SPERRY╋UNIVAC
COMPUTER SYSTEMS

＃＃

ATTN: CHARLIE GIBBS

00955
CAV208M45541   UP   8822   R1A

SPERRY UNIVAC   UAS
SUITE 906
1177 WEST HASTINGS ST
VANCOUVER   BC   V6E 2K3   CAV
##

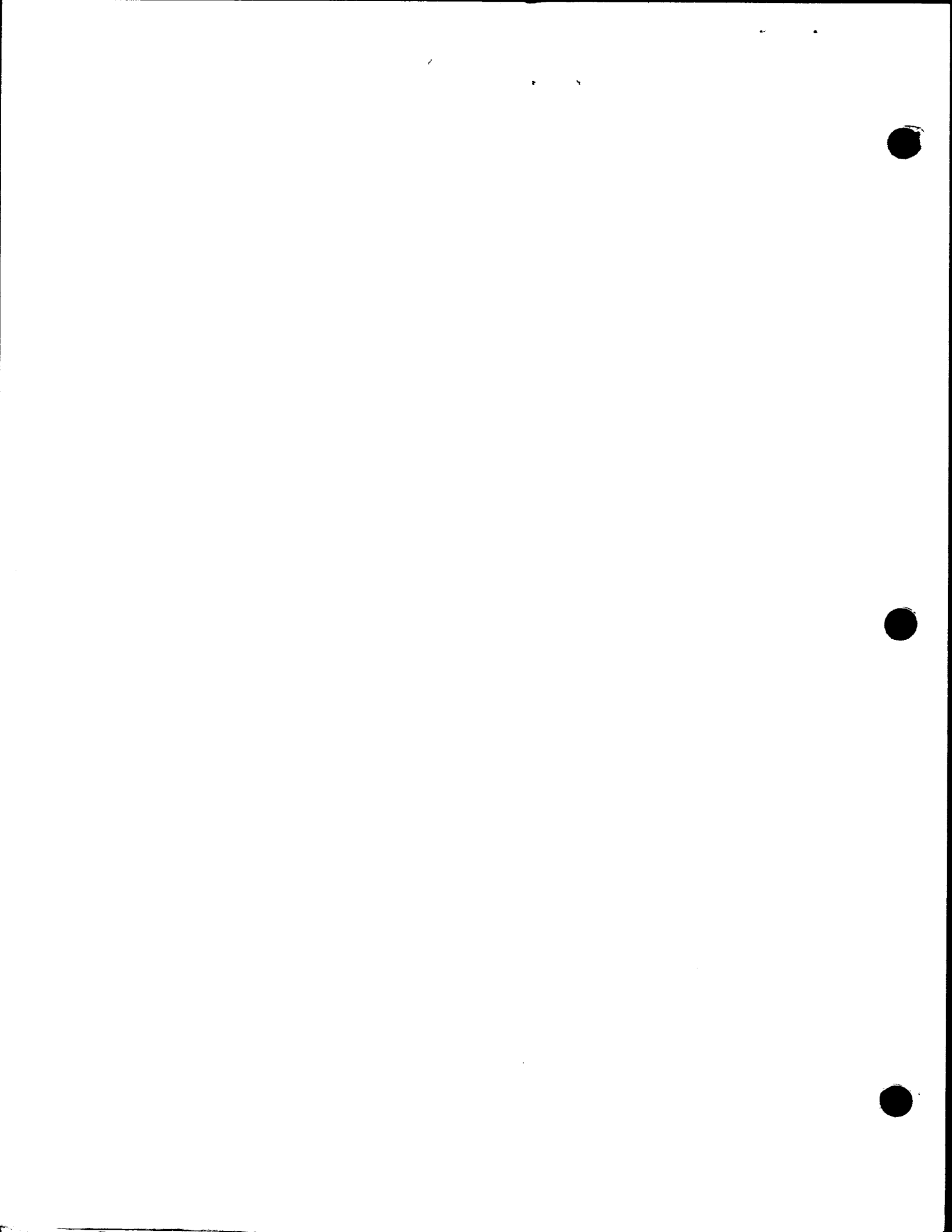**This Library Memo announces the release and availability of Updating Package A to "SPERRY UNIVAC Operating System/3 (OS/3) Interactive Job Control User Guide", UP-8822 Rev. 1.**

This update for the 7.1 release corrects a typographical error in the spelling of the RV JC$BLD command and specifies that RV JC$BLD should be used to initiate a job control dialog that uses the dialog processor audit version.

Copies of Updating Package A are now available for requisitioning. Either the updating package only or the complete manual with the updating package may be requisitioned by your local Sperry Univac representative. To receive only the updating package, order UP-8822 Rev. 1—A. To receive the complete manual, order UP-8822 Rev. 1.

LIBRARY MEMO ONLY	LIBRARY MEMO AND ATTACHMENTS	THIS SHEET IS
Mailing Lists BZ, CZ and MZ	Mailing Lists 18,18U,19,19U,20,20U, 21,21U,28U,29U,75,75U,76 and 76U (Package A to UP-8822 Rev. 1, 9 pages plus Memo)	Library Memo for UP-8822 Rev. 1—A
		RELEASE DATE: December, 1981

SPERRY UNIVAC   UAS
SUITE 906
1177 WEST HASTINGS ST
VANCOUVER   BC   V6E 2K3   CAV

ATTN: CHARLIE GIBBS

00250
CAV208M45541   UP   8076   R8A

This Library Memo announces the release and availability of "SPERRY UNIVAC® Interactive Job Control User Guide", UP-8822 Rev. 1.

The modifications made for this revision include:

— A new explanation of character substitution for the /* job control statement (via // OPTION EOD=xx).

— Addition of the CC job control statement to the job control statement master menu.

— An explanation about specifying the DD, LCB, and VFB job control statements using the GENERAL ENTRY menu selection.

— Changing of the SC JC$BUILD command to SC JC$BLD.

Additional copies may be ordered by your local Sperry Univac representative.

LIBRARY MEMO ONLY	LIBRARY MEMO AND ATTACHMENTS	THIS SHEET IS:
Mailing Lists BZ, CZ and MZ	Mailing Lists 18, 18U, 19, 19U, 20, 20U, 21, 21U, 28U, 29U, 75, 75U, 76 and 76U (Covers and 41 pages)	Library Memo
		RELEASE DATE: February, 1981

Interactive Job Control

OS/3

User Guide

SPERRY ⬩ UNIVAC

H

PAGE STATUS SUMMARY

ISSUE: Update A — UP-8822 Rev. 1
RELEASE LEVEL: 7.1 Forward

Part/Section	Page Number	Update Level	Part/Section	Page Number	Update Level	Part/Section	Page Number	Update Level
Cover/Disclaimer		Orig.						
PSS	1	A						
Preface	1	Orig.						
Contents	1, 2	Orig.						
1	1 thru 4	Orig.						
2	1 2 thru 9 10, 11 12, 13	A Orig. A Orig.						
3	1, 2	Orig.						
4	1 thru 3	Orig.						
5	1 thru 5	Orig.						
6	1, 2	Orig.						
Glossary	1 thru 3	Orig.						
Index	1 thru 3	Orig.						
User Comment Sheet								

All the technical changes are denoted by an arrow (➤) in the margin. A downward pointing arrow (↓) next to a line indicates that technical changes begin at this line and continue until an upward pointing arrow (↑) is found. A horizontal arrow (➤) pointing to a line indicates a technical change in only that line. A horizontal arrow located between two consecutive lines indicates technical changes in both lines or deletions.
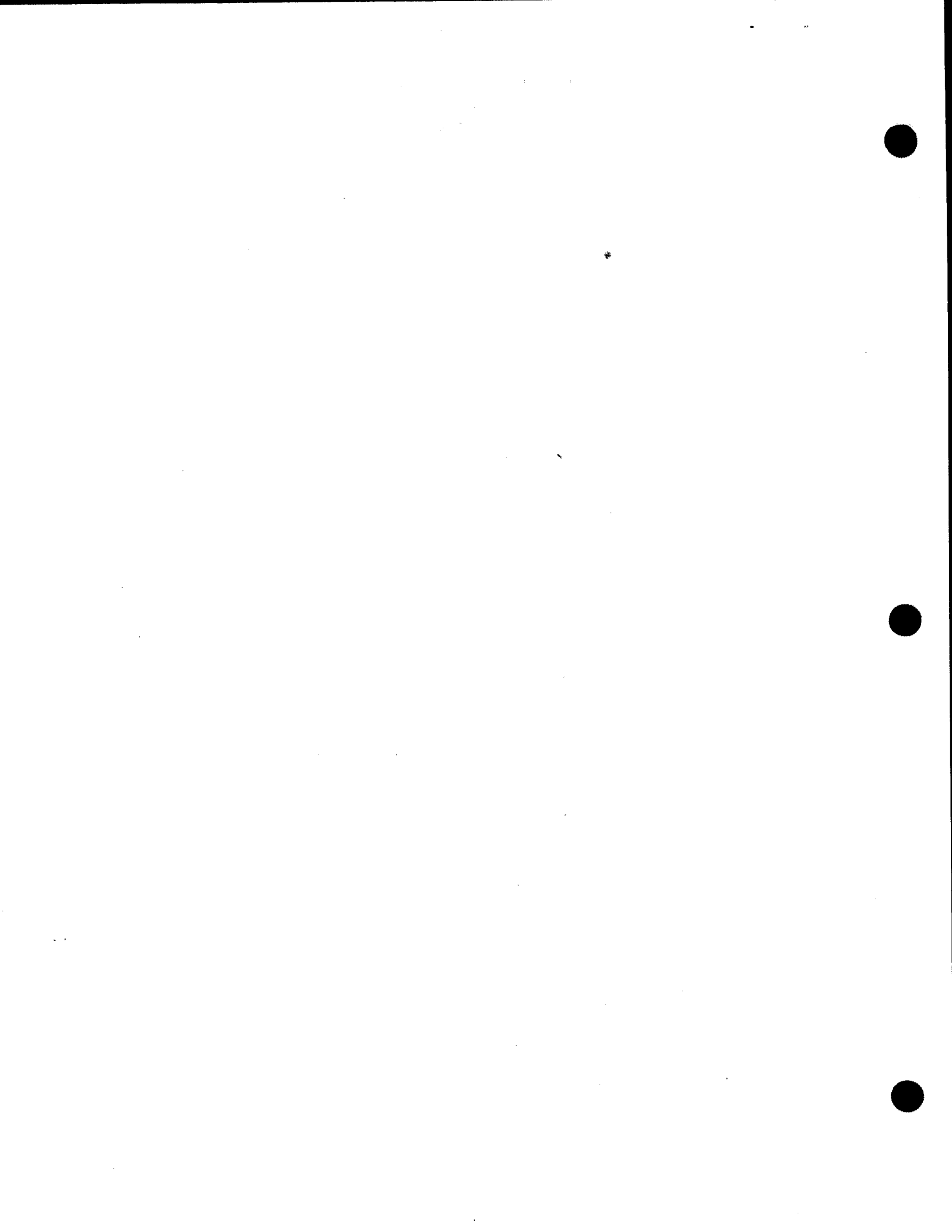
Preface

