REMOTE MAINTENANCE DOS
REMDOS
User's Guide

REMDOS

RDOSSLAV 9404
RDOSMAST 948.1

$DISPLAY .... NACHRICHT IN DIE
UNTERSTE ZEILE AN SLAVE

$ASK ....WIE O. MIT NACHRICHT VON SLAVE
NACH 3MINUTEN TIME OUT

$SEND FILEA, FILE2;0 WIE COPY VON M→SL.
$RECV FILEA, FILE2;0 WIE COPY VON SL→M.
$OFF ZURUCK ZUM DOS
$OFF ---- DOS BEFEHL DURCHFUEHREN
IN MASTER BLEIBT ACTIV

[illegible]
[illegible]

Version 1

August, 1979

Document No. 50465

DATAPOINT CORPORATION

The leader in dispersed data processing ™.

REMOTE MAINTENANCE DOS
REMDOS

User's Guide

Version 1

August, 1979

Document No. 50465

PREFACE

REMote maintenance DOS allows a master processor to control a slave processor by intercepting all console input/output generated by a program executing at the slave processor and performing these functions on the master processor.

Both processors involved must have a 5500 compatible instruction set and support some sort of synchronous communications (9481, 9404 or 1800 ICA). A baud rate of up to 9600 baud is supported on all devices on a non-continuous carrier line.

Additional features allow the master processor to send and recieve files to and from the slave processor and allow the master and slave operaters to talk (keyboard to keyboard) to each other. There is also a command to initiate a job at the slave site upon termination of REMDOS. The master operater has the option of directing his command line to execute either locally or at the remote site.

TABLE OF CONTENTS

CHAPTER 2. SITE REQUIREMENTS

2.1 Hardware requirements

REMDOS requires a synchronous communications adaptor (9481, 9404, 1800 ICA) and Bell compatible modems with non-continuous carrier at both the master and slave sites. The processors at both sites must have a 5500 compatible instruction set along with some form of disk (either stand alone or under ARC) and a minimum of 24K of memory.

REMDOS will support communications of up to 9600 baud over either dial up or leased lines.

2.2 Installation.

The MIN DOS utility is required to load REMDOS onto a datapoint disk. Simply load the REMDOS cassette into the front tape deck and execute the MIN;A command.

2.3 Software requirements

REMDOS requires a functional DOS to load and interface with at both the slave and master sites. REMDOS will not run under PS (the partition supervisor). Only the two files named RDOSSLAV (/CMD and /REL) are neccessary to operate the slave REMDOS site. Likewise, only the two RDOSMAST files need to be on the disk of the master.

CHAPTER 3. OPERATION

3.1 Slave Operation

The operater at the slave site need only type in RDOSSLAV. This program will ask the slave operater what type of communications device he would like to use. The operater should respond with 9481, 9404 or ICA, depending on which he has configured on his system. REMDOS then loads the slave communications driver and DOS intercepts into the available upper memory and waits for communications to be established with the master console. At this point, the console of the slave should display the message "WAITING FOR DATA MODE" until the master station establishes communications. While the slave program is waiting for data mode, the KEYBOARD key may be depressed in order to abort the program and return to normal DOS.

After communications has been established, the scrren of the slave is filled with the message:

"REMOTE MAINTENANCE DOS VERSION 1.n IN OPERATION"

"DO NOT DISTURB"

"UNLESS INSTRUCTED BY MASTER SITE"

From this point on the master has control of all keyin and display functions of the slave. DOS commands may now be invoked from the master site with the results displayed at the master site.

In addition to DOS commands, a special communications command may be invoked which will allow information keyed in at the master console to be displayed at the slave console. The master may also request that the slave respond to a question. When such a response is requested, the slave console will display the question and begin beeping at 4 second intervals until the slave operater responds by hitting a key (any key) to begin answering the question. When the slave operater presses the enter key, the message is transmitted back to the master console and appears on the bottom line of the master console's screen. If after 3 minutes, no key has been hit, the message "*** TIMED OUT ***" will be displayed at the master console. The processor then continues under the control of the master console.

When the processor displays the normal DOS sign-on message, co rol has been returned to the slave processor and DOS operation may resume.

