HPL SOFTWARE LIBRARY

FOR THE 6940B

MODEL 14556A

USER'S MANUAL

HP Part No. 14556-90001
Microfiche Part No. 14556-90002

June, 1979

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

TABLE OF CONTENTS

SECTION I
INTRODUCTION

GENERAL

The Hewlett-Packard program cartridge 14556A provides "friendly" programming access to the I/O functions of the 6940B Multiprogrammer System. The HP 9825A Desktop Computer is used as the controller with either the GPIO interface (Model 98032A-Option 040) or HP-IB (Model 98034A). Programs for both interfaces are recorded on the cartridge.

The 9825A must be equipped with the General I/O, Extended I/O, Advanced Programming, and String Variable ROM's in order to run the programs. When the 6940B is used with the GPIO interface, the select code is normally set to "9" When using the HP-IB interface, the select code is normally set to "7". Procedures for changing the select codes are given in Appendix A. The 6940B Multiprogrammer User's Guide (HP Part No. 59500-90005) is the general reference for system installation, operation, and programming instructions.

CALL SUBPROGRAMS

Programs on the 14556A program cartridge are accessed using a call (cll) statement followed by the name of the subprogram enclosed in single quotes. The following descriptions assume that the reader has knowledge of call subprograms. Call subprogram use is described in the Advanced Programming ROM Manual, HP Part No. 09825-90021.

Syntax

All call subprograms for the 14556A cartridge are of this type:

```
cll 'label' (S0, D0, S1, D1, ----, U)
```

Call labels can either be "long labels" or "short labels." The long labels are more descriptive (e. g., Parallel Output) while the short labels are abbreviations (e. g., PO). The long labels facilitate easier reading of the program but either label will cause the same function to be programmed. Table 1-1 lists the 14556A call subprograms in file number order with corresponding "long" and "short" program labels. Programs for the GPIO interface are recorded on track 0 while HP-IB programs are recorded on track 1 of the cartridge. Both tracks have the same file organization.

Slot Addresses

S0, S1, S2, etc. represent the card slot addresses 400, 401, 402, etc. printed beneath each plug-in card slot (see Figure 1-1). If desired, the numbers 0, 1, 2, etc. can be used instead of 400, 401, 402, etc. For example the following calls perform the same function:

```
cll 'Parallel Output' (401, A, 403, B)
cll 'PO' (1, A, 3, B)
cll 'Parallel Output' (401, A, 3, B)
```

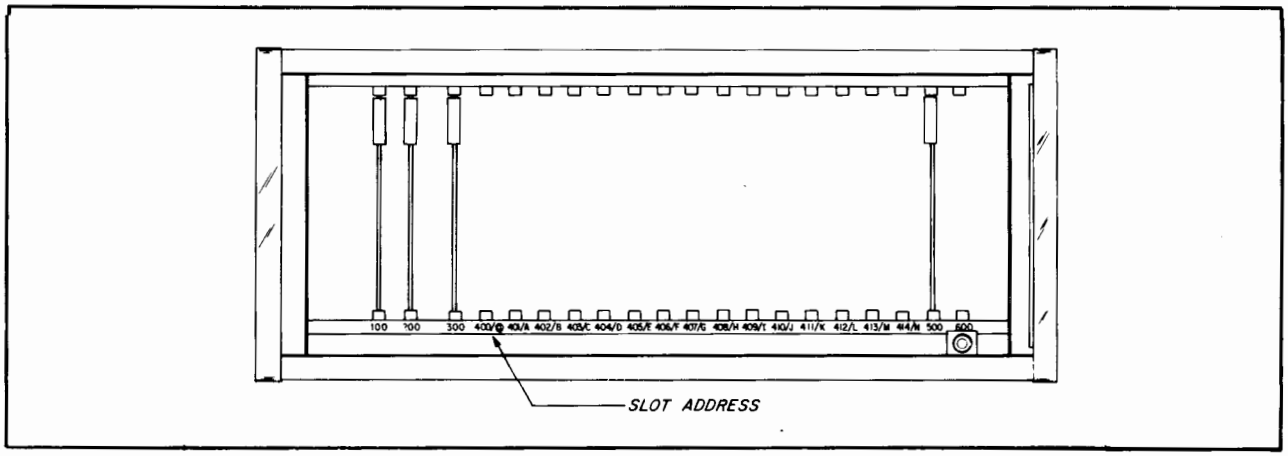


Figure 1-1. Plug-In Card Slots

Data

D0, D1, D2, etc. denote the data that is transferred between the controller and the multiprogrammer. For output transfers, D0, D1, etc. are sent from the controller to the corresponding card (S0, S1, etc.) and can represent variables or can be given as the constants. For example:

Sending Variable Data:
cll 'Parallel Output' (407, A, 402, C, 0)

Sending Constants:
cll 'Parallel Output' (407, 1234, 402, 7777, 0)

For input transfers (multiprogrammer to controller), D0, D1, etc. must be variables. The variables can be simple variables, r-variables, or single elements of any array. The returned value is placed in the proper variable. For example:

cll 'Parallel Input' (407, A, 403, r1, 405. C[3], 2)

In this example, the value from the card in slot 407 is placed in A, the value from the card in slot 403 is placed in r1, and the value from slot 405 is placed in C[3].