This manual is one of a series designed to explain SPERRY UNIVAC Operating System/3 (OS/3). It specifically describes how job control functions in an interactive processing environment.

The intended audience for this manual is the user with a general knowledge of interactive processing and OS/3 job control. The following areas of interactive job control are explained:

- Building a job control stream or user jproc from a workstation using the job control dialog

- Changing dialog responses

- Running a stored control stream from a workstation

- Submitting values for run-time symbols from a workstation

- Changing control stream execution from a workstation

- Identifying user dialogs and screen format services with the USE job control statement

In addition, workstation commands related to job processing are summarized and a glossary is provided that defines terms used throughout the manual.

This manual contains references to other, related OS/3 manuals. A list of these manuals and their UP numbers follows:

- Dialog processor user guide, UP-8858 (current version)

- Interactive services commands and facilities user guide/programmer reference, UP-8845 (current version)

- Job control user guide, UP-8065 (current version)

- Screen format services concepts and facilities manual, UP-8802 (current version)
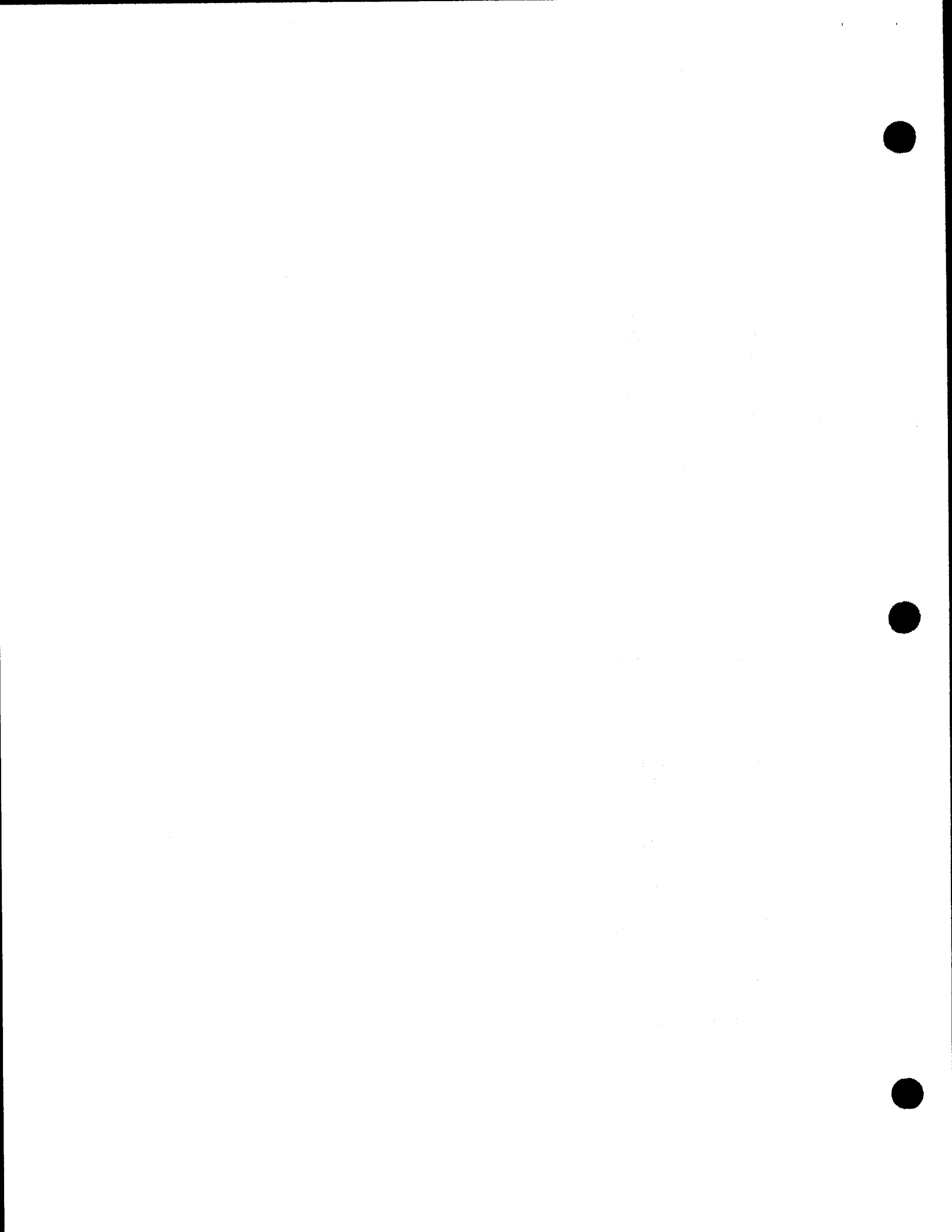
Contents

1. General Concepts

1.1. INTRODUCTION

OS/3 job control:

■ identifies jobs to the operating system;

■ reserves the hardware and software resources needed by jobs;

■ reserves enough main storage to run jobs; and

■ helps control files.

When you use OS/3 job control in an interactive environment, you communicate directly with the operating system through a workstation, receiving immediate verification of your input. You should be familiar with OS/3 job control concepts before you read this manual, which is confined to describing how to use OS/3 job control in an interactive environment (Figure 1-1).

OS/3 job control concepts and detailed explanations of job control statements and jprocs are presented in the OS/3 job control user guide.

1.2. OS/3 JOB CONTROL IN AN INTERACTIVE ENVIRONMENT

OS/3 job control directs the processing of your job. OS/3 *interactive* job control helps you define your processing options through a direct "conversation" between you and the operating system – the job control dialog. Your responses to the job control dialog are used by JC$BLD, a system program, to create a job control stream containing all the job control statements and system jprocs you need to process your job the way you want it processed. The job control dialog "prompts" you with HELP screens (when you request them) that explain the job control statements, system jprocs, and individual parameters you don't understand. The more experienced user can request HELP screens only if they're needed. The job control dialog helps both the experienced and novice user build a valid job control stream, not only because it prompts you, but also because it rejects job control statements that contain syntactical errors. The job control dialog can be used to build a job control stream or a user job control procedure (jproc).

You can use interactive job control in conjunction with other OS/3 components to build, store, and run control streams from the workstation. You can enter control stream embedded data and build multijob-step job streams from the workstation. Another interactive component of OS/3, the dialog processor, allows you to save your dialog responses and change or correct them in a subsequent session – creating a new control stream or user jproc quickly and easily from the workstation.

Interactive job control, then, gives you the power of OS/3 job control and the ease-of-use advantages of interactive processing.

