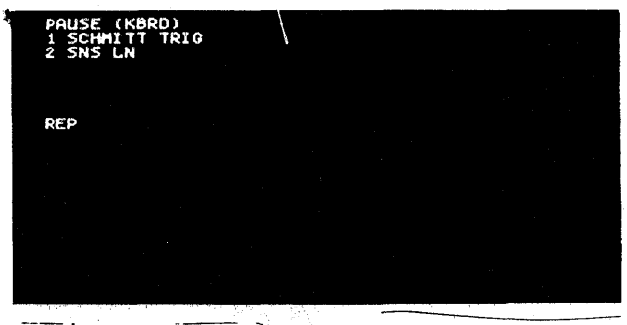
x;y)

If carriage return is typed, MIDAS selects the option in parentheses, the keyboard, as the START condition for the run. Otherwise x is either a 1 for option 1 (Schmitt trigger) or a 2 for option 2 (sense line), or x is illegal input; the y is an octal number which is appropriate to the specified option. The KW12A clock has three Schmitt triggers which are phone jack inputs vertically positioned on the left front face of the PDP-12 and are respectively numbered from top to bottom as inputs 1-3. If option 1 is specified, then y must be 1-3 to specify the appropriate Schmitt trigger. MIDAS interprets no y argument or y=Ø as Schmitt trigger 1. Once the instrument is on line, it need only fire the Schmitt trigger to have MIDAS acknowledge the START condition. If the sense line option is selected, the y argument must be an octal number in the range Ø-14; MIDAS assumes Ø if y is omitted. There are 13 decimal sense lines which may be accessed via connector slot N13 of the PDP12 CPU. Refer to print D-BS-EP12-Ø-ICB for the sense line pin definition.

Sense lines are either grounded (ØV) or floating (+3V) and can be switched back and forth between these two states or maintained indefinitely in either, depending upon the behavior of the interface that controls them. MIDAS interprets a sense line at ØV as START condition met and 3V as START condition not met; therefore, an instrument which is to start off a sense line must maintain that line at 3V and ground it (ØV) to start. If the sense switch option is enabled, then 1Ø-13 for y means sense switches Ø-3. Users may think of sense switches which are raised to be at ØV and those which are lowered to be at 3V i.e., to initiate start from a sense switch, simply raise it.

3.5.4 PAUSE Question

MIDAS requires that an instrument have a PAUSE condition where PAUSE means to temporarily refrain from data acquisition; from a MIDAS point of view, PAUSE means that the instrument is on line but resting. The format of the question is:



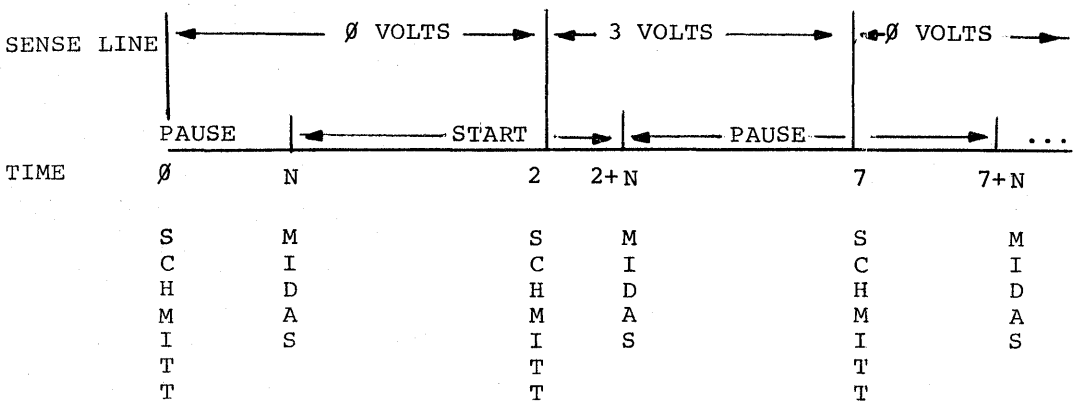
The format of the answer is

x;y)

where x and y are interpreted identically as in the START question. The manner in which MIDAS dispositions a Schmitt trigger for PAUSE is identical to that of a Schmitt trigger START; sense line PAUSE is treated in a reverse manner to sense line START. A sense line at +3V means PAUSE condition met and ØV means not met.

MIDAS allows specification of the same Schmitt trigger or sense line for both START and PAUSE; however, MIDAS requires that the interface to the Schmitt trigger or sense line demonstrate specific properties for correct functioning of the run. MIDAS response times for START, PAUSE, and STOP conditions may vary from 3 milliseconds to less than 1 second, where response time means the elapsed time between the moment that the instrument initiates a condition, e.g., grounds a sense line, and the moment when MIDAS realizes that the sense line has been grounded. Normal response time for MIDAS will be 5Ø milliseconds or less but can grow into the hundreds of milliseconds range for those cases where MIDAS is extremely busy, say six or more instruments with a throughput rate of 3000 Hertz or more. Between response times, MIDAS is not cognizant of events occurring on a given Schmitt trigger or sense line and at the time of response MIDAS dispositions only the last event that occurred on a given Schmitt trigger or sense line. From a MIDAS point of view, a Schmitt trigger that is tripped 19 times between responses is equivalent to a Schmitt trigger that is tripped only once between responses; and a sense line that goes from 3V to ØV and back to 3V between responses is equivalent to a sense line that was maintained at 3V between responses. Suppose the operator wished to throughput data for a 2 second interval followed by a 5 second pause then 2 more seconds of data, etc. all of which is controlled by a given Schmitt trigger. At time Ø the instrument trips the Schmitt trigger, n milliseconds later MIDAS sees the START condition and data throughput begins.

At time 2 seconds the instrument again trips the Schmitt trigger and within n milliseconds, MIDAS acknowledges the condition and the instrument pauses. At time 7 seconds, the instrument again trips the Schmitt trigger and throughput begins again, and so on. For a sense line application to the above example, the instrument grounds the sense line to ØV at time Ø and maintains it at ØV up until time 2 seconds; MIDAS acknowledges the sense line at time n milliseconds and begins throughput. At time 2 seconds, the instrument raises the sense line to 3V and maintains it at 3V until time seven seconds in order to sustain a five second pause.



Suppose that an experiment is to start on Schmitt trigger A and pause on Schmitt trigger B A≠B. For this case, the instrument would fire Schmitt trigger A at time intervale Ø, 7, 14, ..., and B at intervals 2, 9, 16, ... If a sense line is used in conjunction with another sense line or Schmitt trigger that sense line must not only be appropriately ENABLEd but it must also be disabled for one full MIDAS response interval prior to ENABLing its companion condition. For the above example, suppose that the run were to start on sense line A and pause on Schmitt trigger B. At time Ø the instrument grounds sense line A, and at time Ø+N throughput starts. Between the time Ø+N and 2-N, the instrument must raise sense line A to 3V, i.e., before firing Schmitt trigger B. Sense line A would remain at 3V until time 7. More generally, MIDAS will look for a pause condition only on an instrument that is started and for a start condition only on an instrument that is paused; but once an instrument satisfies a given condition MIDAS immediately begins looking for the opposite condition. In the example of sense line A, start and Schmitt trigger B pause, suppose that sense line A remained at ØV for the entire run. Every seven seconds MIDAS would acknowledge Schmitt trigger B, pause the run, and at a time no later than n more

milliseconds would restart because sense line A was at ØV. The above example is not meant to imply that START and PAUSE events occur at periodic intervals; they can be totally asynchronous. The important point to remember is that for a given MIDAS response time interval an instrument should fire a Schmitt trigger only once and that a voltage presented to a sense line for the purpose of either enabling or disabling a given condition must be maintained across the entire interval. The interface to sense lines and Schmitt triggers should be in a manner that is consistent with the behavior of the instrument and other instruments that may be in the MIDAS environment. Sense lines and Schmitt triggers may be shared within and across instruments; it is possible, therefore, to set up eight runs that all start and pause on the same Schmitt trigger. It may be instructive to set up an instrument specifying a set of relays, Schmitt trigger one for START, and Schmitt trigger two for PAUSE.

No instrument or interface is necessary. Upon completion of SETUP, raise both Schmitt triggers to line frequency (the knob to the immediate right of the phone jack) and observe the relay lights and Teletype. MIDAS does what it is told. START and PAUSE are intimately related to MIDAS response time which, unfortunately, can be highly variable and is dependent upon a number of factors in the MIDAS environment, e.g., number of on line instruments, throughput rate, types of instruments, etc. For any given environment the response time is fixed. The average response time is from 3 milliseconds to less than a second and is usually 1ØØ milliseconds or less.

3.5.5 STOP Question

In addition to START and PAUSE, MIDAS requires a STOP condition; some event which directs MIDAS to take the instrument off line. The format of the question is:

The format of the reply is

x;y)

x = options 1-4 and y = numeric argument for x.