Unit No.

The last parameter, U, is optional. This parameter designates the unit (mainframe 0 through 15) containing the cards being called. This means that all cards called in a subroutine must be installed in the same unit. Unit 0 (the 6940B mainframe) is assigned when this parameter is omitted. For example, the following calls perform the same function:

cll 'Parallel Input' (408, A, 401, B, 0)
cll 'Parallel Input' (408, A, 401, B)

Summary

Card Slot addresses and unit numbers are always given in decimal. However, data may be either octal or decimal. Each individual Library routine specifies the proper form for data. All analog Library routines will use decimal data. For example, the Voltage Serial Output (VS) routine uses data in decimal form (e. g., −5.12 for −5.12 volts). Analog input routines(e.g., Single Ended Scan will return the voltage measured in decimal complete with polarity.

Table 1-1. 6940B Software Library for 9825A (14556A Cartridge Organization)

PROGRAM LABELS			PROGRAM LABELS		
FILE NO.	LONG	SHORT	FILE NO.	LONG	SHORT
0	File Lister	—	21	Read Counter	RC
1	Auto Loader	—	22	Timer Pulse	TP
2	Parameter Check	PC	23	Relay Output	RO
3	Simple Output	oo	24	Time Interval	TI
4	Serial Output	SO	25	Frequency Measurement	FM
5	Parallel Output	PO	26	Option 40 Voltage Control	VC
6	Interrupt Output	IO	27	Option 40 Current Control	CC
7	Simple Input	ii	28	Arm	AM
8	Serial Input	SI	29	Serial Poll	SP
9	Parallel Input	PI	30	Disarm	DM
10	Clear Unit	CU	31	Time Out	TO
11	Voltage Serial Output	VS	32	Recycle	RE
12	Voltage Parallel Output	VP	33	Reference Word	RW
13	Current Serial Output	CS	34	T Thermocouple	TT
14	Current Parallel Output	CP	35	J Thermocouple	JT
15	Analog Input	AI	36	E Thermocouple	ET
16	Single Ended Scan	SE	37	R Thermocouple	RT
17	Double Ended Scan	DE	38	S Thermocouple	ST
18	Null File	—	39	Thermocouple Reference	TR
19	Motor Output	MO	40	B Thermocouple	BT
20	Set Counter	SC	41	K Thermocouple	KT

NOTE: Programs for the GPIO interface are recorded on track 0 and the HP-IB interface programs are recorded on track 1.

SECTION II
USING THE SOFTWARE LIBRARY

GENERAL

This section describes the operating instructions for using the 14556A Software Library. Before proceeding with the operating instructions, read the general information provided in Section I and check Table 2-1 (Program Reference Table) to select the Library subprograms that can be used with your particular plug-in card configuration. Table 2-1 provides an insight into how the library works by giving a brief description of each routine and listing the plug-in cards that can be used with each call. Also, before proceeding, make a list of the plug-in card slot addresses. Once the Library routines have been selected and the slot addresses are known, the user can write his program following the instructions outlined in Figure 2-1 and described in the next paragraph.

OPERATING INSTRUCTIONS

The following steps are required to use the 14556A Software Library:

1. Write and record application program (User's program).

   a. Write the application program making use of the selected Library subprograms (see sample program below). Be sure to include an "end" statement at the bottom of the program.
   b. Enter application program into 9825A memory.
   c. Record the application program on a cartridge file. (A blank file in the 14556A Software Library could be used).

Sample Application Program

```
0: "Test Program Track 0; File 50":
1: cll 'Clear Unit'(0)
2: cll 'Serial Output'(401,37,400,7777)
3: cll 'Parallel Input'(404,A)
4: cll 'Dog'
5: cll 'SO'(409,42,0)
6: prt "Data Value in Octal";A
7: end
*18122
```

2. Insert 14556A Software Library cartridge.

   a. Select track 0 for GPIO interface or track 1 for HP-IB interface.
   b. Load the Auto Loader from file 1 of the library.
   c. Press RUN.