It should be noted that the job control streams and user jprocs you build interactively can be run in a batch environment. We'll confine our discussion, however, to interactive applications.


```
                    YOU CAN:


        BUILD A CONTROL STREAM;
        BUILD A USER JPROC;
        CHANGE DIALOG RESPONSES;
        SUBMIT THE VALUES FOR RUN-TIME VARIABLES;
        CHANGE CONTROL STREAM EXECUTION.
```
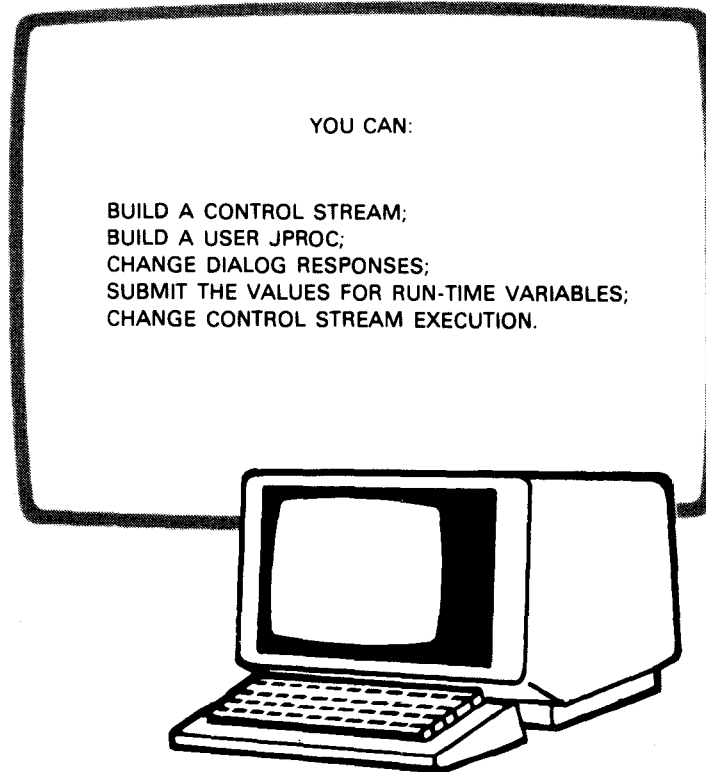
Figure 1—1. Job Control in an Interactive Environment

NOTE:

The job control dialog will not function properly from communications terminals that do not have the field protect feature.

1.3. CONVENTIONS USED IN PRESENTING JOB CONTROL STATEMENTS AND WORKSTATION COMMANDS

The conventions used in this manual to present job control statements and workstation commands are explained in the paragraphs and examples that follow.

1. Commands and parameters in capital letters must be keyed in exactly as shown.

   ```
   CANCEL jobname,D
   ```

2. Underscoring indicates that commands or parameters may be abbreviated; only the underscored characters are required as keyins.

   ```
   RUN [jobname]
   ```

3. Parameters constructed using lowercase letters and embedded hyphens designate variables that must be defined by the user.

   ```
   vsn,lbl,module-name
   ```

4. Optional parameters are enclosed by brackets.

   ```
   [,priority]
   ```

5. Alternative choices for a parameter are enclosed by braces. If you specify the parameter, pick only one of the choices listed.

   $$\left\{ \begin{array}{l} \underline{ALL} \\ \underline{HIGH} \\ jobname \\ \underline{NOR} \\ \underline{PRE} \end{array} \right\}$$

6. An ellipsis following a parameter indicates that more than one parameter may be specified.

   ```
   parameter-1,. . . ,parameter-n
   ```

7. Commas, equal signs, and parentheses must be keyed in exactly as shown.

   $$\underline{BRKPT} \left\{ \begin{array}{l} \left\{ \begin{array}{l} P \\ I \end{array} \right\} \quad , \quad \left\{ \begin{array}{l} PR \\ PU \end{array} \right\} ,JOB=jobname[,modifier-1,. . . ,modifier-n] \\ \underline{CNSLG} \end{array} \right\}$$

   ```
   DUMP[(did)]
   ```

8. Shaded parameters are default values automatically generated by the system even if you don't specify the parameter.

$$//[\text{symbol}] \ \text{USE SFS} \left[,\begin{Bmatrix} \text{format-file-lfd} \\ \text{SYSFMT} \end{Bmatrix} \right] \dots$$

2. Using the Job Control Dialog

2.1. JOB CONTROL DIALOG

The job control dialog is an interactive facility of OS/3 that guides you through the process of building a job control stream or user jproc from a workstation. To begin a job control dialog session, key in SC JC$BLD. This activates the dialog processor and opens the job control dialog file. Dialog text is displayed at the workstation screen and your responses to the dialog are entered at the workstation keyboard. The dialog processor passes your responses to the system program JC$BLD, which creates your control stream or jproc and stores it in the system file YJCS or your own permanent library file. The functions of the dialog processor, which manages a dialog session, are detailed in the dialog processor user guide.

NOTE:

If you encounter system errors when keying in SC JC$BLD, key in RV JC$BLD and press   ←
XMIT. A short paragraph explaining RUN libraries is then displayed. After reading the paragraph, press XMIT again. The following question is then displayed: DO YOU WANT TO SAVE RUN LIBRARIES? (Y OR N). Key in Y so that you'll be able to enter the SC JC$BLD command without encountering any errors in the future.

The job control dialog introduces the concept of job control and (if you're building a control stream) presents job control statements in the form of menu items from which you choose the statements you want. If you need a dialog concept or particular statement explained, you can ask for help – by keying in HELP or a choice that generates HELP screens. HELP screens explain the choice or statement parameters to you. When you make a valid choice, the dialog resumes at the point where it was interrupted. The HELP screen facility of the job control dialog can be used selectively (statement-by-statement) so that you receive detailed explanations only when you need them. More experienced users, then, can execute the dialog session quickly while still being constrained to build syntactically correct statements. Figure 2-1 presents an overview of the process of using the job control dialog to build a control stream or user jproc.

STEP 1

Key in the SC JC$BLD
command to initiate a
job control dialog session.

SC JC$BLD...
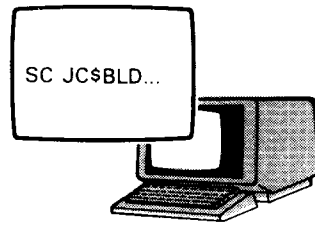
STEP 2

The dialog processor is
activated and the job
control dialog file is opened
in response to the command ...
begin executing the dialog.

JOB CONTROL DIALOG → DIALOG PROCESSOR
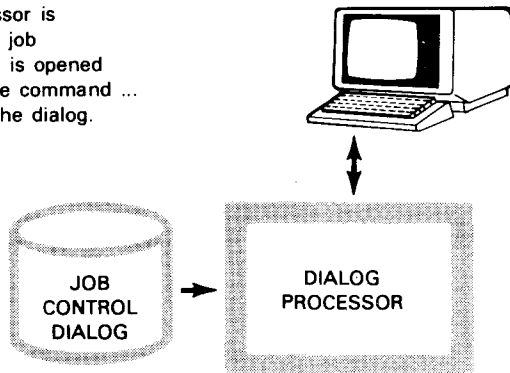
STEP 3

The dialog processor routes
your dialog responses to the
system program JC$BLD.

JOB CONTROL DIALOG → DIALOG PROCESSOR → JC$BLD
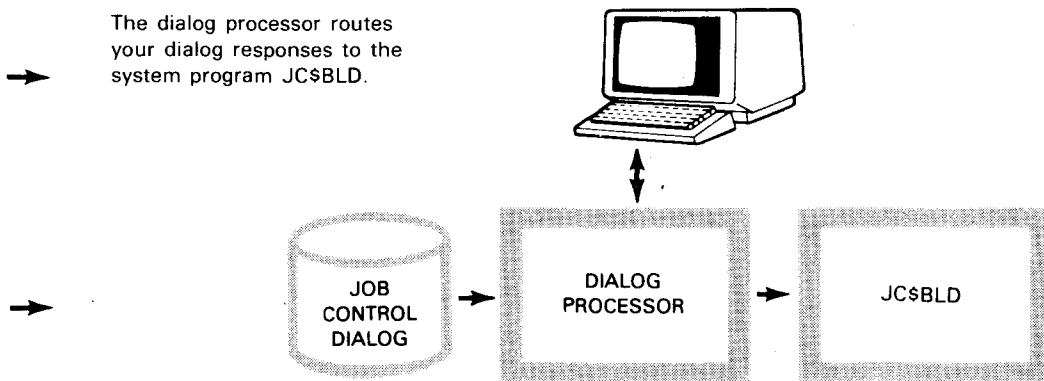
STEP 4

JC$BLD uses your responses
to the dialog to build a job
control stream or user jproc
and stores it in YJCS or your
your own file.