Schmitt trigger and sense line STOP options are identical to those of PAUSE and the default response (striking RETURN) is the keyboard. If a given sense line or Schmitt trigger is defined as either the START or PAUSE condition for the run, then specifying the same sense line or Schmitt trigger for STOP is illegal; however, the STOP condition may be the same sense line or Schmitt trigger that is a START or a PAUSE for another instrument.

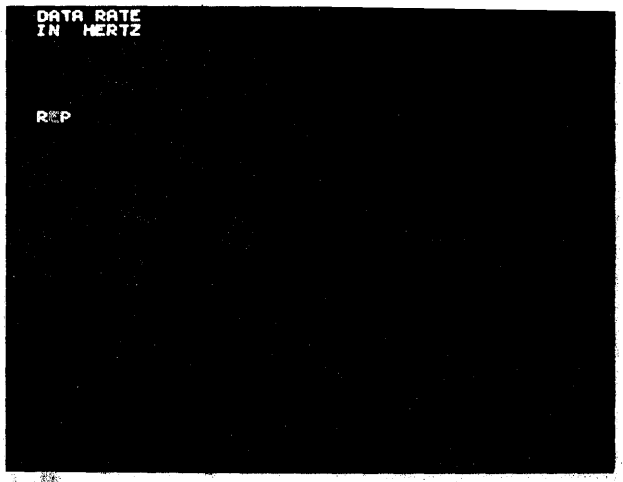
Option 3 is STOP ON POINTS which means that MIDAS will terminate the run after a specified number of points have been throughput. Data acquired under test mode does not count towards STOP ON POINTS. The argument y specifies the number of points where y is in the range 1-999,999. For a STOP ON POINTS condition MIDAS will not terminate on less than the requested number y but may terminate on more. An experiment set up to STOP after 1ØØØ points may actually stop on 1ØØ2, 11Ø9, or 1476, but never on 999.

Option 4 is STOP ON TIME where TIME is the number of seconds specified by y in the range 1-999,999. STOP ON TIME to MIDAS means only that time which elapses while the run is started and throughputting data. MIDAS does not charge the user for time elapsed while his instrument is PAUSEd or in TEST mode. TIME is accurate to within a second.

There are MIDAS defined STOPS which take precedence over user defined STOP conditions. If MIDAS runs out of mass storage, i.e., fills the entire logical MIDAS file with data or if MIDAS encounters a FAIL condition (refer to 3.7.4 *MFAIL*), it will abort all on line instruments and inhibit any further setting up. If a given instrument runs over rate MIDAS will take it off line (refer to 3.5.6 DATA RATE Question).

3.5.6 DATA RATE Question

The format of the DATA RATE question is:



The format of the reply is:

x)

where x is some number in Hertz.

MIDAS requires that every instrument have a data rate associated with it where data rate is points per second or Hertz. The context of data rate depends upon the instrument classification defined in the last question (refer to 3.5.7 X;Y AXES Question), but for all runs the response to the data rate question means a rate that the instrument cannot exceed while throughputting data. MIDAS constantly monitors instrument data rates to ensure that all instruments stay within the rate specified by the data rate question. MIDAS terminates any run which exceeds its data rate. A "run away" instrument can destroy data of other on line runs; therefore MIDAS monitoring attempts to prevent this situation.

Experiments for which a rate of x Hertz is specified at setup time and which run at x+y Hertz at throughput time will be taken off line by MIDAS. An instrument is allowed 7 Hertz over rate after that it is a strong candidate for termination. MIDAS makes the further restriction that the sum of rates across all instruments is not greater than the maximum allowable throughput rate.

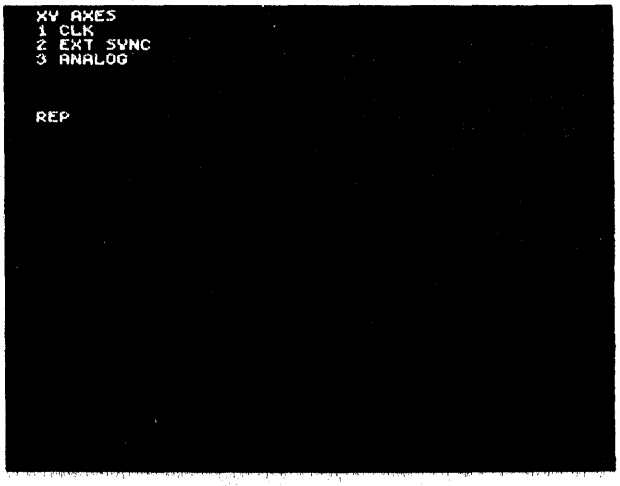
Suppose the maximum rate is 12ØØ Hertz and 1ØØØ Hertz have already been allocated to on-line runs. MIDAS considers all data rate responses greater than 2ØØ Hertz as illegal because it cannot handle

more than 12ØØ Hertz total. If a given instrument must run at 3ØØ Hertz, its setup must wait until enough runs go off line to free 1ØØ additional Hertz. MIDAS does not care if all on line runs are PAUSEd; they may restart at any time. MIDAS does not tell the user how many Hertz are unallocated; this must be established by trial and error. The maximum rates vary with hardware configuration and obey the following table (also see 3.6 BURST MODE).

Amount of Core	LINCtape	RK8 Disk
8K	1200	5700
12K	2700	6900
16K	3000	6900
20K	4000	6900
24K	4000	6900

3.5.7 X;Y AXES Question

In this final question the instrument classification and AIP channel are defined. The format of the question is:



The format of the response is

x;y)

where x is one of the above options and y is an AIP channel.

Option 1 (CLK) specifies those instruments whose sampling is KW12A clock controlled, that is, MIDAS will periodically sample the requested AIP channel (the y) at the rate implied by the user-response to the DATA RATE question. The clock runs at a basic frequency and sampling controlled by the clock must be a multiple of the basic frequency. Clock controlled instruments cannot run at arbitrary rates. The fastest rate for clock controlled or time based instruments is 500 Hertz. The other acceptable rates are those numbers which divide evenly into 500, i.e., 250, 125, 100, 50, 25, 20, 10, 5, 4, 2, and 1. MIDAS sets the data rate for a time based instrument to the largest basic rate that is less than or equal to the rate specified in the DATA RATE question. If a rate of 400 Hertz is specified for a clock controlled instrument, MIDAS sets the rate at 250 Hertz but subtracts 400 from the total unused Hertz in the system. Because time based instruments never run over rate, this sort of response gives 150 Hertz of unnecessary protection. For most efficient operation, specify the basic rates when setting up a clock controlled run.

If option 2 (EXT SYNC) is selected, the sampling of the specified AIP channel will be controlled by the instrument via an interface into the AIP external sync; i.e., whenever the instrument presents a sync to the AIP, the AIP automatically samples the channel (provided, of course, that MIDAS has acknowledged the start condition). This instrument classification is unique because MIDAS has no control over the sampling scheme and hence the data rate. Externally synched instruments are the only reason for MIDAS data rate protection. Because of the asynchronous structure of its environment, MIDAS makes little record of events that occur at some absolute point of time; rather the program acquires a record of specific events across a certain time interval. MIDAS periodically checks for occurrence of defined events in its environment, e.g., has instrument n raised sense line A, has the STOP ON POINTS condition been satisfied by instrument m, has instrument e exceeded its defined data rate by throughputting too many points in the last time interval, and so on. The MIDAS post hoc monitoring of events does not offer full data rate protection on externally synched instruments because AIP limits are 25,000 Hertz and the data rate protection routines execute at 7000 Hertz. If MIDAS recognizes this type of runaway condition, it simply shuts down all data acquisition (refer to 3.7.5 MFAIL) before good data is destroyed. Externally synched instruments provide a number of advantages. Data rate is not restricted to a multiple of a basic

frequency; it can be arbitrary. It is the only method by which an instrument can run over 500 Hertz. It provides an avenue for running arbitrarily slow, e.g., .001 Hertz. Because MIDAS has no sampling responsibility for externally synched instruments, it has time for other things. For instance, MIDAS response time to sense lines and Schmitt triggers will improve significantly in an environment of only externally synched instruments.

The third option is analog controlled instruments (ANALOG). This instrument class is halfway between external sync and clock controlled and exhibits properties of both. Sampling of the specified AIP channel is controlled by the PDP-12 A/D (not AIP). For analog controlled instruments MIDAS sets up a one to one correspondence between the PDP-12 A/D and the AIP A/D channels. For N=Ø, 7 PDP-12 A/D channel 1N is associated to AIP channel 4N, i.e., (1Ø,4Ø),(11,41),...(17,47). Selecting option 3 and not specifying an AIP channel in the range 40-47 is illegal. MIDAS samples analog controlled channels in the following manner. The appropriate PDP-12 A/D channel is periodically sampled and compared with the last registered PDP-12 analog reading. If the new reading is greater than the old (2 millivolt increment is sufficient) the new reading replaces the old reading for subsequent comparisons and the corresponding AIP channel is sampled and retained as data from the instrument.