3. When the 9825A display shows:

   ```
   Insert User Program Tape
   ```

   a. Remove the 14556A Software Library cartridge.
   b. Insert the tape cartridge prepared in step 1c.
   c. Press CONTINUE.

4. When the 9825A display shows:

   ```
   Track # ?
   ```

Enter the track number where the program prepared in step 1 is stored. Press CONTINUE.

5. When the 9825A display shows:

```
Program FILE no. ?
```

Enter the file number where the program prepared in step 1 is stored. Press CONTINUE.

6. When the 9825A display shows:

```
Insert SOFTWARE LIBRARY
```

a. Remove the tape cartridge prepared in step 1c.
b. Insert the 14556A Software Library cartridge.
c. Press CONTINUE.

7. Check the 9825A printer.

a. If subprograms in the user program are not found in the 14556A Software Library directory, the following printout will occur:

```
Not in Directory
cll '-----'
```

Could be a valid user prepared subprogram label or a call label for the 14556A software library which was typed incorrectly. Note that if the sample user's program (see line no. 4) was used in step 1c, the cll 'Dog' would be printed out.

(1) If call label was not found in 14556A directory because it is a valid user prepared subprogram, refer to Appendix B for procedures explaining how to add these programs.

(2) If call label was not found because the user typed in an incorrect 14556A call label, go back to step 1 and correct your program.

b. After all required subprograms found in the directory and used in the user program are loaded into memory, the following printout will occur:

```
Subroutines from
the Directory
have been loaded
```

and the display will show:

```
Record or Edit?
```

c. At this point the user may choose to record, edit, or run the program stored in the calculator. If it is desired to record the program, proceed to step 8. To edit or run the program, go to step 9.

8. To record program, proceed as follows:

NOTE

The program can be recorded on a previously marked file on the 14556A cartridge or the user's own cartridge. Procedures for marking files are given in the next paragraph.

a. Slide cartridge tab to RECORD position and insert cartridge into 9825A.
b. Enter the word "Record" and press CONTINUE.
c. When the 9825A display shows:

```
What trk?
```

Enter the applicable track number and press CONTINUE.

d. When the 9825A display shows:

```
What File?
```

Enter the File number where the program is to be stored. Press CONTINUE.

e. When the 9825A display shows:

```
Done
```

User's completed program (Auto Loader is automatically removed) has been recorded on specified track and file. Note that the Auto Loader is still part of the program stored in calculator. If the user wishes to run the program stored in the calculator, he must first remove the Auto Loader by executing: del 0,56,*

9. To edit and/or run the program, proceed as follows:

a. Enter the word "Edit" and press CONTINUE. Printer reads:

```
To remove loader
type in and
execute:
   del 0,56,*
```

b. After the Auto Loader is removed by executing the above statement, the user can edit and/or run the program. If desired, additional user prepared subprograms can be added during the edit phase. The completed program can be stored on a previously marked file (on 14556A cartridge or user's own cartridge).

MARKING TAPE FILES

The user first determines how many bytes of program he wishes to record. This can be done by executing a "list-1" from the keyboard when the program is stored in memory. The user then inserts his tape cartridge into the 9825A and finds which file he will use to record the program into. This can be done by a "tlist" which will show which file number is the last one on the tape. The user now executes a "fdfN" where N is the last file on the tape. Now the user executes a "mrk1,B" where B is the number of bytes the user wishes to allow for. Usually, the user will wish to mark a file for more bytes than are in his program to allow for future editing.

ERROR MESSAGES

While the program is running, the slot address, card data, and unit number parameters are checked to determine if they are within limits. The Parameter Check Subprogram (File 2) checks for the following limits:

Parameter	Limits
Slot address	400 to 414, or 0 to 14
Output data	0 to 7777*
Unit number	0 to 15 (0 is assigned if this parameter is omitted)

*Certain other subprograms (e.g., Analog Output Subprograms) check output data parameters for specific decimal value limits (e.g., +10.235). The specific limits are given in Table 2-1 and in the applicable program description provided in Section III.

Parameters that are out of range will cause the program to stop and wait for a new value before continuing. The calculator display shows an "Error Message" that specifies which parameter is out of range.

Slot Address

For an out of range slot address, the display shows:

```
Slot Address p--- out of range
```

odd numbered parameter p1, p3, p5, etc.

To continue the program, type in the correct slot address and assign it to the parameter number specified in the error message. Example, if parameter p3 is out of range:

Type: 403 → p3 and press EXECUTE
Press: CONTINUE

Output Data

For out of range data, the display shows:

```
Data p--- out of range
```

even numbered parameter p2, p4, p6, etc.