DIALOG RESPONSES

JC$BLD → YJCS
OR
→ YOUR OWN FILE
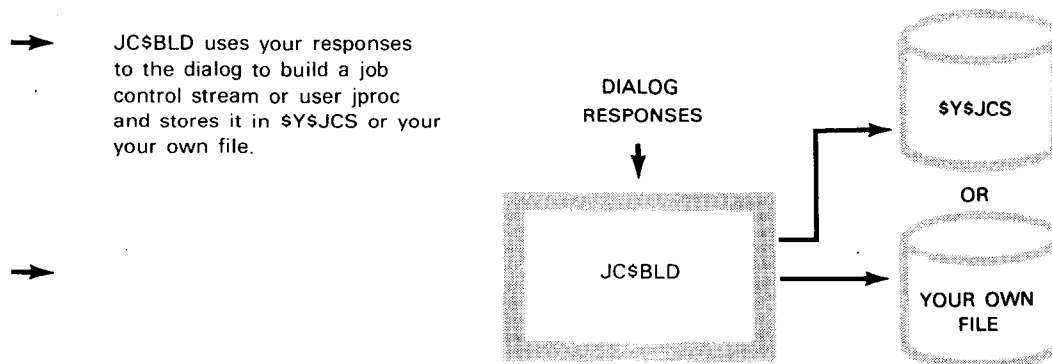
Figure 2—1. Using the Job Control Dialog to Build a Control Stream or User Jproc

2.1.1. Building a Control Stream

Let's begin a sample job control dialog session. First, you perform the system LOGON procedures described in the workstation user guide. Then, you key in SC JC$BLD and its associated parameters. The first dialog screen looks like this:

```
                    DIALOG FOR JOB CONTROL

PROGRAM=
THIS DIALOG  PREPARES A JOB CONTROL STREAM OR PROCEDURE
(JPROC).  FOR AN EXPLANATION OF THE DIALOG  PROCESS, ENTER
'HELP' IN THE SPACE PROVIDED. HELP
```

If you key in HELP, these screens are displayed:

```
    THE DIALOG FOR JOB CONTROL IS A METHOD OF CONSTRUCTING
    JOB CONTROL STREAMS AND PROCEDURES (JPROCS) USING COMPUTER
    ASSISTANCE. PROMPTING FOR DATA ENTRY OR SELECTING FROM
    AMONG AVAILABLE OPTIONS IS ALWAYS PROVIDED, AND YOU CAN
    ASK FOR MORE DETAILED EXPLANATIONS OF STATEMENTS,
    PARAMETERS, AND OPTIONS. AFTER A STATEMENT IS COMPLETED,
    THE IMAGE BUILT BY THE COMPUTER AS A RESULT OF YOUR CHOICES
    IS DISPLAYED ON THE WORKSTATION SCREEN. YOU MAY ACCEPT IT
    FOR OUTPUT, CORRECT IT, OR REJECT IT ALTOGETHER.
```

NOTE:

To proceed from one screen to the next, you usually press the transmit key. Whenever necessary, a note will appear at the bottom of the screen reminding you to do this.

```
THE JOB CONTROL SETS ARE FORMED BY MAKING SELECTIONS FROM

MENUS OF AVAILABLE OPTIONS, AND ENTERING SOME TYPES OF

DATA DIRECTLY. THIS ALLOWS YOU AS MUCH FREEDOM IN YOUR JOB

CONTROL AS OTHER MEDIA, BUT AT THE SAME TIME PROVIDES

A STRUCTURE TO JOB CONTROL CREATION WHICH HELPS TO

PREVENT MANY COMMON ERRORS. REMEMBER, HOWEVER, THAT THE

DIALOG DOES NOT RECOGNIZE THE SAME JOB CONTROL ERRORS AS

THE RUN PROCESSOR. DIALOG ERROR CHECKING IS LIMITED TO

DIALOG OPERATION ERRORS, AND DATA TARGET MISMATCHES

(SUCH AS TRYING TO PUT ALPHABETIC DATA IN A STRICTLY

NUMBER FIELD).
```

The next screen asks what type of module you want to build:

```
JOB CONTROL MODULE TYPES


USE THIS MENU TO SELECT THE TYPE OF MODULE TO BE PREPARED:

1.  JOB CONTROL STREAM

2.  USER WRITTEN JOB CONTROL PROCEDURE (JPROC)

3.  HELP

SELECT ITEM BY ENTERING NUMBER. ▶ _ _
```

If you ask for HELP, these screens are displayed:

```
IN ORDER TO EXECUTE ANY JOB, IT IS NECESSARY TO CONVEY TO
THE COMPUTER EXACTLY WHAT YOU WANT TO DO, AND WHAT RESOURCES
(PRINTER, READER, DISKS, ETC) ARE NEEDED. THIS IS ACCOMPLISHED
THROUGH THE USE OF JOB CONTROL. THERE ARE TWO TYPES OF JOB
CONTROL MODULES. THE COLLECTION OF JOB CONTROL STATEMENTS USED
TO RUN A JOB IS CALLED A JOB CONTROL STREAM, SOMETIMES REFERRED
TO AS THE JOB STREAM OR CONTROL STREAM. IN IT, THERE MAY BE JOB
CONTROL STATEMENTS, CALLS TO SYSTEM SUPPLIED PROCEDURES, AND THE
SECOND TYPE OF MODULE — USER-WRITTEN PROCEDURES (JPROCS).
```

```
JOB CONTROL PROCEDURES HAVE TWO PARTS — THE DEFINITION
AND THE CALL. THE DEFINITION IS THE JPROC MODULE CREATED
BY THE DIALOG. THE CALL IS A STATEMENT IN THE CONTROL
STREAM WHICH HAS THE JPROC NAME AS THE COMMAND, AND
PROVIDES ANY NECESSARY PARAMETERS. THE JPROC CALL IS USED
AS AN ABBREVIATION TO PREVENT CODING THE DEFINITION MANY
TIMES. WHEN THE CONTROL STREAM IS PROCESSED, EACH CALL IS
REPLACED BY THE APPROPRIATE DEFINITION WHICH HAS BEEN PUT
AT THE BEGINNING OF THE STREAM OR STORED IN A SYSTEM FILE
($Y$JCS). THE RESULT IS THE SAME AS IF THE DEFINITION HAD
BEEN CODED INSTEAD OF THE CALL.
```

Once again, you're asked what type of module you want to build.

```
JOB CONTROL MODULE TYPES


USE THIS MENU TO SELECT THE TYPE OF MODULE TO BE PREPARED

1.   JOB CONTROL STREAM

2.   USER WRITTEN JOB CONTROL PROCEDURE (JPROC)

3.   HELP

SELECT ITEM BY ENTERING NUMBER. ▶ _ _
```

You can ask that HELP screens explaining the choices be displayed again (by keying in 3), but let's assume you want to build a control stream. The next screen displayed is the JOB control statement screen:

```
STATEMENT:   JOB

FORMAT:   //SYMBOL JOB JOBNAME,PRI,MINSTORE,MAXSTORE,TASKS,
          TIME,OPTIONS,ACCT,BUFFERS,LOG,HDR


FUNCTION:   THIS STATEMENT IDENTIFIES A JOB AND INDICATES
            THE BEGINNING OF CONTROL INFORMATION FOR THE
            JOB.  THE SAME NAME IS GIVEN TO THE JOB'S RUN
            FILE ($Y$RUN).


IF YOU WILL NEED HELP WITH THIS STATEMENT, ENTER HELP.
```

What if you didn't need HELP screens? The job control dialog screens vary according to the responses you make to the dialog. The initial screen is the same:

```
                    DIALOG FOR JOB CONTROL



        THIS DIALOG PREPARES A JOB CONTROL STREAM OR PROCEDURE

        (JPROC).  FOR AN EXPLANATION OF THE DIALOG PROCESS, ENTER

        'HELP' IN THE SPACE PROVIDED. ____
```

Because you don't need HELP screens to explain the dialog process, simply press the transmit key to display the next screen. The next screen displayed is:

```
        JOB CONTROL MODULE TYPES:
        USE THIS MENU TO SELECT THE TYPE OF MODULE TO BE PREPARED:
        1.   JOB CONTROL STREAM
        2.   USER WRITTEN JOB CONTROL PROCEDURE (JPROC)
        3.   HELP
        1
```

You key in 1, indicating that a job control stream is being prepared. The next screen displayed (since HELP screens weren't requested) is the JOB control statement screen:

```
        STATEMENT:  JOB
        FORMAT:   //SYMBOL JOB JOBNAME,PRI,MINSTORE,MAXSTORE,TASKS,
                    TIME,OPTIONS,ACCT,BUFFERS,LOG,HDR


        FUNCTION:   THIS STATEMENT IDENTIFIES A JOB AND INDICATES
                    THE BEGINNING OF CONTROL INFORMATION FOR THE
                    JOB. THE SAME NAME IS GIVEN TO THE JOB'S RUN
                    FILE ($Y$RUN).


        IF YOU WILL NEED HELP WITH THIS STATEMENT, ENTER 'HELP'.
```

As you can see, there is a big difference in the path the job control dialog takes, depending on your responses to the dialog.

Let's take the dialog one step further. If you key in HELP in response to the JOB statement screen, each parameter of the JOB statement is explained. If HELP is not requested, you are simply asked to key in the parametric values, without benefit of prompting screens. When the JOB statement is built, it is displayed and you have a final chance to change the parameters of the statement, with or without HELP screens, or accept the statement as it appears. When the JOB statement is accepted, the next screen presented is the job control statement master menu.

```
               JOB CONTROL STATEMENT MASTER MENU


      1.  ALTER    11.  EXEC     21.  MTC      31.  RUN/RV   41.  /*
      2.  ALTJCS   12.  EXT      22.  NOP      32.  SCR      42.  /&
      3.  CAT      13.  FREE     23.  OPR      33.  SET      43.  SYSTEM
      4.  CC       14.  GBL      24.  OPTION   34.  SFT           JPROCS
      5.  CR       15.  GO       25.  PARAM    35.  SKIP     44.  GENERAL
      6.  DATA     16.  IF       26   PAUSE    36.  SPL           ENTRY
      7.  DECAT    17.  JNOTE    27.  QGBL     37.  UID      45.  END SESSION
      8.  DST      18.  JSET     28.  QUAL     38.  USE      46.  HELP
      9.  DVC      19.  LBL      29.  REN      39.  VOL
     10.  EQU      20.  LFD      30.  RST      40.  /$
```

The rest of the job control dialog works in the same way as for the initial module-type choice and the JOB statement screens. Each statement you choose from the master menu is displayed and you are asked if you need help to build it. If you do, HELP screens are displayed that explain the parameters of each statement.