Like external sync, analog instruments are asynchronously sampled and externally controlled, i.e., MIDAS samples the y axis (AIP A/D) only when the instrument increments the x axis (PDP-12 A/D). The only way to know that an analog channel has incremented is to periodically read it. The sampling of the PDP-12 A/D channels is done under control of the clock at a rate of 500 Hertz. Analog controlled instruments are similar to time based instruments in three respects:

- the clock is critical to their function;
- their maximum rate is 500 Hertz;
- they cannot run away because MIDAS controls their data rate.

If an analog controlled instrument runs faster than 500 Hertz, MIDAS will miss the data on the corresponding AIP channel and be unaware that it has lost the data. The data rate for the analog classification can be arbitrarily slow but it must not exceed the rate specified in the DATA RATE question. Because the PDP-12 A/D is 10 bits, the maximum number of increments (and therefore samples) across the full

range of values is 1024 and even less if the increments are not all 2MV (1 bit). To enable more than 1024 points for analog controlled instruments, the PAUSE condition resets the PDP-12 A/D internal reading to -1V so that another 1024 points can be taken on a restart. Note that it is the responsibility of the instrument also to reset its input to the PDP-12 A/D back to -1V or at least something less than 1V, otherwise a restart will produce an initial increment of 2V, one sample and no subsequent samples. Analog controlled instruments provide no data acquisition properties not available under external sync; however, the analog class offers two advantages: the interface into the PDP-12 A/D is less expensive than into the AIP external sync and from ease of interface some instruments more readily fall into the analog controlled class.

The y argument defines the AIP channel through which the instrument will input its data. If the incoming data is analog, y must be in the range 40-57. If the incoming data is either 18 bit binary or 8 digit BCD, then y must be 20-27. For digital channels a knob on the AIP hardware selects one of the two above digital formats. MIDAS does not take note of which one is selected. From a MIDAS point of view one may effectively double the data rate across a digital channel by switching from 18 bit binary to 8 digit BCD. The primary reason for this effect is that the AIP transfer format for 18 bit binary is 2 12-bit words per datum but for 8 digit BCD it is 4 12-bit words per datum. Those users who are running 8 digit BCD input at a rate of n Hertz are really adding a load of 2n to aggregate throughputs but MIDAS will only be aware of an additional n; note that the 8 digit BCD format comprises data rate protection. Furthermore, MSORT will only retain the high order five digits of the conversion; MIDAS gives minimal support for BCD to binary format. AIP channels cannot be shared; one to a run number only. If an AIP channel is requested that is already on line, MIDAS will produce an error message and reask the X;Y AXES question.

3.6 BURST MODE

BURST MODE is an option that may be selected at initialization time, effectively exchanging certain MIDAS features for increased throughput rates. The following table describes the approximate BURST MODE limits for various hardware configurations.

Amount of Core	LINC TAPE	TAPE INCREASE	RK8 DISK	DISK INCREASE
8K	2500	1300	7400	1700
12K	4000	1300	10000	3100
16K	4000	1000	13000	6100
10K	4000	0	15000	8100
24K	4000	0	15000	8100

The INCREASE columns show the improvement in throughput for BURST MODE over non-BURST MODE. However, selecting BURST MODE means there is:

- No data rate protection
- Externally synced instruments only
- No STOP ON POINTS or STOP ON TIME

In BURST MODE MIDAS does not display options 3 and 4 in the STOP question and inhibits the data rate question entirely. MIDAS displays options 1 and 3 of the X;Y AXES question but they are illegal just the same. In BURST MODE MIDAS offers MFAIL protection only (refer to MFAIL protection). All other aspects of MIDAS are the same as in non-BURST MODE.

3.7 Teletype Messages

MIDAS outputs Teletype messages whenever an important event occurs. Messages are output in a chronological manner, i.e., if event A occurs before event B then the message appropriate to A is typed before the one for B. MIDAS considers the Teletype as a low echelon task and types messages when it has time. Teletype output and keyboard input are independent of one another. It is possible for MIDAS to output a Teletype message right in the middle of inputting a string of characters. Input on top of messages is accepted by MIDAS.
In order to get neat output to the Teletype, one should wait for MIDAS to ask the run number question (refer to 3.5.1) before answering it.

Because events associated to Teletype messages can occur at a rate faster than MIDAS can output them and because MIDAS does not backlog a great many of them, it is possible for MIDAS to "forget" a Teletype message. MIDAS will backlog a maximum of 30 messages after which it will inhibit Teletype messages until the number of

untyped messages is less than 30. This backlog is more than enough for normal MIDAS usage. In the event of an overloaded Teletype, the event related to a given message will always be dispositioned, e.g., MIDAS will not ignore the sense lines just because the Teletype is overloaded.

Those messages which occur at a time when an operator may be expected to be at the keyboard usually contain neither asterisk nor bell characters (↑G). Those messages which may occur when an operator may not be at the keyboard but perhaps near it, i.e., somewhere in the laboratory, have the bell character in them. The asterisks shown in the messages below stand for two characters (the bell and the asterisk).

3.7.1 BURST MODE ?)
TYPE Y FOR YES)

MIDAS is asking for the BURST MODE option and waiting for a reply. Anything but Y) is a no. This message is output at initialization time.

3.7.2 I AM BUSY???)

This message is output during SETUP mode for one of the following reasons:

- The twenty-second SETUP has been attempted without a RESET.
- SETUP requested with eight instruments on line.
- An AIP channel that is already on line was specified in the X;Y AXES question.

3.7.3 INPUT ERR <?>)

This message occurs in both command input and SETUP mode in a wide variety of cases most of which will be syntax or typing errors. The input error message may occur for specifying incorrect numeric arguments or numeric arguments that are out of range or missing. Terminating a command with a character other than carriage return will also cause the error message. Other sources of the input error messages are listed below.

3.7.3.1 TEST

An undefined run number was specified.

3.7.3.2 SETUP

SETUP was requested after MIDAS terminated data acquisition because of run time failure or no more mass storage.

3.7.3.3 RESET, EXIT, RELAYS

One of the above commands was requested with runs on line.

3.7.3.4 RUN NUMBER QUESTION

More than six characters were specified for name.

3.7.3.5 STOP Question

The same Schmitt trigger or sense line was requested for either STOP and START or STOP and PAUSE.

3.7.3.6 DATA RATE Question

A rate was specified that would extend MIDAS beyond maximum allowable throughput rates.

3.7.3.7 X;Y AXES Question

The analog controlled instrument class was requested for an AIP channel other than 40-47; or either the analog or clock controlled instrument class was requested in BURST mode.

3.7.4 *LOST DT n)

The AIP hardware has failed to sample the AIP channel associated to run number n ; which means that MIDAS has failed to receive data from the instrument and that data is said to be lost. The most common reason for lost data is an externally synched instrument that is supplying syncs to the AIP hardware at too fast a rate. If the message is persistent the run is terminated by MIDAS data rate protection. Sustained lost data messages indicate that interface to

the AIP external sync is incorrect or malfunctioning or that the AIP hardware is failing or perhaps both. If the message is sporadic it means that transient syncs are occurring most likely from either a poor connection or noise on the "sync" line. A lost data message should never occur on clock or analog controlled instruments; however, a "runaway" AIP sync is capable of generating lost data for all other on-line runs and thus is capable of destroying the integrity of their incoming waveforms. A series of lost data messages occurring across a number of runs indicates that something fatal has occurred to either the AIP hardware or MIDAS. An EXIT should be performed immediately, or one runs the risk of losing all throughput data. If data is lost for a given run, the only action that MIDAS takes is to output the lost data message; the run remains on line.

3.7.5 *MFAIL* n↵

The main reason for MFAIL is to safeguard against hardware malfunction. In a given MIDAS environment there is a class of events which disrupt the primary MIDAS task of throughputting data. When a malfunction occurs, MIDAS shuts down data acquisition immediately and entirely in an effort to retain any good data that it may have. Since MIDAS data is undefined to the LDP world until it is saved via the EXIT command, it is advisable to do a quick EXIT subsequent to the MFAIL message. If any runs are on line at the end of data acquisition, the *MFAIL* message will be preceded by a series of STOP messages. Be sure to correct or remove the source of MFAIL before reloading MIDAS. The n in the MFAIL message indicates the type of failure of which there are four.

3.7.5.1 *MFAIL* ØØØ↵ or *MFAIL*ØØ1↵

MIDAS has failed to throughput data to mass storage; the numbers Ø and 1 indicate the classification of the failure. Both may be considered as I/O failures. The failure may occur if an externally synched instrument exceeds its data rate and pushes MIDAS beyond its maximum allowable throughput limit. Marginal LINCtape or disk hardware may cause the same failure; scratched LINCtapes or RK8 disks, improperly marked tapes or tapes with bad blocks also cause this type failure. Excessive lost data on the AIP, KW12A clock or CPU failure are still other reasons for this failure.

3.7.5.2 *MFAIL*ØØ2)

This failure is the result of the MIDAS data rate protection routines failing to execute in a given time interval. FPP malfunction can generate this type of failure but the most common reason is a "run away" instrument where run away means a data rate exceeding 7000 Hertz.