To continue the program, type in the correct output data value and assign it to the parameter number specified in the error message. Example, if parameter p6 is out of range:

Type: 3777 → p6 and press EXECUTE
Press: CONTINUE

Unit Number

For an out of range unit number, the display shows:

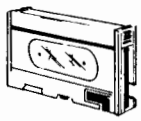
```
Unit no.--- out of range
```

unit number > 15

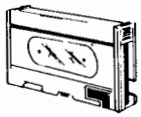
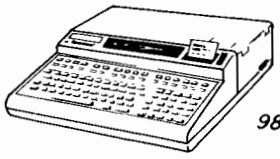
To continue the program, type in the correct unit number and assign it to parameter pp0. Example:

Type: 3 → pp0 and press EXECUTE
Press: CONTINUE

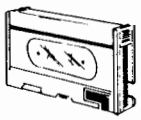
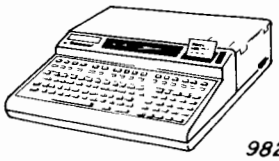
1. 



USER WRITES AND RECORDS HIS PROGRAM

2. 


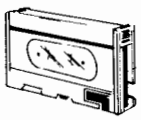
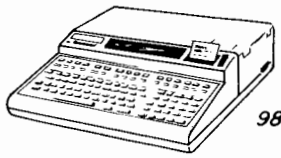
SELECT TRACK Ø (GPIO) OR TRACK 1 (HP-IB)
LOAD AUTO LOADER (FILE 1)
PRESS RUN

3. 


PRESS CONTINUE

4. ENTER TRACK NO. WHERE USER'S PROGRAM IS RECORDED.
PRESS CONTINUE

5. ENTER FILE NO. WHERE USER'S PROGRAM IS RECORDED.
PRESS CONTINUE

6. 


PRESS CONTINUE

7. EDIT, RUN, OR RECORD THE PROGRAM

NOTE: BEFORE EDITING OR RUNNING THE PROGRAM, THE AUTO LOADER MUST BE REMOVED BY EXECUTING:

del Ø, 56, *

Figure 2-1. Using 14556A Software Library

TABLE 2-1. PROGRAM REFERENCE TABLE

Type	Program Label	Plug-in Cards	File No.	Description
Digital Output	Interrupt Output (IO)	69330A Relay Output 69331A Digital Output	6	Sends octal data (0 to 7777$_8$) to the output cards specified. Uses the 9825A's interrupt capability to indicate when data transfers with external devices are completed. See page 3-14.
	Parallel Output (PO)	69330A Relay Output 69331A Digital Output 69433A Relay/Readback	5	Sends octal data (0 to 7777$_8$) to output cards specified. Card outputs are updated simultaneously. See page 3-12.
	Relay Output (RO)	69433A Relay/Readback	23	Changes the state of the desired relay (1 through 12) on the specified card. Data must be in the range of 1 to 12 or −12 to −1. Relays are closed for positive numbers and opened for negative numbers. See page 3-57.
	Serial Output (SO)	69330A Relay Output 69331A Digital Output 69433A Relay/Readback	4	Sends octal data (0 to 7777$_8$) to the output cards specified. Card outputs are updated sequentially. See page 3-10.
	Simple Output (oo)	69332A Open Collector 69380A Breadboard Output	3	Sends octal data (0 to 7777$_8$) to the output card (only one card per call) specified.Does not cause 6940B to wait for flag. See page 3-8.
Analog Output	Option 40 Current Control (CC)	69510A through 69513A Resistance Output	27	Sends desired current output (0 to 1.000 amps) in decimal form (e.g., 0.55 for 0.55A) to the option 40 power supply programming cards specified. Each card has 2 output channels. See page 3-70.
	Current Serial Output (CS)	69370A Current D/A	13	Sends desired output current (0 to 20.475mA) in decimal form (e.g., 4.5 for +4.5mA) to the cards specified. Cards are updated sequentially to their new output currents. See page 3-32.
	Current Parallel Output (CP)	69370A Current D/A	14	Sends desired output current (0 to 20.475mA) in decimal form (e.g., 4.5 for +4.5mA) to the cards specified. The cards are simultaneously updated to their new output currents. See page 3-35.
	Option 40 Voltage Control (VC)	69500A through 69506A Resistance Output	26	Sends desired output voltage (0 to 100.00) in decimal form to the option 40 power supply programming cards specified. See page 3-66.
	Voltage Parallel Output (VP)	69321B Voltage D/A 69322A Quad D/A	12	Sends desired output voltage (−10.240 to +10.235V) in decimal form (e.g., 7.5 for +7.5 volts) to the cards specified. 69322A Quad D/A card outputs are updated when the applicable outputs are addressed (e.g., to address channel 2 of 69322A in slot 3 use address 3.2). The 69321B cards are simultaneously updated to their new output voltages. See page 3-29.