3.2 Master Operation

.2.1 Initialization

REMDOS master operation begins by executing the program DOSMAST. The first thing the program will ask for will be the communications device type. The operater must type in 9481, 9404, CA (or * to abort). The console of the master port will then isplay "WAITING FOR DATA MODE". The master operater should then al the phone to connect with the slave who should be waiting for nging. If the master operater decides to abort operation, he y do so by depressing the KEYBOARD key. From this point on the ter has control of all keyin and display functions of the e. DOS commands may be invoked from the master site and rected to either the masters local processor (by preceeding the mmand with a "&") or to the remote processor. REMDOS sets the and $REMARC flags. Please see the appendix for programs that ll NOT run under REMDOS.

In addition to the standard DOS commands, there are several pecial REMDOS commands to aid the diagnostic operater. These mmands are as follows:

$DISPLAY <string> --- Displays the entered string on the bottom line of the slave console.

$ASK <string> --- Displays the entered string on the bottom line of the slave and alerts the slave operater that a one line response is requested. After the slave keys in a response, it is displayed at the master site. If after 3 minutes, the slave has not responded, the string "*** TIMED-OUT ***" will be displayed.

$SEND <filespec1>[,<filespec2>][;O] --- Transmits a file from the master site to the slave site. The first file (<filespec1>) is the input file to be read from the master site's disk. TXT is the default extension for this file spec. The second file (<filespec2>) is the output file name to be written on the slave site's disk. The default for filespec2 is filespec1. If the file already exists on the slave's disk, the ;O option must be specified to over-write it. While the transfer is taking place, a counter will appear on the screen of the master console indicating how many sectors have been transfered. If at any time during the transfer the master operater decides to abort the send, the KEYBOARD key may be depressed and a call to CLOSE$ will be executed, closing the file and aborting the send.

$RECV <filespec1>[,<filespec2>][;O] --- Transmits a file from the slave site to the master site. The first file (<filespec1>) is the input file to be read from the slave site's disk. TXT is the default extension for this file spec. The second file (<filespec2>) is the output file name to be written on the master site's disk. The default for filespec2 is filespec1. If the file already exists on the master's disk, the ;O option must be specified to over-write it. While the transfer is taking place, a counter will appear on the screen of the master console indicating how many sectors have been transfered. If at any time during the transfer the master operater decides to abort the send, the KEYBOARD key may be depressed and a call to CLOSE$ will be executed, closing the file and aborting the send.

$OFF --- Terminates communication between the master and slave and returns both processors back to DOS READY.

$OFF;R --- Terminates communication between the master and slave and returns the master back to DOS READY. The slave, however, waits for the master to call back and re-establish communications, leaving the master free from the cost of the phone line while he does extended local processing.

$OFF <DOS command> --- Terminates communication between the master and slave, while the slave station executes the DOS command line. The master will return to DOS READY. This command is very useful in initiating a remote job at REMote DOS termination.

CHAPTER 4. SUGGESTED USAGE

When software problems occur and there is not a competent programmer around to fix them, this program will allow a programmer with a local Datapoint system to dial up the problemmed remote user and work his magic without having to talk the poor, unsuspecting remote slave operater through making difficult changes in a system the programmer knows intimately.

Several disk maintenance utilities exist (for example: DSKCHECK) which allow the systems programmer to take a look into the internals of a disk pack and perhaps modify suspected errors which the local operater probably wouldn't know to look for. These uses are perfect for REMDOS and can in many cases be taken care of in a 10 minute long distance phone call (processor to processor).

CHAPTER 5. PROGRAMMER'S GUIDE.

5.1 How it works.

REMDOS works by intercepting some of the DOS entry points in the slave's processor and re-routing them into the REMDOS intercept routines and communications driver which are loaded into a 4K region of upper memory allocated by the loader program RDOSSLAV at initialization time. When RDOSSLAV is entered, it first determines a load address (if ARC is running, system RAM, otherwise it loads into the highest memory available by calling DOS function 10, 1), then it loads the propper communications driver from RDOSSLAV/REL into that location and transfrs control to its' load address. When slave program takes over, it intercepts the LOADX$, DSPLY$, KEYIN$, TCWAIT and DOSFNC as described below and saves the memory size of the machine (as obtained from DOS function 10, 1). If ARC is active, 4K is subtracted from this value to handle the allocation of non-system RAM for future DOS function 10, 1 calls (see below). Both the PSACT flag from DOSFLAG and thee $REMARC flag from DOSFLG2 are set by the REMDOS slave. The DOS entry points intercepted are:

5.1.1 LOADX$ intercept.

Step 1 --- Invoke the loader at the slave site.