NOTE:

The DD, LCB, and VFB job control statements are not provided on the job control statement master menu. To include these statements in your job control stream, make the GENERAL ENTRY menu selection (44), then enter the statement and its parameters in the space provided. For the DD, LCB, and VFB job control statement formats and descriptions, see the job control user guide, UP-8065 (current version).

The control stream you create is stored (by default) in YJCS; however, the job control dialog also gives you the opportunity to specify your own permanent library file for storing the control stream. A printed summary of the dialog session, organized by sequentially-numbered paragraphs, is produced by the dialog processor. The default logical unit number of the printer file (printed summary) output is 20 – any printer. You can accept this default or, during the dialog session, provide a specific printer's logical unit number. Table 2-1 lists the OS/3 logical unit numbers for printers.

Table 2—1. OS/3 Logical Unit Numbers for Printers

Logical Unit Number	Device Type
7	9200/9300 printer
16, 17	Spare
18, 19	0789 printer
20, 21	Any printer
22, 23	0773/0778 printer, no optional features
24, 25	0776 printer, no optional features
26, 27	0768 printer, no optional features
28, 29	0770 printer, no optional features

A complete OS/3 logical unit number table is presented in the job control user guide.

2.1.2. Building a User Jproc

The dialog for creating a jproc guides you through the process of defining your jproc and building the job control statements and system jprocs you want to include in the body of the jproc definition.

The procedure for initiating the dialog is the same as for building a job control stream: perform the system LOGON procedures and key in SC JC$BLD.

When the job control dialog asks you whether you're building a job control stream or user jproc, key in the choice for user jproc. The dialog then presents menus for:

■ Beginning the jproc (PROC, NAME)

■ Choosing job control statements

■ Choosing system jprocs

■ Ending the jproc (END)

As is the case when you're building a job control stream, these menus generate other menus based on your responses to the dialog.

You can request HELP screens at any point in the dialog where you need choices or parameters explained. After the HELP screens are displayed and you make a valid choice, the dialog returns to the point where it was interrupted.

JC$BLD uses your dialog responses to create a jproc, which is stored in YJCS or your own library file.

NOTE:

If you store a jproc in your own (alternate) library file instead of YJCS, you must include the ALTJCS job control statement in any subsequent job control stream that calls the jproc. ALTJCS identifies the jproc and applies only to jprocs.

2.1.3. Entering Embedded Data from a Workstation

You can include embedded data with the control stream you create from a workstation just as you can in a batch environment.

To enter embedded data from a workstation, first choose the /$ (start-of-data) statement from the job control statement master menu. Then, when the master menu is redisplayed, make the GENERAL ENTRY selection (44). Once this is done, you'll be able to enter your embedded data. When all embedded data is entered and the master menu is presented again, choose the /* (end-of-data) job control statement.

If you plan to enter dialog specification language (DSL) source code as embedded data from the workstation, a special situation arises because the characters that denote the start of a DSL comment are the same as the end-of-data job control statement (/*). It's necessary, then, to substitute another set of characters for the end-of-data job control statement. You do this through the OPTION job control statement.

When the OPTION statement menu is displayed at the workstation screen, choose an OPTION EOD statement. The format is OPTION EOD=xx. The first character you select must be a slash (/). The second character can be anything but a slash (/), an asterisk (*), an ampersand (&), or a currency symbol ($). Let's say you choose /Z. Then, when the end-of-data statement is displayed as part of the job control dialog menu, you choose GENERAL ENTRY and key in your substitute characters; /Z in this case. The control stream you create, then, will include these job control statements:

```
        .
        .
        .

// OPTION EOD=/Z

        .
        .
        .
/$          (start of data)

        .       (DSL source code)
        .
/Z          (end of data)
        .
        .
        .
```

You key in your DSL source code when the dialog requests it. By substituting different characters for the end-of-data job control statement, you avoid any conflict with the DSL start-of-comment delimiter.

2.2. CHANGING DIALOG RESPONSES

Once you build a control stream or jproc from a workstation, you may be able to use it for other jobs by making only a few changes to it or, you may discover that you need to correct it. Rather than building a new control stream or jproc from scratch to incorporate the changes you want, you can use the audit version of the dialog processor to change or edit the responses you made in a previous job control dialog session. The audit version of the dialog processor outputs an audit file containing a complete record of your dialog responses; or, it accepts as input an existing audit file of your responses to a previous dialog, or both. An existing audit file used as input is considered an old audit file. The audit file produced as output of the current dialog session is considered a new audit file.

You begin a dialog session, which uses the audit version of the dialog processor, by performing the system LOGON procedures and keying in RV JC$BLD. When you identify a new and/or old audit file (by volume serial number and file label) during the resulting dialog session, the system loads the audit version of the dialog processor.

NOTE:

Old and new audit file names cannot be the same when responding to JC$BLD queries.

The audit version of the dialog processor (Figure 2-2) also outputs a printed summary of a dialog session that is used as a guide to changing dialog responses in a subsequent session. The summary is organized by sequentially-numbered paragraphs. When you use the audit file as input to the dialog processor in a subsequent session, the job control dialog asks you to enter the numbers of the paragraphs you want to change. The summary lists these paragraph numbers.
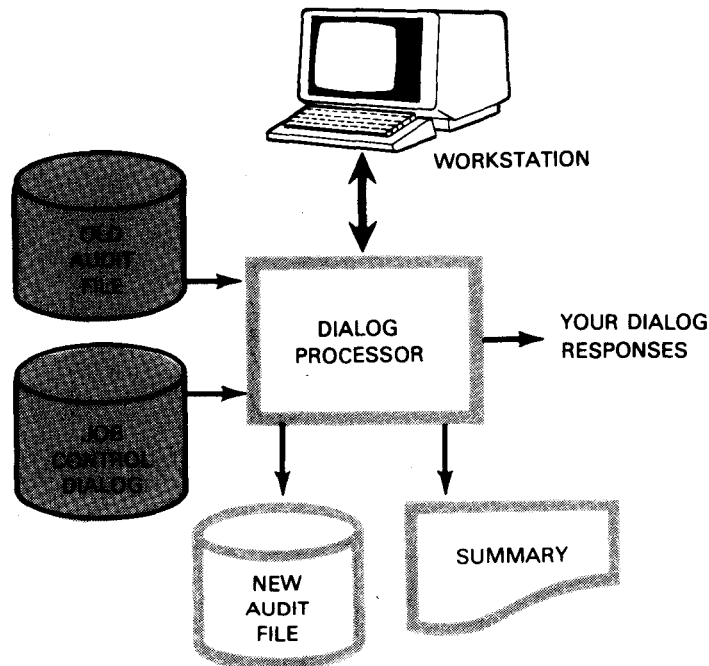

Figure 2—2. Audit Version of the Dialog Processor

NOTE:

Audit files must be previously allocated MIRAM files.

The audit version of the dialog processor allows you to transact the job control dialog quickly and create a "new" control stream or user jproc by changing only the responses that need to be changed. Unchanged responses are automatically routed from the old audit file by the dialog processor to JC$BUILD – without your intervention. During the same session, you enter your new responses to the job control dialog. You can also produce a new audit file (if you've specified it in the build command) that contains a mix of responses from the old audit file and responses entered during the current session. This audit file can then be used as input to the dialog processor in a subsequent session.

NOTE:

Only control streams and user jprocs created using the job control dialog can be changed in a subsequent dialog session.

Suppose you build a control stream for a job that runs nearly every day with only a few changes to the control stream. Perhaps you want disk and print output on some days, and disk output only on other days. You first build the control stream on Monday, specifying that a new audit file and a printed summary of the session be produced. You use the audit file as input to Tuesday's dialog session and use the summary report as your guide to changing the appropriate dialog responses. Figure 2-3 traces the process of changing your dialog responses in a subsequent session.

The dialog processor user guide has more information about using the audit version of the dialog processor, including information about breaking off a session and continuing it at a later time – without losing your changed dialog responses.

ON MONDAY, you create a job control stream and output a new audit file (SESSION1) that contains your responses to the job control dialog.

WORKSTATION

DIALOG PROCESSOR

DIALOG RESPONSES

JC$BUILD

JOB CONTROL STREAM

JOB CONTROL DIALOG

$Y4JCS

SESSION1 (NEW AUDIT FILE)

SUMMARY
PARA 1
PARA 2

SESSION1 IS SPECIFIED AS THE OLD AUDIT FILE FOR TUESDAY'S SESSION

SESSION1 (OLD AUDIT FILE)

ON TUESDAY, you create a new control stream, using SESSION1 and the job control dialog as input to the dialog processor. You change only those responses that need to be changed, using Monday's printed summary as a guide. Unchanged responses are automatically routed from SESSION1 (old audit file) to JC$BUILD. In addition, you create a new audit file (SESSION2) and a printed summary of Tuesday's session, which can be used as input to a subsequent dialog session . . .