. TABLE 2-1. PROGRAM REFERENCE TABLE (Continued)

Type	Program Label	Plug-in Cards	File No.	Description
Analog Output (cont.)	Voltage Serial Output (VS)	69321B Voltage D/A 69322A Quad D/A	11	Sends desired output voltage (−10.240 to +10.235V) in decimal form (e.g., −6.5 for −6.5 volts) to the cards specified. Card outputs are updated sequentially. Note that a 69322A card has four output channels which can be individually updated. (e.g., to update channel 2 of 69322A in slot 3 use address 3.2). See page 3-26.
Digital Input	Parallel Input (PI)	69431A Digital Input	9	Returns octal data (IRQ bit removed) from the input cards specified. The first card that has data ready is read first, followed by the next ready card, etc. The subprogram will not complete until all cards have been read. See page 3-21.
	Serial Input (SI)	69431A Digital Input	8	Returns octal data (IRQ bit removed) from the input cards specified. Cards are read sequentially. See page 3-19.
	Simple Input (ii)	69430A Isolated Digital Input 69480A Breadboard Output	7	Reads octal data from the input card (only one card per call). The data is returned in octal (IRQ bit not removed). Does not cause the 6940B to wait for flag. See page 3-17.
Analog Input	Analog Input (AI)	69421A Voltage Monitor 69422A High Speed A/D	15	Reads voltage data (−10.240 to +10.235V) from the cards specified. Returned data is in decimal form (e.g., −5.5 for −5.5 volts). Cards are read sequentially. See page 3-38.
	Double Ended Scan (DE)	69421A Voltage Monitor 69422A High Speed A/D 69330A Relay Output 69433A Relay/Readback	17	Scans and reads voltages from multiplexed DC sources. Measured voltages in decimal form are read from 69421A or 69422A cards. The 69330A or 69433A cards are used to scan the dc voltages to be measured. The voltages scanned need not share a single common or ground wire. All readings are stored in r-variables. See page 3-44.
	Single Ended Scan (SE)	69421A Voltage Monitor 69422A High Speed A/D 69330A Relay Output 69433A Relay/Readback	16	Scans and read voltages from multiplexed dc sources. Measured voltages in decimal form are read from 69421A or 69422A cards. The 69330A or 69433A cards are used to scan the dc voltages to be measured. All readings are stored in r-variables. The voltages scanned must share a single common or ground wire. If this is not possible, see file 17, "Double Ended Scan". See page 3-41.
Pulse Counting	Read Counter (RC)	69435A Pulse Counter	21	Reads the specified counter cards. Returned data is in decimal form and is stored in the specified variables. See page 3-52.
	Set Counter (SC)	69435A Pulse Counter	20	Presets specified counters to the decimal count sent as data. Maximum count that card can be set to is 4095. See page 3-50.

TABLE 2-1. PROGRAM REFERENCE TABLE (Continued)

Type	Program Label	Plug-in Cards	File No.	Description
Stepping Motor Control	Motor Output (MO)	69335A Stepping Motor Control	19	Sends signed decimal data to the card specified. Positive numbers will cause stepping motors to rotate in one direction, and negative numbers in the reverse direction. Data greater than ±2047 steps will cause a stop. See page 3-48.
Timing	Frequency Measurement (FM)	69435A Pulse Counter 69600B Timer	25	Returns the decimal value of an external frequency source. The program adjusts the sample interval (from 69600B) until the best resolution is achieved. See page 3-63.
	Time Interval (TI)	69435A Pulse Counter 69601B Frequency Reference	24	Returns the decimal value of an external pulse. The time base (69601B) is assumed to be 100kHz. See page 3-60.
	Timer Pulse (TP)	69600B Timer	22	Sends the time (in seconds) to the card specified. Time interval resolution is 0.001 seconds. This call assumes a jumper wire has been installed between output connector (P2) pins 4 and 11. See page 3-54.
Interrupt Mode	Arm (AM)	69431A Digital Input 69434A Event Sense 69436A Process Interrupt 69600B Timer	28	Arms cards specified to permit an interrupt whenever any card detects an interrupt condition. (Assumes all W6 jumpers have been removed.) After last card is armed, program puts the Multiprogrammer system in interrupt mode and enables 9825A interrupt system before returning. See page 3-73.
	Disarm (DM)	69431A Digital Input 69434A Event Sense 69436A Process Interrupt 69600B Timer	30	Selectively disarms cards specified. Assumes all W6 jumpers have been removed. See page 3-79.
	Recycle (RE)	69436A Process Interrupt 69600B Timer	32	Recycles specified cards, before they are armed by the subprogram 'Arm' (File 28). See page 3-84.
	Reference Word (RW)	69434A Event Sense	33	Reads back the reference words from the desired Event Sense cards and sends new reference words to these cards. See page 3-87.
	Serial Poll (SP)	69431A Digital Input 69434A Event Sense 69436A Process Interrupt 69600B Timer	29	Performs a Serial Poll on cards specified. The data returned is in octal and includes an IRQ bit if the card generated an interrupt. See page 3-76.