3.7.5.3 *MFAIL* ØØ3)

This message is caused by FPP malfunction. At the end of data acquisition, the MFAIL message will be preceded by a series of STOP messages. If an EXIT is performed subsequent to MFAIL, most and usually all of the data is recoverable. The philosophy of MFAIL assumes that a failure leaves the central processor and a minimal portion of the software (not excluding the LDP Monitor) able to perform an EXIT. MIDAS offers no aid for massive CPU failure, the results of which are totally unpredictable. If the keyboard goes dead at data acquisition time, or if MIDAS ever halts, all is lost.

3.7.6 *MIDAS* n)

This message appears at two different times with two different n arguments. The message first occurs (no bells in this one) at the end of MIDAS initialization and signals that MIDAS has entered data acquisition mode and is waiting for command input. The n indicates the binary version of MIDAS that was loaded by JCL. The initial version is ØØ, modifications of the binary will produce updates of the version number.

The second version of the message occurs when MIDAS terminates data acquisition because the logical MIDAS file is full. The argument is the MIDAS internal run number and indicates the current data acquisition mode of MIDAS; there are two: ØØ1 is non-BURST mode and ØØ2 is BURST mode. If MIDAS happened to be in SETUP mode at the time this message is output, MIDAS reverts automatically to command input mode upon termination of the current question, i.e., MIDAS aborts the SETUP. If any runs are on line at end of data acquisition, the *MIDAS* message will be preceded by a series of STOP messages.

3.7.7 NO AIP

This is an initialization time message which indicates that the AIP hardware is off line, nonexistent, or malfunctioning. Upon completion of the message MIDAS returns to JCL.

3.7.8 NO CLOCK

This message also occurs at initialization time and means that MIDAS has found no KW12A clock or that the clock is not working properly. The check for no clock is bypassed if BURST mode was required. MIDAS returns to JCL subsequent to the NO CLOCK message.

3.7.9 NO FPP

This message means that the FPP is malfunctioning or nonexistent and causes MIDAS to exit to JCL.

3.7.10 *OVER RT n

MIDAS outputs this message when run n is aborted for exceeding the data rate specified for it at SETUP time; i.e., the instrument is running too fast and has forced MIDAS to take it off line.

3.7.11 *PAUSE n

Whenever MIDAS acknowledges the PAUSE condition for instrument n it outputs the above PAUSE message.

3.7.12 PRIMARY LABEL :

MIDAS outputs this message at initialization time and waits for the primary label for the logical MIDAS file. The response is a string of up to 12 numeric characters delimited by either comma or space (refer to 3.3 INITIALIZATION).

3.7.13 RUN NUMBER n:

This is the first SETUP question and the only one that MIDAS outputs to the Teletype (refer to 3.5.1 RUN NUMBER Question).

3.7.14 SSW OPTION?)
TYPE Y FOR YES)

This message is output at initialization time. If Y is typed, MIDAS will substitute sense switches Ø - 3 for sense lines 1Ø - 13.

3.7.15 *START n)

MIDAS types this message whenever it acknowledges the START condition for RUN NUMBER n (refer to 3.5.3 START Question).

3.7.16 *STOP n)

This message is output by MIDAS when instrument n satisfies its STOP condition or when MIDAS terminates all data acquisition.

3.7.17 TEST INST n)

This is the MIDAS response to the command TEST;n and signifies that MIDAS has just put instrument n into TEST mode.

3.7.18 WA TOO SMALL n)

MIDAS types this error message when an output file specified at MIDAS JCL time is too small for MIDAS output. Program returns to JCL. The argument n is in the range Ø - 7 and indicates which output file was too small. The first output file specified to JCL is Ø, the second is 1, etc.

4.0 MSORT

4.1 General Description

MSORT is a utility program like the File Handling functions whose purpose is to convert from one data format to another. A MIDAS generated file contains data in a format that is not interpretable by standard AIPOS software, e.g., DORA and FORCON file handlers; therefore, that data is inaccessible to the user. MSORT converts MIDAS data from a given run into a standard AIPOS three-word floating point file. The output of MIDAS is input for MSORT and the output of MSORT can be input for a number of AIPOS programs. MSORT converts only MIDAS files. It allows creation of a maximum of eight files in one MSORT run and specification of up to eight sorts in one command. MSORT supports up to 2ØK of core.

4.2 JCL Time

The general format of the JCL command is:

MSORT dev:outØ,dev:outl,..,dev:outn=dev:PRIME,dev:filea,...,dev:fileb

MSORT does not require output to run; however, the SORT command (refer to 4.4.3 SORT) will be illegal if no output is specified. The output file names are completely arbitrary and are ignored by MSORT. The primary MIDAS file must be specified for input and that file should be the first MSORT input file specified[1]. Input subsequent to the primary should be those MIDAS secondaries that define the rest of the logical MIDAS file. The order in which the secondaries are specified to JCL is immaterial and MSORT does not require that all secondaries be specified, but if they are not then certain SORTs may be either inhibited or aborted prematurely.

4.3 Initialization

Like MIDAS, MSORT must perform certain initialization tasks before it will accept command input. One option is available at initialization time. MSORT asks if there is a 15-bit A/D converter on the AIP and assumes any response other than Y) means a 12-bit A/D converter.

---

[1] Strictly speaking, there exists a certain class of input files that MSORT will look at and ignore. Nothing is gained by specifying non-MIDAS files for input.

Subsequent to the A/D question, MSORT processes the files specified to JCL and makes certain legality checks. If MSORT finds a primary MIDAS input file, it types out the binary version number signaling that initialization is complete and it waits for command input. If the first MIDAS input file specified is not a MIDAS primary, if the primary is omitted entirely, or if the FPP is nonexistent, MSORT outputs an error message and returns to JCL. Unlike MIDAS, the LDP ↑C is always active during MSORT execution.

4.4 Keyboard Commands

The only legal terminator to command input is carriage return; any other terminating character generates an error message. The set of terminators are the same as MIDAS with the exception of ↑C. MSORT has three commands: EXIT, ID, and SORT.

4.4.1 EXIT

The exit command directs MSORT to save all sorted files and return to JCL; it requires no arguments.

4.4.2 ID

This command requires no arguments and describes the logical MIDAS file associated to the MIDAS primary that is on line to MSORT. The set consists of the primary and all those secondaries that were SAVEd with the primary at MIDAS EXIT time; this ID set should be distinguished from the set of files that were specified at MIDAS JCL time, of which they are a subset.

The point is that at MIDAS exit only those files which contain data are saved. The ID set is described on the Teletype in the following format:

```
P       XX      XX      XX      XX      XX      XX
Ø       dev     PRIME           LABEL
1       dev     FNAME           LABEL
.
.
.
n       (?)     FNAME           LABEL
```

The top line (P XX) is the primary label that was defined at MIDAS

initialization time (refer to 3.3 Initialization).

The first column of digits underneath the P is the sequential position of each LDP file within the logical MIDAS file. Ø is the primary, 1 is the first secondary, n is the n^{th} secondary; the secondaries may or may not exist. The information appropriate to a file is output on the same line as the sequence position number.

The first description of the file is the device upon which it is currently sitting; this is the device column where dev is in the set LTn, DKn n=Ø,7 e.g., LT4. If a secondary exists for the primary but was not specified at MSORT JCL time, that secondary is not on line and the file is missing. MSORT indicates this by entering (?) in the device column.

The next entry on the line is the name under which the file was saved at MIDAS exit time. It is not necessarily the same one that was specified by the user at MSORT JCL time. Since JCL has alias and renaming conventions over which MSORT has no control, it is possible to redefine a MIDAS file name between MIDAS EXIT time and MSORT time. MSORT supplies only the name under which the file was originally saved by MIDAS. MIDAS files that are moved via the File Handling Functions are still sorted by MSORT.

The last entry on the line is the label of the file that was entered in the parameter string at MIDAS JCL time. The ID command may be executed as often as necessary.

4.4.3 SORT

This command is given to do the actual sorting. The most general format is:

SORT;dev:n_1,dev:n_2,...,dev:n_m)

dev specifies the LDP volume (LTn,DKn) onto which the given run will be sorted, i.e., where it will be saved at MSORT EXIT time. The n argument is the run number that is to be sorted. A colon must separate device and run number. The SORT command itself must be delimited by a semicolon.

To specify a series of SORTS, repeat the sequence dev:n. Each SORT must be delimited by a comma with a carriage return after the last. File name specification is not necessary; this was done at MIDAS SETUP time (refer to 3.5.1 RUN NUMBER Question). If the file name was not properly specified at SETUP time, then JCL asks the user to rename at MSORT EXIT time. If the file name already exists on the LDP volume or if a given run is sorted more than once onto the same volume, the user gets a duplicate file name error from JCL.