JOB CONTROL DIALOG

DIALOG PROCESSOR

responses taken from SESSION1; data entered and changes made in response to the job control dialog

JC$BUILD

JOB CONTROL STREAM

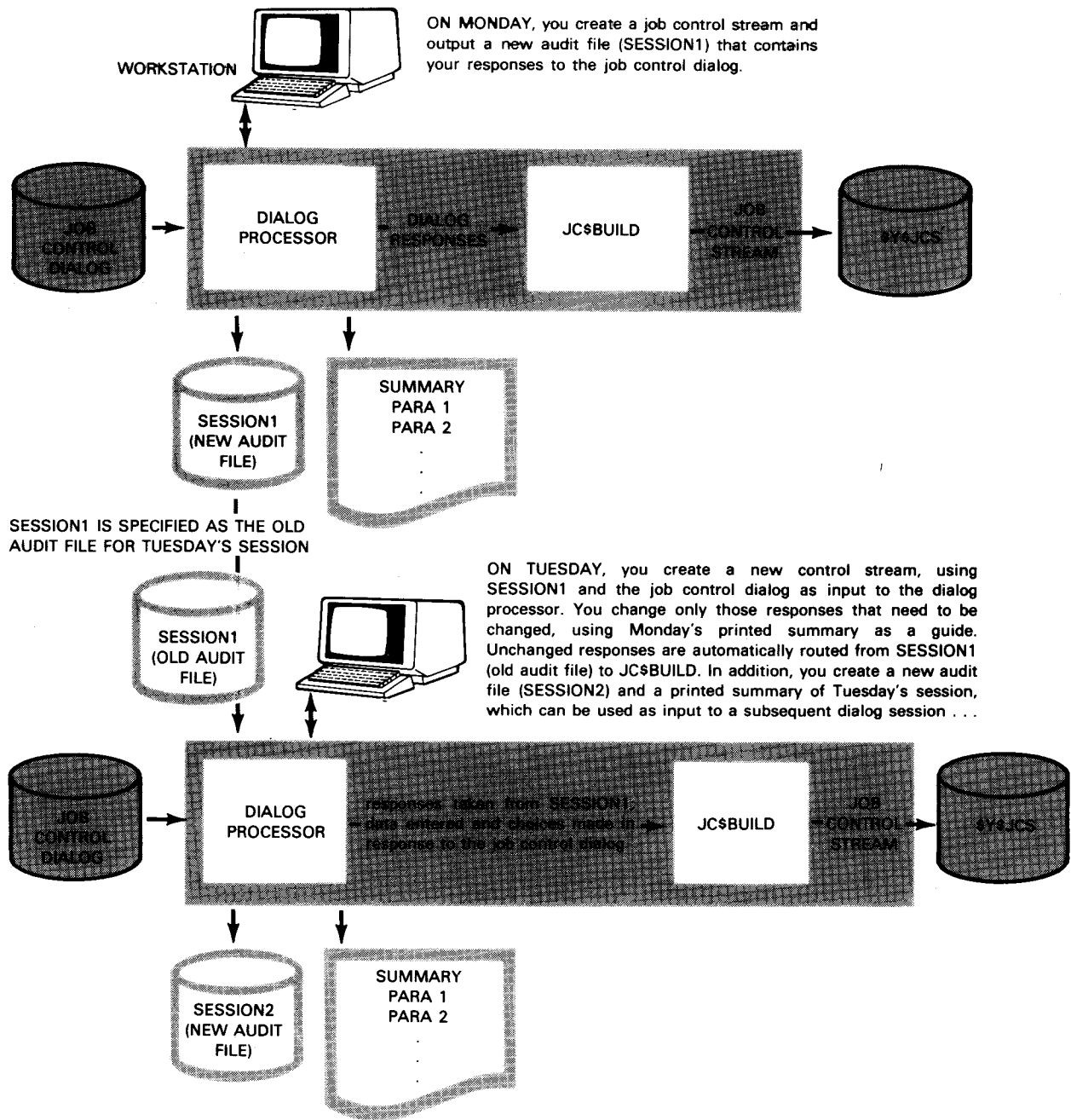$Y4JCS

SESSION2 (NEW AUDIT FILE)

SUMMARY
PARA 1
PARA 2

Figure 2—3. Changing Your Dialog Responses

3. Building, Storing, and Running Control Streams from a Workstation

This is a good point in our discussion to bring together the processes of building, storing, and running a control stream from a workstation. We've already discussed how you build a control stream from a workstation – by executing the job control dialog, which is initiated by the SC JC$BLD command.

Your control stream is stored automatically by JC$BLD in the system job control stream file, YJCS, or an alternate file you specify through the job control dialog.

Once your control stream is stored, it can be run by keying in the RUN or RV workstation command.

With interactive job control you have control over your job from control stream preparation to job execution. Figure 3-1 illustrates these processes. (You can, of course, build and store user jprocs from the workstation. They are used when a control stream that references them is run.)
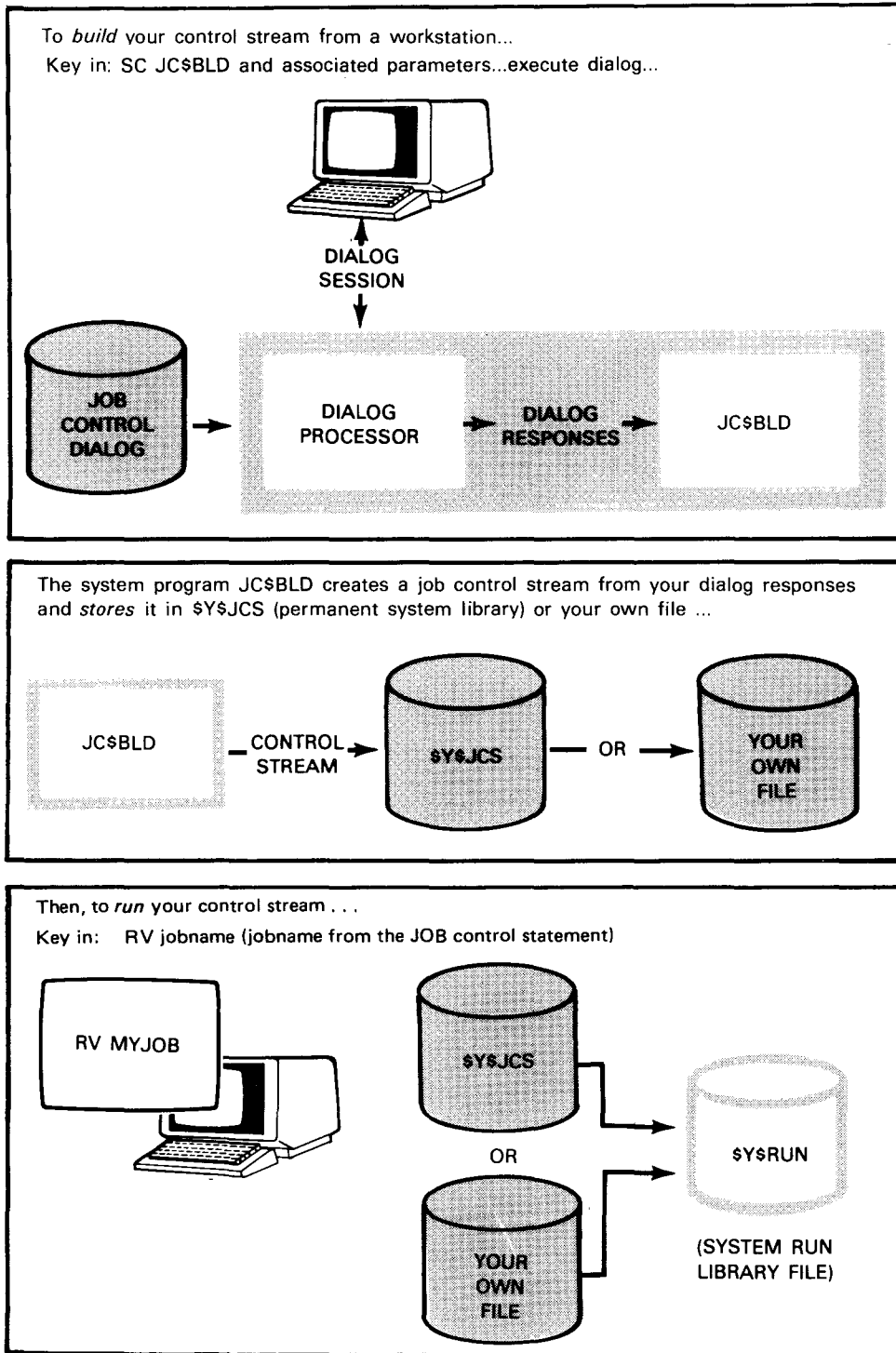
To *build* your control stream from a workstation...
Key in: SC JC$BLD and associated parameters...execute dialog...

DIALOG
SESSION

JOB
CONTROL
DIALOG

DIALOG
PROCESSOR

DIALOG
RESPONSES

JC$BLD

The system program JC$BLD creates a job control stream from your dialog responses and *stores* it in YJCS (permanent system library) or your own file ...