TABLE 2-1. PROGRAM REFERENCE TABLE (Continued)

Type	Program Label	Plug-in Cards	File No.	Description
Temperature Measurement	B Thermocouple (BT)	69423A Low Level A/D	40	Reads the temperature in °C from the 69423A card and the applicable thermocouple type. A 69423A card has six input channels that are read individually. For example, to read channel 5 of the 69423A card in slot 2, use address 402.5 or 2.5. See pages 3-92 through 3-118.
	E Thermocouple (ET)		36	
	J Thermocouple (JT)		35	
	K Thermocouple (KT)		41	
	R Thermocouple (RT)		37	
	S Thermocouple (ST)		38	
	T Thermocouple (TT)		34	
	Thermocouple Reference (TR)		39	Reads the temperature in °C of the reference on the 69423A card. See page 3-108.
Special	Clear Unit (CU)	All	10	Sends logic zeroes to clear all cards in unit specified. For example, sending logic zeroes will clear a Relay Output Card (open all 12 relays). See page 3-24.
	Time Out (TO)	All Output Cards	31	Halts system until all output cards complete their operations. See page 3-82.

SECTION III
PROGRAM DESCRIPTIONS

UTILITY PROGRAMS

The following paragraphs contain detailed descriptions of the File Lister and Auto Loader utility programs which are recorded on Files 0 and 1, respectively. Program listings are included with each description.

File Lister (File 0)

The File Lister program is the same for both the GPIO and the HP-IB interfaces and can be used with any 9825A program cartridge. The program starts by loading file zero (from the current track) off the program cartridge in the tape transport. The program from the cartridge is loaded into memory after all program lines in the File Lister. After the program is in memory, the first line of that program is stored in a string variable by lines 10 and 16. The string is then printed on the printer whose select code is represented by the variable P. With P = 16, the printout occurs on the internal strip printer. Null files, binary program files, etc. will not be loaded, but will cause a printout of the file type, thus:

```
----------------

FILE # 40
 0: "B Thermocou
ple":
----------------

FILE # 41
 0: "K Thermocou
ple":
----------------

FILE # 42
NULL FILE
```


The File Lister is most useful with programs containing a label in their first line. Such labels will help the user identify programs. The File Lister enables users to easily update documentation of program cartridges and makes finding a labeled program very easy. Changing variable N in line zero will cause the program to start from a file other than zero.

To use the File Lister:

1. Load file zero from the Software Library.
2. Insert the 9825A cartridge to be listed.
3. Set the 9825A to the desired track by the "trk 0" or the "trk 1" command.
4. Press the 9825A RUN button.

To start at a file other than zero:

1. Load file zero from the Software Library.
2. Insert the 9825A cartridge to be listed.
3. Fetch line zero.
4. Change variable N to one less than the starting file number (e.g., to start at file 6: 5 – N).
5. Store the updated line zero.
6. Set the 9825A to the desired track.
7. Press the 9825A RUN button.

Program Description. This program starts by assigning the value 16 to variable P. Variable P is used to designate the printout device (internal printer). An error linkage is established to label "DoneLister". A buffer and a string variable are also dimensioned. The program then updates the file number variable, N, and finds and identifies that file.

When the file number variable N, is zero, a printout will occur that indicates the current track number. The program will print out file number and the first line of that file until all files have been accessed. Key files, memory files, data files, numeric files, binary files and null files will cause the printout to indicate only file number and the file type.

The key to printing out the first line of each file as line number zero is the use of the list statement to an external device. However, in this case (line 10), the external device is specified by the function "L", line 17. The function "L" defined by line 17 returns the buffer "list" rather than a select code. The line listed can now be examined by examining string variables A$ because the buffer "list" was defined as that string. Use of a non existent format (format seven) deletes extraneous carriage returns and line feeds.