Specifying more than eight SORTs in a SORT command or attempting to SORT after eight SORTs have been completed is not allowed by MSORT. When the SORT request is terminated with carriage return, MSORT scans and interprets the entire character string. Each SORT request is executed sequentially, beginning with the first. MSORT requires no user action until the entire SORT is completed and command input is inactive until that time. MSORT outputs the disposition of each SORT when a SORT request is completed and before going onto the next sort so that a log of the entire SORT will immediately follow the SORT command (refer to 4.5 Teletype Messages). MSORT sorts any run number that was assigned by MIDAS at SETUP time to any LDP volume that was defined as output at MSORT JCL TIME. Any number of sorts may be specified to any number of LDP output volumes, the limit, of course, is eight in all. The sorter will retain full information for 12 bit A/D, 15 bit A/D and 18 bit binary; however, for BCD to binary format MSORT will only retain the high order 18 of the 27 bits converted. Therefore users who have all their significance in the low order 9 bits effectively have no BCD to binary support MSORT.

4.5 Teletype Messages

SORT command messages differ from other Teletype messages because they usually come in pairs. The first message is the MSORT disposition for a given SORT and the second is the SORT request to which the disposition relates. This second message is an echo of the input string that defined the SORT request. Suppose the input was

```
SORT;ABC:28,DKØ:789QST;;)
```

MSORT responds with the following set of messages:

```
?20)          (DISPOSITION)
;ABC:28,)     (ECHO OF USER INPUT)
?22)          (DISPOSITION)
,DKØ:789QST;;)  (ECHO)
```

Where an echo type message is appropriate in the explanation of Teletype messages below it is referred to as ECHO). The ID message is explained above (refer to 4.4.2 ID).

4.5.1 DO YOU HAVE 15 BIT A/D?)
TYPE Y FOR YES)

This is the 15 bit A/D option asked at initialization time. Responding with a Y) and not having 15 bit data effectively tells MSORT to multiply the data by a power of 4.

4.5.2 GOOD SORT)
ECHO)

This message indicates that MSORT had all good input and all good output and that a good LDP file now exists.

4.5.3 INCONSISTENT INPUT)

This is an initialization time error message that means the sorter has found a non-MIDAS input file in the MSORT JCL command; a MIDAS secondary that does not agree with the primary (different primary labels) or duplicate MIDAS files in the same sequential position that both agree with the MIDAS primary (the same primary label). The offending input file is ignored by MSORT. Suppose that a user executes two different MIDAS runs but specifies the same primary label for both:

```
MIDAS DKØ:PØ,LTØ:SØ )
MIDAS LTØ:P1,LT1:S1)
```

MIDAS now has created two logical MIDAS files: (PØ,SØ), (P1,S1). Suppose that MSORT now is called by:

```
MSORT LT2:OUTPUT=DKØ:PØ,LTØ:P1)
```

MSORT accepts PØ as the primary MIDAS file and also realizes that PØ and P1 are MIDAS files with the same primary label but also in the same sequential position, i.e., both primaries. MSORT rejects P1 and works only with PØ. Consider the additional case:

```
MSORT LT2:OUTPUT=DKØ:PØ,LTØ:SØ,LT1:S1,LTØ:P1)
```

MSORT accepts PØ and SØ; rejects S1 because it is a duplicate of SØ and rejects P1 because it is a duplicate of PØ; this is a generous example that resulted in proper input to MSORT.

However, MSORT also accepts the following improper input:

```
MSORT LT2:OUTPUT=DKØ:PØ,LT1:S1)
            or
MSORT LT2:OUTPUT=LTØ:P1,LTØ:SØ)
```

The above input could produce some very unusual MSORT output files. The essential point here is that beyond primary label and sequence position MSORT cannot distinguish logical MIDAS files.

4.5.4 INPUT ERR <?>)

An illegal or undefined command has been given. MSORT ignores the input.

4.5.5 INPUT MISSING)
ECHO)

The run number specified in the sort begins on a MIDAS secondary file that is not on line. The sort request is ignored and MSORT goes on to the next SORT request or reverts to command input whichever is appropriate. This message never occurs if the entire logical MIDAS file is on line.

4.5.6 INPUT>OUTPUT)
ECHO)

MSORT outputs this message if the requested output volume is not large enough to hold all the data that will be sorted onto it, i.e., the input is greater than the output. In such cases, MSORT types out the

above message and ignores the sort. Note that MSORT cannot always tell in advance if the output file is too small but when it can it will refuse to SORT (refer to 4.5.12 NO MORE OUTPUT).

4.5.7 xxxx LOST POINTS↵
ECHO↵

If any data for the specified run has been lost at data acquisition time (refer to 3.7.4 *LOST DT n) MSORT has a record of it. Upon completion of the sort, MSORT outputs the number of lost data points for the run; if no points are lost, there is no message. xxxx[1] is the number of lost points. MSORT is referring to data that MIDAS lost and not to data that it lost; the sorter does not lose data.

4.5.8 MIDAS PRIMARY IS MISSING↵

If MSORT fails to find the primary MIDAS file at initialization time, it outputs the above message and returns to JCL; the program will not execute without it.

4.5.9 *MSORT* n↵

This is the binary version number message and is output upon successful completion of MSORT initialization. The initial version number (n) is ØØ (refer to 3.7.6 *MIDAS*n).

4.5.10 NO FPP↵

If the FPP is nonexistent or malfunctioning, MSORT outputs this message at initialization time and returns to JCL.

4.5.11 NO MORE INPUT↵
ECHO↵

MSORT outputs this message if an input file that is required to continue sorting the run number is not on line, i.e., a MIDAS secondary is missing. In the example in 4.5.3 above, suppose a run began somewhere on PØ and continued onto SØ, and only PØ was specified to MSORT. MSORT would retrieve the data on PØ but at the end of PØ it

[1]xxxx is modular 4096 so that 1 and 4097 lost points are 1 lost point to MSORT; xxxx is an octal number.

would look for SØ and not find it. MSORT would terminate the sort, retain the data it had processed, and output the above message. The run is called partially sorted. MSORT also outputs this message if the input was bad to begin with, e.g., (PØ,S1) in the example of 4.5.3. The message never occurs if a complete MIDAS file is on line.

4.5.12 NO MORE OUTPUT)
ECHO)

If the specified output device is filled before all the data for a given run is processed, MSORT outputs the above message and terminates the sort; the run is called partially sorted. Because of the INPUT> OUTPUT message, the NO MORE OUTPUT message will be uncommon.

4.5.13 8 SORTS ONLY)

If 9 sorts are requested in one sort command, MSORT outputs this message. This message is also output if 8 sorts have already been completed and more are requested. MSORT ignores sort requests in either of the above cases.

4.5.14 ?2Ø)
ECHO)

This message is a syntax or typing error for output device specification, e.g.,

```
,DKØ2Ø,        (missing colon)
,ABD:38,       (not LTn or DKn)
```

The sort request is ignored.

4.5.15 ?21)
ECHO)

A legal output device (LTn,DKn) was specified but was not defined as output at MSORT JCL time. This error covers the degenerate case of no output specified at MSORT JCL time. MSORT ignores the sort.

4.5.16 ?22)
ECHO)

This error is for non-numeric input to run number or missing delimiter (,) e.g.,

```
,LT2:A,              (non-numeric)
,LTØ:38DKØ:1Ø7,      (missing comma)
```

4.5.17 ?23)
ECHO)

A legal run number was specified but was never assigned by MIDAS during setup time (refer to 3.5.1 RUN NUMBER Question). MSORT ignores the sort request.

APPENDIX A

TIPS FOR THE USER

A.1 MIDAS

Putting MIDAS through a dry run is a good way to see if the hardware is functioning properly. A bad RK8 disk or LINCtape will be singled out via MFAIL or Monitor failure. To execute a dry run, call MIDAS in the usual manner. Respond yes or no to the BURST mode question as is appropriate and input carriage return in response to primary label (it is assumed that dry run data will be deleted upon return to JCL). Take clock input channel 1 to off line frequency. Set up a number of time based instruments all of which start on clock channel 1. The more instruments and the higher the data rates the better the test will be. If an external sync is readily available, set up an externally synched instrument also. When all instruments have been set up, raise clock input channel 1 (that is the Schmitt trigger) to line frequency to start MIDAS. If MIDAS proceeds to the internal run number message (*MIDAS*n) then the hardware and mass storage are good.

If MIDAS does not output the binary version number which signals the end of initialization or execute an error return to JCL, check that all RK8 disks and LINCtapes are write enabled.

Disk users desiring high throughput rates should not specify both LINCtape and disk at MIDAS JCL time because MIDAS sets throughput limit at the maximum limit of the slowest device (refer to the table in 3.5.6 DATA RATE Question). If n mass storage devices are available, take care to ensure that MIDAS does not use all of them for data acquisition because MSORT needs room for output. One way to guarantee this is never to specify all n of them at MIDAS JCL time. It is a good practice to specify LDP output volumes with large working areas in an effort to minimize the back and forth time between MIDAS, MSORT, and JCL. Disk users would do well to reserve at least 500_{10} blocks for sorting (see below). MIDAS is much more responsive in an environment that is dominated by externally synched instruments than in one that is not. Users who do not find it inconvenient to run under external synch are advised to do so.