JC$BLD

CONTROL
STREAM

YJCS

OR

YOUR
OWN
FILE

Then, to *run* your control stream . . .

Key in: RV jobname (jobname from the JOB control statement)

RV MYJOB

YJCS

OR

YOUR
OWN
FILE

YRUN

(SYSTEM RUN
LIBRARY FILE)

Figure 3—1. Building, Storing, and Running Control Streams from a Workstation

4. Interactive Communication and Job Control

4.1. GENERAL

In 2.1 you saw the job control statement master menu. When a statement is selected from the menu, its format is displayed along with a description of its function. Additional (HELP) screens further explaining the statement's parameters are displayed upon request so that the job control dialog is actually a job control manual at your fingertips.

Even though this makes detailed, written explanation of each statement unnecessary, there are certain statements that we'll briefly explain in the following paragraphs because they are directly concerned with interactive processing. Each involves communication with the workstation user or the system console operator. They are:

■   QGBL – By communicating with the workstation user through screen displays, this statement allows you to change the value of run-time variable symbols from a workstation.

■   OPTION QUERY – By communicating with the workstation user through screen displays, this statement allows you to change control stream execution at run-time.

■   OPR, PAUSE, and JNOTE – These statements allow you to send messages to the system operator or specific workstations.

NOTE:

Since these statements can also be included in a control stream created in a batch environment, they are also described in detail in the job control user guide, UP-8065 (current version).

4.2. SUBMITTING VALUES FOR RUN-TIME SYMBOLS FROM A WORKSTATION (QGBL)

The QGBL job control statement allows you to submit values for a global set symbol at run time. To use this facility, create a job control stream that contains the QGBL statement. Then, when you run the control stream, the global set symbol you've created is displayed at the workstation. The screen display asks that you provide a value for the symbol. A null response indicates that the default value you established when you created the global is valid.

If you build a job control stream that includes these statements:

```
// JOB MYJOB
// QGBL DVC=20
// DVC &DVC
// LFD PRNTR
    .
    .
    .
/&
```

and then run the control stream (RV MYJOB), you will be asked, through a workstation screen display, if you want to submit a value for the DVC statement. You could, for example, assign a specific printer by submitting a value of 22. If the global symbol is given a value through the RV workstation command, a GBL statement in the control stream, or a JSET statement in the control stream, you won't be asked to submit a value at the workstation, even though the control stream includes the QGBL statement.

The job control user guide contains more information about global set symbols.

4.3. CHANGING CONTROL STREAM EXECUTION FROM A WORKSTATION (OPTION QUERY)

The OPTION QUERY job control statement allows you to change control stream execution by dynamically skipping parts of the control stream at run time. To use this facility, specify an OPTION QUERY job control statement when you create your control stream. Then, when you run the control stream (key in RV jobname), and the OPTION QUERY statement is processed, the following messages are displayed at the workstation screen:

```
JC 36 ENTER SKIP PARAMETER (DISPLAY,CANC,STEP=,LABEL=,,OFF,NONE)
JC 37 UPSI= XXXXXXXX QUERY LABEL= YYYYYYYY
```

If you enter a null response to the message, the system assumes you want to proceed without a skip.

The type of skip you want is specified by keying in one of the following options:

Option	Meaning
NONE	Discontinue this function in the job step
CANC	Cancels the job
STEP=	Resume processing at the specified job step (program name)
LABEL=	Resume processing at the label specified on the NOP QUERY job control statement

Option	Meaning
OFF	Discontinue this function in the job
DISPLAY	Display all labels and job step names in the control stream. Step names are preceded by an asterisk (*) to distinguish them from labels.
X	UPSI setting
Y	Label of QUERY job control statement

The OPTION QUERY and NOP QUERY job control statements are also discussed in the OS/3 job control user guide.

4.4. SENDING MESSAGES TO A WORKSTATION OR SYSTEM OPERATOR (OPR, PAUSE, JNOTE)

You can send a message to the system operator, specific workstations, or the master workstation with the OPR job control statement. The text of the message you specify is displayed at interstep processing time (between job steps). You might, for example, want to tell the operator that an error is going to occur but that the job is to continue processing by specifying the following, or a similar message:

```
AN ERROR WILL OCCUR-DO NOT CANCEL JOB
```

The PAUSE job control statement also allows you to communicate with the system console operator, specific workstations, or the master workstation; however, unlike OPR, it also causes the processing of the job to stop until an appropriate response to the message is received. Suppose you want the operator to check a job's printer listing for errors before the job is run. You could specify the following message:

```
CHECK FOR ERROR-IF NONE, CONTINUE, OTHERWISE CANCEL
```

Because PAUSE suspends all job control activity for the control stream in which it appears, the specified message is not displayed until immediately before execution of the program regardless of the statement's position in the job control stream.

The JNOTE job control statement is also used to communicate with the system console operator or a workstation user, but it is acted upon by the run processor. This allows you to send a message before the job is even scheduled. You can for example, ask the operator to terminate the run processor if the particular conditions you specify in the message exist.

5. Job Control and the Workstation

5.1. GENERAL

The workstation is an important part of the interactive environment, and as you would expect, there are certain job control statements specifically related to the workstation. The statements that we'll briefly discuss in the following paragraphs are:

- USE (SFS, DP) – This statement is specified in the workstation's device assignment set whenever you are using screen format services (SFS), or the dialog processor (DP), with your application program.

- UID – This statement is specified in the workstation's device assignment set whenever you want to assign particular workstations to a job.

- DVC – This statement is always specified in the workstation's device assignment set and may additionally be used to assign multiple workstations to the same file.

- FREE – This statement is used to release a workstation from a job.

- OPTION (MASTER=user-id, MASTER=user-id (EXEC), ORIGINATOR=user-id) – These   ◄ OPTION statement features are specified when you want to reassign a job's master workstation.

NOTE:

Since these statements can also be included in a control stream created in a batch environment, they are described in detail in the job control user guide, UP-8065 (current version).

5.2. CALLING USER DIALOGS AND SCREEN FORMAT SERVICES (USE)

The USE statement is part of the device assignment set for a workstation and must be specified between the DVC and LFD statements for the workstation. USE is specified in the control stream for an application program that calls either the dialog processor (USE DP) or screen format services (USE SFS). Any files referenced in the USE statement must have been previously identified in the control stream with device assignment sets (with the exception of the system format file YFMT).

Let's look at a sample application for the USE DP statement. You have a program, PERSNL, that uses data submitted through a dialog session. The dialog has been written to solicit information about newly hired employees; it's named DIALOG1.

You decide to create a control stream that executes PERSNL, which in turn calls DIALOG1. Because you're creating a control stream for a program that calls the dialog processor, you need to include the USE DP job control statement in the control stream you create.

First, you must log on, and then key in:

```
SC JC$BLD
```

You execute the job control dialog and create a control stream that includes these job control statements:

```
// JOB NEWHIRE
        .
        .
        .

// DVC 51          ⎫
// VOL DSKØ2        ⎬ Identifies the dialog file
// LBL DSKDLGFIL    ⎪
// LFD DIALOG1      ⎭
        .
        .
        .

// DVC 2ØØ      ⎫ Identifies the workstation file
// USE DP        ⎬ (DIALOG1) needed by the
// LFD WKSTATN   ⎭ application program
        .
        .
        .

// EXEC PERSNL
/&
```

This control stream is routed to the system spool file and stored in YJCS. When you are ready to execute the application program, key in:

```
RV NEWHIRE
```

(and any additional parameters of the RV command required to run the job in your environment).

RV NEWHIRE executes your control stream, which executes PERSNL. PERSNL contains an instruction to open WKSTATN which, when processed, causes DIALOG1 to execute at the workstation.

Let's examine a sample application for the USE SFS job control statement. You have an application program (PAYPROG) that calls a screen format file (FORMAT) and uses the data entered in response to the screen format to produce a payroll report. You must first create a control stream that executes PAYPROG. Key in:

```
SC JCSBLD
```

The control stream you create includes these job control statements:

```
// JOB PAYMSTR
        .
        .
        .
// DVC 50           ⎫
// VOL ABC          ⎬ Identifies the screen
// LBL DSKFRMATFIL  ⎪ format file
// LFD FORMAT       ⎭
        .
        .
        .
// DVC 200          ⎫ Identifies the workstation
// USE SFS,FORMAT   ⎬ file SFSFILE
// LFD SFSFILE      ⎭
        .
        .
        .
```

This control stream is routed to the system spool file and stored in YJCS (unless you specify an alternate library through the dialog). When you are ready to execute the application program, key in:

```
RV PAYMSTR
```

(and any additional parameters of the RV command required to run the job in your environment).

RV PAYMSTR executes the control stream, which executes PAYPROG. PAYPROG contains an instruction to open SFSFILE which, when processed, causes FORMAT to be displayed at the workstation screen.

NOTES:

1. *The current version of the OS/3 job control user guide contains a complete explanation of how devices (including workstations) are assigned to a job through device assignment sets. The logical unit numbers specified in DVC job control statements are listed in the Standard Logical Unit Number Table in the OS/3 job control user guide.*

2. As you'll see when you select the USE statement from the job control statement
    master menu, there are other parameters that can be specified for both USE SFS and
    USE DP. The job control dialog explains each parameter. For information about the
    dialog processor, as well as the USE DP statement, see the dialog processor user
    guide, UP-8858 (current version). For information about screen format services, see
    the screen format services concepts and facilities manual, UP-8802 (current version).
    The job control user guide, UP-8065 (current version) also explains both statements.