Once the line of program is in a string variable, the program:

1. Finds the line feed.
2. Replaces the line number with line number zero.
3. Prints the string variable A$, 16 characters at a time, on the device with a select code specified by P.

File 0 Program Listing (Identical for both the GPIO and HP-IB Interfaces)

```
0: "Lister":-1→N;16→P;on err "DoneLister";dim A$[106];buf "list",A$,1
1: N+1→N;fdf N;idf A,B,C,D,E,F;if N=0;fmt 2,c,f2.0;wrt P+.2,"Track",E
2: fmt 1,16"-",2/,c,f3.0;wrt P+.1,"FILE #",A
3: if B=6;ldf N,18,10
4: if B=5;wrt P,"KEY FILE";gto 1
5: if B=4;wrt P,"MEMORY FILE";gto 1
6: if B=3;wrt P,"DATA FILE";gto 1
7: if B=2;wrt P,"NUMERIC DATA";gto 1
8: if B=1;wrt P,"BINARY FILE";gto 1
9: if B=0;wrt P,"NULL FILE";gto 1
10: list #'L',18,18
11: red "list",A$
12: "next":len(A$)→Z;" 0"→A$[1,2]
13: for W=1 to 16*int(Z/16) by 16;wrt P,A$[W,W+15];next W
14: buf "list";gto 1;if W<Z;wrt P,A$[W,Z]
15: "DoneLister":if ern=65 and erl=1;dsp "DONE";spc 3;end
16: fmt 3,c,f2.0,c,f2.0;wrt .3,"Error ",char(rom),ern," line",erl;beep;stp
17: "L":ret "list.7"
*6478
```

Auto Loader (File 1)

With the exception of the track specified in line 7, the Auto Loader is identical for both GPIO and HP-IB interfaces. The Auto Loader is designed to search through the user written program and find "cll" statements. Then, the program will load card programs as required by the user program. When the program completes, the user is left with the original program plus all the card programs required from the Software Library. However, the user will have to load any programs not in the Software Library in order to complete the user program.

Detailed instructions on using the Auto Loader are contained in Section II. These instructions will not be repeated here.

Program Description. The key to the Auto Loader is the method of guessing the next available line number. Once this number is found, additional program lines can be loaded from the Library cartridge. The guessing process consists of these steps:

1. Set variable B equal to the number of bytes used to store the last file loaded.
2. Use 10 bytes per line as a starting point to guess the line number.
3. Guess that the next available line is the previous next available line number, 5, plus B/10, i.e., S + B/10 → S).
4. Try to load the next file starting at line number S.
5. If you have guessed wrong and generate an error, go the the error trapping routine and reduce your guess of the next available line by one (S − 1 → S).
6. Go to step 4 and try again until you succeed.

Another important part of the Auto Loader is the portion of the program that searches for "cll" statements. This portion of the program makes use of the list statement to an external device. However, in this case (line 12), the external device is specified by the function, "prog", line 38. The function "prog" defined by line 38 returns the buffer "buf" rather than a select code. The line listed can now be examined by examining string variables A$ because the buffer "buf" was defined as that string.

The program looks at each line to determine if a "cll" statement is on that line. When a "cll" statement is found, the label is isolated. Labels not found in the "directory" represented by string variable C$ will cause the following printout:

```
Not in Directory
cll '------'
```

Label not found

The program uses string variable B$ to keep a running list of Library programs that have been loaded. No Library program will be loaded twice, regardless of how many times it is called or whether both long and short labels are used.