The BURST mode table in section 2.6 represents best case conditions and is only approximate and may vary from MIDAS run to MIDAS run or from PDP-12 to PDP-12. Users should determine their own BURST mode limits. Temperature and humidity play a significant role in LINCtape transfer rates. The manner in which tapes are marked will also affect transfer rates; be sure that the mark clock is running within specification; keep tape heads clean. MIDAS is an interrupt driven program with a wide variety of time critical tasks to perform and, as such, it requires hardware that is reasonably healthy. If DIAL works that does not imply that MIDAS will. To run a LINCtape configuration in BURST mode for maximum speed use tapes that are 1000 blocks long, and specify only one output file. LINCtape transfer rates are lower at high block numbers. The LDP file name extension .EXT offers a convenient way to earmark a logical MIDAS file .P or .Ø for the primary and .1 through .7 for the secondaries.

A.2 MSORT

Sorting may be very time consuming, particularly if there is a large MIDAS file and LINCtape. The following table lists the hierarchy of device configurations for optimal sorting; the best configuration is first:

LEVEL	MIDAS INPUT	MSORT OUTPUT	
Ø	DKN_1	DKN_2	(across disk)
1	DKN_1	DKN_1	(within disk)
2	DKN_1	LTN_2	(disk to tape)
3	LTN_1	DKN_2	(tape to disk)
4	LTN_1	LTN_2	(across tapes)
5	LTN_1	LTN_1	(within tape)

The break between level 1 and 2 is very sharp and is the reason why single disk users are advised to leave a significant portion of their disk for MSORT. If some runs can be sorted within disk but not all of them, consider the possibility of sorting within disk and moving the sorted files via MOVE to free up the disk for more within disk sorts. The alternative is to do some sorting from disk to tape. Level 3 users may be able to advance to level 2 or 1 by simply moving the file from LINCtape to disk, provided that there is room on the disk.

LINC tape users are at levels 4 and 5. Level 4 allows multiple sorts to be specified. A number of across tape sorts should take only a few minutes. At level 5, MSORT travels up and down the tape for each read and write and takes much longer to run.

APPENDIX B

MIDAS OPERATION SUMMARY

JCL TIME

MIDAS dev:PRIME,...,dev:SECOND;LABELS)

PRIMARY LABEL

MO DAY YR HR MIN SEC)

KEYBOARD COMMANDS

EXIT), PAUSE;n), RELAYS), RESET),
SETUP), START;n), STOP;n), TEST;n)

SETUP (LINE FEED BACKS UP, CARIG RTN TERMINATES)

RNUMn:
supply an alphanumeric file name and remember your run number

USE RELAY?
reply with openn:closen)
START or PAUSE or STOP?
reply with 1;n) for Schmitt trig n (n=1,2,3)
or 2;n) for SNS LN n (n=Ø,14)
or 3;n) for STOP ON n POINTS
or 4;n) for STOP ON n SECONDS

DATA RATE? 8K AND TAPE = 12ØØ
8K AND DISK = 6ØØØ

reply with n HERTZ)
(Time based rates are 5ØØ, 25Ø, 125, 1ØØ, 5Ø, 25, 2Ø, 1Ø, 5,4,2,1)

X;Y AXES?

reply with 1;AIP) for time based
or 2;AIP) for AIP external sync
3;AIP) for PDP-12-A/D analog controlled

$20 \leq AIP \leq 27$ is a digital channel
$40 \leq AIP \leq 57$ is an analog channel
for analog controlled ln=4n

TTY MESSAGES	EXPLANATION
I AM BUSY???)	Reset needed, 8 on LN, AIP chan on LN.
INPUT ERR ?)	Syntax error; illegal RNUM; invalid reset, exit, or relays; FNAME too big; illegal argument; not 4Ø-47 for analog controlled; invalid in Burst mode.
*LOST DT n)	Data lost on run n.
MFAIL n)	MIDAS failure, exit now.
MIDAS n)	End of initialization or data acquisition.
*OVER RT n)	Too fast run n, abort.

APPENDIX C

MSORT OPERATION SUMMARY

JCL TIME

```
    MSORT dev:outØ,,,dev:outn=dev:PRIME,...,dev:SECOND)
```

KEYBOARD COMMANDS

```
    EXIT, ID, SORT;outdev:RNUM,...,outdev:RNUM
          ID
(PLAB)    P          XXX...XX)
(PRIME)   Ø          DEV  NAME      LABEL )
  S       1          (?)   "          "
  E       .
  C       .
  O       .
  N       N          DEV   "          "
  D
            (?) = not on line                )
```

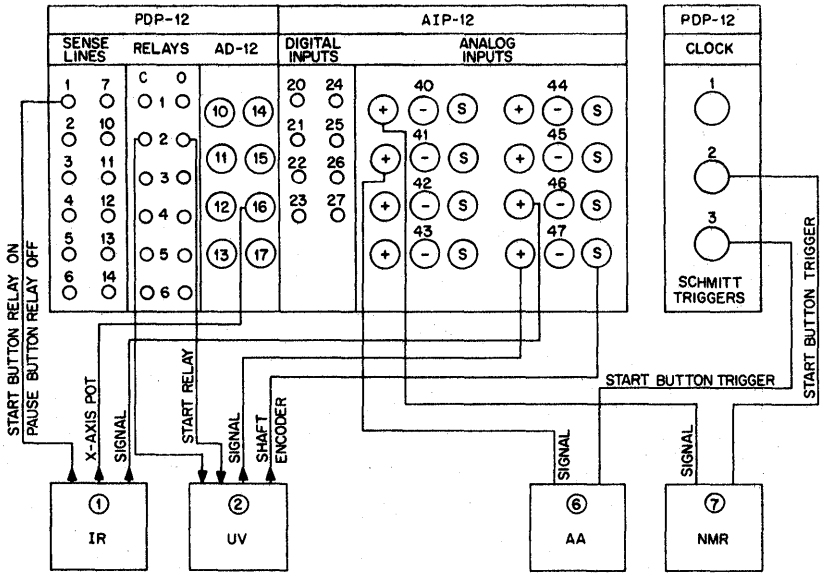
TTY MESSAGES (ECHO=echo of user sort request)	Explanation
GOOD SORT) ECHO)	Sort is successfully completed.
INCONSISTENT INPUT)	Bad JCL time input. MSORT ignores input file.
INPUT MISSING) ECHO)	Initial secondary not on line. MSORT ignores sort request.
INPUT>OUTPUT) ECHO)	Input too big. MSORT ignores sort request.
xxxx LOST POINTS) ECHO)	xxxx points were lost by MIDAS. MSORT retains output.
NO MORE INPUT) ECHO)	Subsequent secondary missing or bad input. Partial sort only.
NO MORE OUTPUT)	Input too big. Partial sort only.

8 SORTS ONLY↵	Too many sort requests.
?2Ø ↵ ECHO↵	Syntax error output device. SORT ignored.
?21↵ ECHO↵	Output device is not on line. SORT ignored.
?22↵ ECHO↵	Syntax error run number. SORT ignored.
?23↵ ECHO↵	Unassigned MIDAS run number. SORT ignored.

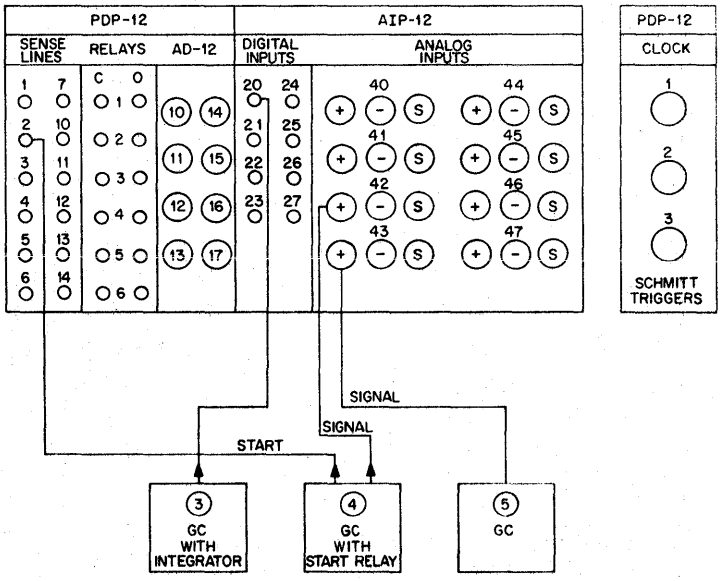
APPENDIX D

TYPICAL LAB ENVIRONMENT AND INSTRUMENT CONNECTIONS

A typical lab environment and instrument connections with their setup parameters are shown below. The operation need not be as follows; this is one possibility.



NOTE: All AIP signals should connect to both + and -.