Step 2 --- If the DOS function loader is reloaded, intercept it.

Step 3 --- If PFN 0 (SYSTEM0) is reloaded, intercept KEYIN$, DSPLY$, LOADX$ and TCWAIT.

Step 4 --- Return to caller.

5.1.2 DSPLY$ intercept. .

consStep 1 --- Send the string and all necesary registers to the master console.

Step 2 --- Display the string on the master's console.

Step 3 --- Send acknowledging message back to the slave.

Step 4 --- Return to caller.

5.1.3 KEYIN$ intercept.

Step 1 --- Send all necesary registers to the master console.

Step 2 --- Obtain the string from the master's keyboard.

Step 3 --- Send the string and registers back to the slave.

Step 4 --- Return to caller.

5.1.4 DOSFNC intercept.

Step 1 --- If DOS function 10, subfunction 1, execute the function, restore the saved memory size to the DOS returned table and return to caller.

Step 2 --- If DOS function 6, send all neccesary registers to the master console, execute the DOSFNC there and return the results to the caller in the slave processor.

Step 3 --- Otherwise, go back to DOS to execute the DOS function normally at the slave and return to the caller.

5.1.5 TCWAIT intercept.

Step 1 --- Send a DOS idle message to the master and wait for a response.

Step 2 --- At the master, the cursor is displayed at the bottom left hand side of the screen and a response is waited for. If the first character is an anpersand (&), the response is a local command and will be run on the REMDOS master consolse as if under normal DOS. If the first character is a dollar sign ($), the response is a REMDOS command (see above). Otherwise, the response is sent to the slave to be entered as a DOS command. If the REMDOS command is either a SEND or RECV command, files are opened and prepared on both ends and a file transfer is begun. If the command is an ASK or DISP, then the string is sent and a response is waited for. If the command is an OFF, the string is sent and communications is terminated with the slave.

Step 3 --- After the respone is obtained, a DOS command is checked for. If so, the command is entered into MCR$ and CMDINT is jumped to. If not, then the appropriate action is taken to process the REMDOS command (see the above section on REMDOS commands).

5.2 Programs written to run under REMDOS.

The following are some hints in writing good remote DOS programs:

The first and most important rule is DO NOT ADDRESS THE KEYBOARD OR SCREEN DIRECTLY. Always use the DOS functions and KEYIN$ and DSPLY$ to talk to the console. This has always been a good programming practice due to the differences in hardware in some processors, but under remote DOS, a program can hang up until the system is booted.

Programs which intercept DOS entry points (such as TCWAIT, DSPLY$, etc.) will not run under REMDOS. There are a few exceptions to this rule (such as ARC, CHAIN) that do run, but only due to hours of programming effort.

It is a good idea to limit program interaction with the operater to a tasteful minimum. Datapoint software as a rule is highly interactive and this philosophy should not change under REMDOS. It is important for the remote operater to know what is going on at the slave, but remember that every time a display or keyin or check for the depression of the keyboard and display keys is done, a packet must be transmitted to the master site, and a reponse must be sent from the master to the slave. It is best therefore to combine DSPLY$ calls into one long string when possible and to limit the checking of the depression of the KEYBOARD and DISPLAY keys.

The use of KEYIN$ has always been a much better choice than successive DOS function 6 calls for getting character strings from the keyboard. Although it is nice to achieve the keyin continuous effect of DATASHARE, you will find under REMDOS that this effect is nullified by the tremendous overhead of the line turn-around speed.

APPENDIX A.SUPPORTED DOS UTILITIES.

The following DOS utilities should run correctly under REMDOS. All utilities should be DOS version 2.4 or later except where noted.

```
APP        ABTONOFF   BLOKEDIT   BOOT       BUILD      CAT
CHAIN      CHANGE     COPY       DSKCHECK   ENCODE     FILES
FIX        FIXAPPLY   FREE       INDEX      KILL       LIST
NAME       REFORMAT   SAPP       SORT       SUR
                                *FASTSORT*
```

APPENDIX B.OTHER SUPPORTED PROGRAMS.

The following Datapoint software is supported to run under REMDOS.

FASTSORT FIXREL LIBSYS MOUNT