5.3. IDENTIFYING SPECIFIC WORKSTATIONS (UID)

The UID job control statement may be included in the device assignment set for a
workstation and is used to indicate that specific workstations, identified by user-id, should
be automatically connected to a job.

You can assign specific workstations through UID, or the job's master workstation (by
specifying YMAS as a user-id). The user-id is one to six alphanumeric characters in
length. A maximum of eight user-id's may be specified.

If you are building a control stream for a job that needs a specific workstation (identified
by the user-id of JONES1), you would build a device assignment set that looks like this:

```
        .
        .
        .
// DVC 200
// UID JONES1
// LFD WKSTN
        .
        .
        .
```

If you wanted to specify the system master workstation, your device assignment set could
look like this:

```
        .
        .
        .
// DVC 200
// UID $Y$MAS
// LFD WKSTN
        .
        .
        .
```

5.4. ASSIGNING MULTIPLE WORKSTATIONS TO A FILE AND FREEING WORKSTATIONS (DVC, FREE)

Let's suppose you have a job that needs multiple workstations assigned to the same file
(perhaps you want to update a transaction file from several different workstations). You
can use the DVC job control statement to assign a maximum of eight workstations to a
single file and to specify which, if any, of those workstations must be connected before the
job is scheduled.

You can use the FREE job control statement to release a workstation from a job once that workstation is no longer needed.

NOTES:

1. *A workstation specified in the DVC statement is connected to a job through the CONNECT workstation command. A workstation identified on the UID job control statement is automatically connected.*

2. *The DVC and FREE job control statements are also discussed in the job control user guide, UP-8065 (current version).*

5.5. REASSIGNING WORKSTATIONS (OPTION)

When a job is initiated at a workstation the workstation normally has control of that job, regardless of the number of workstations subsequently connected to it. The controlling workstation is designated as the master workstation for the job. You can, however, designate another workstation (or the system console) as a master workstation using one of the following OPTION job control statement features: MASTER=user-id, MASTER=user-id(EXEC), or ORIGINATOR=user-id.

- MASTER=user-id

  Assigns whatever workstation is logged on under the specified user-id as the master workstation. YCON can be specified as a user-id to assign the system console as the master workstation. The assignment is effective while the job is in the job queue; that is, after the run processor finishes reading and expanding the job control stream and it is effective at execution time. If you save a translated stream (OPTION SAVE or OPTION NOSCHED) that includes this option, the option is in effect when the stream is restored (using the SC/SI command).

- MASTER=user-id(EXEC)

  Functions the same as MASTER=user-id, except that the assignment is effective at execution time only.

- ORIGINATOR=user-id

  Functions the same as MASTER=user-id, except that the assignment becomes effective when the job is being acted upon by the run processor.

The OPTION statement and its features are also discussed in the job control user guide, UP-8065 (current version). For more information about the master workstation, see the interactive services commands and facilities user guide/programmer reference, UP-8845 (current version).

6. Workstation Commands Associated with Your Job

This section presents some workstation commands related to job processing, but not part of job control.

Complete formats and descriptions of all the workstation commands are available in the interactive services commands and facilities user guide/programmer reference, UP-8845 (current version).

Command	Function
BEGIN	Reschedules jobs that are inactive as a result of a HOLD command.
BRKPT	Used to breakpoint subfiles created by the spooler. In effect, BRKPT closes subfiles so that the information they contain is available for processing by the output writer.
CANCEL	Causes an immediate halt to the processing of a job or symbiont.
CHANGE	Used to move a job from one scheduling queue to another (e.g., to change a job from NORMAL priority to HIGH priority).
CONNECT	Attaches a workstation (in workstation mode) to a job.
DELETE	Deletes a specific job from a scheduling queue, a queue of jobs, or all jobs in all queues. Log files for deleted jobs can optionally be listed.
DISPLAY	Displays job scheduling queues.
GO	Activates a job suspended by the PAUSE command.
HOLD	Defers the scheduling of all jobs, a specific job, or jobs of a given priority. It is released with the BEGIN command.

Command	Function
OCL/OC	Enables you to run an IBM System/3 job control stream (written in IBM Operations Control Language or OCL) in an OS/3 environment.
PAUSE	Suspends the processing of a job. Job processing is reactivated with the GO command.
RUN/RV	Initiates the running of a prestored control stream or jproc. RUN initiates jobs that need an input reader device (card reader, diskette, spool file); RV initiates jobs that do not need an input reader device.

NOTE:

> The workstation may not be used to initiate (via RUN) a job that requires card input. Card input required for a job that's going to be initiated from a workstation must first be spooled (using the IN command). Then, provided the stored control stream contains a CR statement, RUN can be used to initiate the job.

Command	Function
STATUS	Displays information about the system environment: terminals in use; main storage and buffer space in use; active jobs and symbionts; tape and disk volumes mounted; active programs and utilities, by user id. If you key in only STATUS, your user-id, the time, and the date are displayed.
STOP	Terminates the specified job at the end of the executing job step and provides for the orderly termination of the job.
SC/SI	Initiates the running of a job saved (through the OPTION SAVE or OPTION NOSCHED job control statement) in its *expanded* state. SC is used to initiate jobs that do not require an input reader device (card reader, diskette, spool file), and SI initiates jobs that do need an input reader device. To initiate the job control build command that calls the job control dialog, the SC/SI command takes this form:

```
SC JC$BLD
```

NOTE:

For performing more specific functions, qualifiers are usually added to many of the workstation commands. For example, BEGIN SPL enables you to release held spooled files. For a complete list of the available qualifiers for each command, see the interactive services commands and facilities user guide/programmer reference, UP-8845 (current version).

Glossary

A

audit file
> Optional file produced by the audit version of the dialog processor. It contains a record of your responses during a dialog session. An audit file can be used as input to the dialog processor in a subsequent dialog session. When used as input to the dialog processor, it's identified as an old audit file; when produced as output, it's identified as a new audit file.

audit version of the dialog processor
> The version of the dialog processor that produces an audit file. The audit version of the dialog processor is used if you want to change your responses to the job control dialog in a subsequent session.

D

dialog processor
> An OS/3 system program that manages interactive dialogs, including the job control dialog. The dialog processor displays dialog text at a workstation screen, accepts your responses to the job control dialog, and routes your responses to JC$BUILD. The control stream for an application program that calls the dialog processor must include the USE DP job control statement.

I

interactive processing
> A computing environment in which you exchange information with the operating system through a workstation. Interactive processing may also be conversational, with the workstation user or system console operator engaged in a dialog with the operating system.

J

JC$BLD

System program that uses your responses to the job control dialog to build a job control stream or user jproc.

job control

OS/3 software component that directs job processing.

job control build command

System command that initiates the job control dialog:

```
SC JC$BLD
```

job control dialog

Sperry Univac-supplied dialog that guides you through the process of building a job control stream or a user jproc.

jproc

A job control procedure. A single reference (jproc) in a control stream will, when processed, generate all the job control statements needed to perform a particular procedure. See system jproc and user jproc.

S

SC JC$BLD

Workstation command that initiates the job control dialog.

screen format services

System programs that help you create and manage screen formats. The control stream for an application program that calls screen format services must include the USE SFS job control statement.

summary report

A printed report output by the dialog processor that contains a summary of a dialog session organized by sequentially-numbered paragraphs.

system jproc

Job control procedure supplied by Sperry Univac which, when processed, generates the job control statements needed to perform a particular procedure (executing the COBOL compiler, for example).

U

user-id
Identifier assigned to each interactive user of OS/3; must be specified when the workstation user logs on to the system.

user jproc
Job control procedure defined by the user which, when processed, generates all the job control statements needed to perform a particular function. User jprocs can be created using the job control dialog.

W

workstation
A terminal device with a screen for display of system information, including dialog text, and a keyboard for entry of user input.

Y

YJCS
The OS/3 job control stream and jproc system library file.

Index

SPERRY✦UNIVAC

USER COMMENT SHEET

Your comments concerning this document will be welcomed by Sperry Univac for use in improving subsequent editions.

Please note: This form is not intended to be used as an order blank.

_____

(Document Title)

_____   _____   _____

(Document No.)   *(Revision No.)*   *(Update No.)*

Comments:

From:

_____

(Name of User)

_____

(Business Address)

Fold on dotted lines, and mail. (No postage stamp is necessary if mailed in the U.S.A.)
Thank you for your cooperation

Cut along line.

FOLD

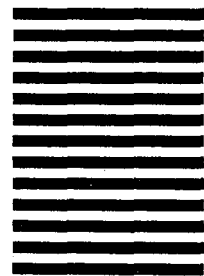# BUSINESS REPLY MAIL

FIRST CLASS   PERMIT NO. 21   BLUE BELL, PA.

POSTAGE WILL BE PAID BY ADDRESSEE

SPERRY UNIVAC

ATTN.: SYSTEMS PUBLICATIONS

P.O. BOX 500
BLUE BELL, PENNSYLVANIA 19424

FOLD

CUT