RUN #	START	PAUSE	STOP	DATE RATE	RELAY	X-Y AXES
1	2;1	2;1	POINTS	7	NO	3;46
2	KBD	KBD	POINTS	10	2	2;47
3	KBD	KBD	TIME	10	NO	1;20
4	2;2	KBD	POINTS	5	NO	1;42
5	KBD	KBD	POINTS	20	NO	1;43
6	1;3	KBD	TIME	15	NO	1;41
7	1;2	1;2	POINTS	200	NO	1;40

APPENDIX E

ASSEMBLY INSTRUCTIONS FOR MIDAS/MSORT

MIDAS/MSORT are described by 8 sources: M, MØ, M1, M2, M3, MSØ, MS1, MS2; the binaries are generated by FPPASM. The sources M, MØ, M1, M2, M3 generate the MIDAS binary, and the sources M, MØ, MSØ, MS1, MS2 generate the MSORT binary. The source M is a series of equate statements which controls the assembly of the other sources. In M the symbols MØU, M1U, M2U, M3U, MSØU, MS1U, MS2U are unit chain arguments for the respective sources MØ, M1, M2, M3, MSØ, MS1, MS2; the sources can reside on any unit provided, of course, that they obey the conventions of the FPPASM CHAIN pseudo-op. To avoid chaining errors at assembly time, it is necessary that the chain arguments in M are consistent with the actual unit locations of the sources. The symbol MIDAS controls the assembly of MIDAS/MSORT and is currently set to 1. If MIDAS = 1, then FPPASM will generate the MIDAS binary; and if MIDAS = Ø, FPPASM will generate the MSORT binary. Other symbols in the source M are for non-standard applications and should be left alone.

1. ASSEMBLING MIDAS

```
→CL)
→AP M,UNIT)
→ZE)
→LO FPPASM,UNIT)
→SB MIDAS,UNIT,P14ØØØ)
```

2. ASSEMBLING MSORT

```
→CL)
→AP M,UNIT)
 REDEFINE THE SYMBOL MIDAS TO MIDAS = Ø
→ZE)
→LO FPPASM,UNIT)
→SB MSORT,UNIT,P1Ø2ØØ)
```

3. BUILDING MIDAS (MSORT)

```
BUILD DEV:MIDAS(MSORT))
RESPOND TO THE FIRST BUILD QUESTION WITH:
   MIDAS(MSORT),UNIT)
RESPOND TO ALL OTHER BUILD QUESTIONS WITH )
```

The parentheses are not to be taken literally; they imply that the BUILD procedure for MSORT is identical to that for MIDAS.

GLOSSARY

ACQUIRED DATA
Information that has been transmitted by an instrument and retrieved into core by MIDAS via some AIP channel (20-27, 40-57) (See also throughput data.)

ANALOG CONTROLLED INSTRUMENT
One of 3 types of MIDAS instruments whose data rate and acquisition is controlled externally by an interface into a PDP-12 A/D channel (10-17). (See also externally synched and time-based instrument).

BINARY VERSION NUMBER
A number which is attached to either the MIDAS or MSORT binary that is loaded by JCL and is output by the program at initialization time, and whose function is for program maintenance.

BURST MODE
One of two MIDAS data acquisition time modes selected at initialization time and which yields higher than normal throughput rates in exchange for certain MIDAS features.

CLOCK CHANNEL
A synonym for Schmitt trigger, and one of three phone jack inputs on the KW12A clock which MIDAS uses for starting, pausing, and stopping on line instruments.

COMMAND
A string of keyboard characters that is input by the user, delimited by a semicolon, terminated by a carriage return, and which directs either MIDAS or MSORT to perform a specific task, e.g., EXIT.

COMMAND INPUT MODE
MIDAS or MSORT are said to be in command input mode whenever they will accept and disposition a command.

DATA RATE
The number of points per second (Hertz) that an instrument is transmitting to MIDAS.

DEVICE (mass storage)
The PDP-12 hardware or peripheral that acts in conjunction with the CPU for the purpose of transferring data between core and some mass storage volume is called a device, i.e., the I/O controller. (See also LDP volume.)

EXIT TIME
The point of execution in the program when permanent record is made of output files.

EXTERNALLY SYNCHED INSTRUMENT
One of 3 types of MIDAS instruments whose data rate and acquisition are externally controlled by the instrument and AIP hardware via an interface into the AIP external sync (see also analog controlled and time based instrument).

INITIALIZATION
That portion of the program that occurs between loading by JCL and run time which executes tasks that are done only once; e.g., starting hardware, positioning mass storage volumes, etc.

INSTRUMENT
That hardware which is external to the PDP-12 but which is defined to the MIDAS environment via SETUP and user interface (see also run number).

INTERNAL RUN NUMBER
That number which MIDAS assigns to itself and which defines its current data acquisition mode (e.g., burst) (see also run number).

JCL TIME
The point of execution in the LDP system when the user is at the keyboard and the Job Control message "enter function" is on the scope.

LABEL
The portion of a MIDAS JCL time parameter string which allows a user to physically identify an LDP volume that contains a specific MIDAS file.

LDP VOLUME
That entity which is capable of storing data that is block addressable by some mass storage device is called a mass storage volume. A mass storage volume which has an LDP file structure (usually created by INIT) is an LDP volume; these are normally LINCtapes or RK8 disk paks. LDP volumes are accessed by MIDAS for throughputting data and subsequently by MSORT for the purpose of creating LDP floating point files.

LOGICAL MIDAS FILE
A string of LDP output files which contain the throughput image of an entire MIDAS run and which are saved at MIDAS exit time on separate LDP volumes. Logical MIDAS files differ from standard LDP files because they are not necessarily volume contained.

LOST DATA
Data transmitted by a started instrument but which is not accepted by the AIP hardware.

MAXIMUM THROUGHPUT RATE
That rate beyond which MIDAS cannot maintain on mass storage a true image of the data being generated within its environment; it depends on operating mode and hardware configuration.

MIDAS ENVIRONMENT
At a given point of time, the totality of all hardware that is capable of interacting with MIDAS.

OFF/ON LINE
All hardware that is geographically enveloped by the MIDAS environment is treated by MIDAS at any given point in time as either off line or on line. Hardware is on line if it is capable of acting upon or being acted upon by the MIDAS environment. If MIDAS is acquiring data without the use of Schmitt triggers, then the Schmitt triggers are off line. If a user

specifies only disk at MIDAS JCL time, then his LINCtape hardware is off line. Instruments that are throughputting data or that have been set up are on line. Instruments that are stopped or that have not been set up are off line.

OVER RATE
Instruments that throughput data at a rate greater than the one specified for it in the data rate question are over rate.

PAUSE
One of two states or modes peculiar to on line instruments. A paused instrument is dormant and incapable of transmitting data, but is on line and is capable of starting or stopping (see START).

PRIMARY LABEL
A string of up to 12 digits that is specified by the user at initialization and which is assigned to every LDP file that comprises the logical MIDAS file (see logical MIDAS file).

PRIMARY MIDAS FILE
The first LDP file in a logical MIDAS file; its sequence position is Ø and it is defined as the first LDP output file specified at MIDAS JCL time.

RESPONSE TIME INTERVAL
The elapsed time between the moment an event occurs in the MIDAS environment and the moment that MIDAS dispositions that event; the interval will vary across MIDAS environments but is fixed within a MIDAS environment.

RUN NUMBER
A number assigned by MIDAS to an instrument that is about to be set up and through which MIDAS, MSORT, and the user will address that instrument subsequent to setup. Run number and instrument are used synonymously in the manual.

SCHMITT TRIGGER
A synonym for clock channel (see CLOCK CHANNEL).

SECONDARY MIDAS FILE
Any LDP file that is part of a logical MIDAS file and is not the primary.

SENSE LINE
An external input level which is a digital input to the I/O bus and is either at +3V or at ground ØV. There are 13 sense lines all of which are used by MIDAS for starting, pausing, and stopping instruments.

SEQUENTIAL POSITION
The order assigned by MIDAS to an LDP file that is part of a logical MIDAS file. The primary is assigned sequential position Ø, the first secondary is 1, etc. The position is determined by the order in which the files are specified at MIDAS JCL time.

SETUP MODE
One of two MIDAS keyboard input modes which the user directs MIDAS to enter for the purpose of setting up an instrument (see also COMMAND INPUT MODE).

SORT CONTROL INFORMATION
Every primary MIDAS file contains information in a string of header blocks which are created by MIDAS at acquisition time and which are necessary input for the sorter; the set of header blocks is called the sort control information.

START
One of two states or modes peculiar to on line instruments. A started instrument is capable of transmitting and throughputting data.

STOP
The state of an off line instrument; one that is incapable of either transmitting data or starting.

TERMINATOR
A character used to terminate keyboard input. In MIDAS and MSORT, the terminators are the following characters and, in some cases, a subset of the following: carriage return, line feed, alt mode, and any character of the form ↑x, e.g., ↑A, ↑B, ↑C, etc.

TEST MODE
A special mode in MIDAS that allows users to acquire and display data from a given RUN without throughputting it.

THROUGHPUT DATA
Acquired data that is transferred out to mass storage (see acquired data).

THROUGHPUT RATE
The rate at which an instrument is throughputting data. The sum of the throughput rates of all on line instruments is the aggregate of MIDAS throughput rate.

TIME BASED INSTRUMENT
One of 3 types of MIDAS instruments whose data rate and acquisition are controlled internally via the KW12A clock (see also analog controlled and externally synched instrument).

INDEX[1]

[1]Refer also to the GLOSSARY, page G-1.

HOW TO OBTAIN SOFTWARE INFORMATION

Announcements for new and revised software, as well as programming notes, software problems, and documentation corrections are published by Software Information Service in the following newsletters.

Digital Software News for the PDP-8 & PDP-12
Digital Software News for the PDP-11
Digital Software News for the PDP-9/15 Family

These newsletters contain information applicable to software available from Digital's Program Library, Articles in Digital Software News update the cumulative Software Performance Summary which is contained in each basic kit of system software for new computers. To assure that the monthly Digital Software News is sent to the appropriate software contact at your installation, please check with the Software Specialist or Sales Engineer at your nearest Digital office.

Questions or problems concerning Digital's Software should be reported to the Software Specialist. In cases where no Software Specialist is available, please send a Software Performance Report form with details of the problem to:

Software Information Service
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

These forms which are provided in the software kit should be fully filled out and accompanied by teletype output as well as listings or tapes of the user program to facilitate a complete investigation. An answer will be sent to the individual and appropriate topics of general interest will be printed in the newsletter.

Orders for new and revised software and manuals, additional Software Performance Report forms, and software price lists should be directed to the nearest Digital Field office or representative. U.S.A. customers may order directly from the Program Library in Maynard. When ordering, include the code number and a brief description of the software requested.

Digital Equipment Computer Users Society (DECUS) maintains a user library and publishes a catalog of programs as well as the DECUSCOPE magazine for its members and non-members who request it. For further information please write to:

DECUS
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

MIDAS User's Manual
DEC-12-SQ3A-D

READER'S COMMENTS

Digital Equipment Corporation maintains a continuous effort to improve the quality and usefulness of its publications. To do this effectively we need user feedback -- your critical evaluation of this manual.

Please comment on this manual's completeness, accuracy, organization, usability and readability.

Did you find errors in this manual? If so, specify by page.

How can this manual be improved?

Other comments?

Please state your position. _____ Date: _____

Name: _____ Organization: _____

Street: _____ Department: _____

City: _____ State: _____ Zip or Country _____

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Digital Equipment Corporation
Software Information Services
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

OPERATION SUMMARIES

M I D A S

JCL TIME
MIDAS dev:PRIME,...,dev:SECOND;LABELS)

PRIMARY LABEL
MO DAY YR HR MIN SEC)

KEYBOARD COMMANDS
EXIT), PAUSE:n), RELAYS), RESET),
SETUP), START;n), STOP;n), TEST;n)

SETUP (line feed backs up, carriage return terminates)
RNUMn: supply an alphanumeric file name and remember your run number

USE RELAY? reply with open:close)

START or PAUSE or STOP?
reply with one of following:
1;n) for Schmitt trig n (n=1,2,3)
2;n) for SNS LN n (n=Ø,14)
3;n) for STOP ON n POINTS
4;n) for STOP ON n SECONDS

DATA RATE? 8K and tape = 12ØØ
8K and disk = 6ØØØ
reply with n HERTZ)
(Time based rates are 5ØØ, 25Ø, 125, 1ØØ, 5Ø, 25, 2Ø, 1Ø, 5, 4, 2, 1)

X;Y AXES?
reply with one of following:
1;AIP) for time based
2;AIP) for AIP external sync
3;AIP) for PDP-12-A/D analog controlled

20 ≤ AIP ≤ 27 is a digital channel

40 ≤ AIP ≤ 57 is an analog channel

for analog controlled ln=4n

TTY MESSAGES	EXPLANATION
I AM BUSY???)	Reset needed, 8 on LN, AIP chan on LN.
INPUT ERR ?)	Syntax error: illegal RNUM; invalid reset, exit, or relays; FNAME too big; illegal argument; not 4Ø-47 for analog controlled; invalid in Burst Mode.
*LOST DT n)	Data lost on run n.
MFAIL n)	MIDAS failure, exit now.
MIDAS n)	End of initialization or data acquisition.
*OVER RT n)	Too fast run n,,abort.

M S O R T

JCL TIME
MSORT dev:outØ,,,dev:outn=dev:PRIME,...,dev:SECOND)

KEYBOARD COMMANDS
EXIT, ID, SORT; outdev:RNUM,...,outdev:RNUM

ID

(PLAB)	P	XXX.XX)		
(PRIME)	Ø	DEV	NAME	LABEL)
S	1	(?)	"	"
E	.			
C	.			
O	.			
N	N	DEV	"	"
D				

(?) = not on line)

TTY MESSAGES	EXPLANATION
(ECHO = echo of user sort request)	
GOOD SORT) ECHO)	Sort is successfully completed.
INCONSISTENT INPUT)	Bad JCL time input. MSORT ignores input file.
INPUT MISSING) ECHO)	Initial secondary not on line. MSORT ignores sort request.
INPUT>OUTPUT) ECHO)	Input too big. MSORT ignores sort request.
xxxx LOST POINTS)	xxxx points were lost by MIDAS. MSORT retains output.
NO MORE INPUT) ECHO)	Subsequent secondary missing or bad input. Partial sort only.
NO MORE OUTPUT)	Input too big. Partial sort only.
8 SORTS ONLY)	Too many sort requests.
?2Ø) ECHO)	Syntax error output device. SORT ignored.
?21) ECHO)	Output device is not on line. SORT ignored.
?22) ECHO)	Syntax error run number. SORT ignored.
?23) ECHO)	Unassigned MIDAS run number. SORT ignored.

OPERATION SUMMARIES

M I D A S

JCL TIME
MIDAS dev:PRIME,...,dev:SECOND;LABELS)

PRIMARY LABEL
MO DAY YR HR MIN SEC)

KEYBOARD COMMANDS
EXIT), PAUSE:n), RELAYS), RESET),
SETUP), START;n), STOP;n), TEST;n)

SETUP (line feed backs up, carriage return terminates)
RNUMn: supply an alphanumeric file name and remember your run number

USE RELAY? reply with open:close)

START or PAUSE or STOP?
reply with one of following:
1;n) for Schmitt trig n (n=1,2,3)
2;n) for SNS LN n (n=Ø,14)
3;n) for STOP ON n POINTS
4;n) for STOP ON n SECONDS

DATA RATE? 8K and tape = 12ØØ
8K and disk = 6ØØØ
reply with n HERTZ)
(Time based rates are 5ØØ, 25Ø, 125, 1ØØ, 5Ø, 25, 2Ø, 1Ø, 5, 4, 2, 1)

X;Y AXES?
reply with one of following:
1;AIP) for time based
2;AIP) for AIP external sync
3;AIP) for PDP-12-A/D analog controlled

20 ≤ AIP ≤ 27 is a digital channel

40 ≤ AIP ≤ 57 is an analog channel

for analog controlled ln=4n

TTY MESSAGES	EXPLANATION
I AM BUSY???)	Reset needed, 8 on LN, AIP chan on LN.
INPUT ERR ?)	Syntax error; illegal RNUM; invalid reset, exit, or relays; FNAME too big; illegal argument; not 4Ø-47 for analog controlled; invalid in Burst Mode.
*LOST DT n)	Data lost on run n.
MFAIL n)	MIDAS failure, exit now.
MIDAS n)	End of initialization or data acquisition.
*OVER RT n)	Too fast run n,,abort.

M S O R T

JCL TIME
MSORT dev:outØ,,,dev:outn=dev:PRIME,...,dev:SECOND)

KEYBOARD COMMANDS
EXIT, ID, SORT;outdev:RNUM,...,outdev:RNUM

```
        ID
(PLAB)  P          XXX.XX)
(PRIME) Ø          DEV  NAME      LABEL)
   S    1          (?)   "         "
   E    .
   C    .
   O    .
   N    N          DEV   "         "
   D
        (?) = not on line                )
```

TTY MESSAGES	EXPLANATION
(ECHO = echo of user sort request)	
GOOD SORT) ECHO)	Sort is successfully completed.
INCONSISTENT INPUT)	Bad JCL time input. MSORT ignores input file.
INPUT MISSING) ECHO)	Initial secondary not on line. MSORT ignores sort request.
INPUT>OUTPUT) ECHO)	Input too big. MSORT ignores sort request.
xxxx LOST POINTS)	xxxx points were lost by MIDAS. MSORT retains output.
NO MORE INPUT) ECHO)	Subsequent secondary missing or bad input. Partial sort only.
NO MORE OUTPUT)	Input too big. Partial sort only.
8 SORTS ONLY)	Too many sort requests.
?2Ø) ECHO)	Syntax error output device. SORT ignored.
?21) ECHO)	Output device is not on line. SORT ignored.
?22) ECHO)	Syntax error run number. SORT ignored.
?23) ECHO)	Unassigned MIDAS run number. SORT ignored.