File 1 Program Listings

GPIO (Track 0)

```
0: "AUTO LOADER":57→S;dim A$[106],B$[50],C$[900],D$[5];sto "Directory"
1: on err "error";buf "buf",A$,1;"cll '"→D$;"02"→B$
2: dsp "Insert User Program Tape";stp
3: ent "Track # ?",T,"Program FILE no. ?",F
4: trk T;fdf F;idf A,A,B
5: if A#6;prt "File type ",A;spc 2;jmp -2
6: S→P;ldf F,S,7
7: S+B/10→S;dsp "Insert SOFTWARE LIBRARY";stp ;trk 0
8: fdf 2;idf A,A,B
9: sfg 1;ldf 2,S,10
10: prnd(S,0)-1→G;cfg 1;S+B/10→S
11: for Z=P to G
12: list #'prog',Z,Z
13: red "buf",A$
14: if (pos(A$,D$)→X)#0;sto 18
15: if X#0;if len(A$)>X;A$[X]→A$;jmp -1
16: next Z
17: sto "end"
18: X+5→X;pos(A$[X],"'")→Y;Y-2→Y
19: if (pos(C$,A$[X,X+Y])→D)=0;prt "Not in Directory",A$[X-5,X+Y+1];sto 15
20: if pos(B$,C$[D-2,D-1])#0;sto 15
21: B$&",&C$[D-2,D-1]→B$
22: val(C$[D-2,D-1])→D
23: "lf":ldf D,S,24
24: fdf D;idf A,A,B;S+B/10→S
25: X+Y→X;sto 15
26: "error":
27: fmt 8,c,2x,c,f2.0
28: on err "error";S-1→S
29: if S<G or ern#31 or rom#0;wrt 16.8,"error",char(rom),ern;spc 2;beep;stp
30: if flg1=0;sto "lf"
31: cfg 1;sto 9
32: "end":spc ;prt "Subprograms from","the Directory","have been loaded"
33: ent "Record or Edit?",A$;if A$#"Edit" and A$#"EDIT" and A$#"edit";sto +3
34: spc ;prt "To remove loader","type in and","execute:"
35: prt "   del 0,56,*";spc 3;end
36: ent "What trk?",T;trk T;ent "What File?",F
37: rcf F,1*57;dsp "Done";end
38: "prog":ret "buf.1"
39: "Directory":"02Parameter Check02PC"→C$
40: "03oo03Simple Output04Serial Output04SO05Parallel Output05PO"&C$→C$
41: C$&"06Interrupt Output06IO07Simple Input07ii08Serial Input08SI"→C$
42: C$&"09Parallel Input09PI02Parameter Check02PC"→C$
43: C$&"10Clear Unit10CU"→C$
44: C$&"11Voltage Serial Output11VS12Voltage Parallel Output12VP"→C$
45: C$&"13Current Serial Output13CS14Current Parallel Output14CP"→C$
46: C$&"15Analog Input15AI16Single Ended Scan16SE"→C$
47: C$&"17Double Ended Scan17DE18External DVM Scanner18ES"→C$
48: C$&"19Motor Output19MO20Set Counter20SC21Read Counter21RC"→C$
49: C$&"22Timer Pulse22TP23Relay Output23RO24Time Interval24TI"→C$
50: C$&"25Frequency Measurement25FM26Option 40 Voltage Control26VC"→C$
51: C$&"27Option 40 Current Control27CC28Arm28AM29Serial Poll"→C$
52: C$&"29SP30Disarm30DM31Time Out31TO32Recycle32RE"→C$
53: C$&"33Reference Word33RW34T Thermocouple34TT35J Thermocouple35JT"→C$
54: C$&"36E Thermocouple36ET37R Thermocouple37RT38S Thermocouple38ST"→C$
55: C$&"39Thermocouple Reference39TR40B Thermocouple40BT"→C$
56: C$&"41K Thermocouple41KT"→C$;sto 1
*12704
```

```
0: "AUTO LOADER":57→S;dim A$[106],B$[50],C$[900],D$[5];gto "Directory"
1: on err "error";buf "buf",A$,1;"cll '"→D$;"02"→B$
2: dsp "Insert User Program Tape";stp
3: ent "Track # ?",T,"Program FILE no. ?",F
4: trk T;fdf F;idf A,A,B
5: if A#6;prt "File type ",A;spc 2;jmp -2
6: S→P;ldf F,S,7
7: S+B/10→S;dsp "Insert SOFTWARE LIBRARY";stp ;trk 1
8: fdf 2;idf A,A,B
9: sfg 1;ldf 2,S,10
10: prnd(S,0)-1→G;cfg 1;S+B/10→S
11: for Z=P to G
12: list #'prog',Z,Z
13: red "buf",A$
14: if (pos(A$,D$)→X)#0;gto 18
15: if X#0;if len(A$)>X;A$[X]→A$;jmp -1
16: next Z
17: gto "end"
18: X+5→X;pos(A$[X],"'")→Y;Y-2→Y
19: if (pos(C$,A$[X,X+Y])→D)=0;prt "Not in Directory",A$[X-5,X+Y+1];gto 15
20: if pos(B$,C$[D-2,D-1])#0;gto 15
21: B$&",",&C$[D-2,D-1]→B$
22: val(C$[D-2,D-1])→D
23: "lf":ldf D,S,24
24: fdf D;idf A,A,B;S+B/10→S
25: X+Y→X;gto 15
26: "error":
27: fmt 8,c,2x,c,f2.0
28: on err "error";S-1→S
29: if S<G or ern#31 or rom#0;wrt 16.8,"error",char(rom),ern;spc 2;beep;stp
30: if flg1=0;gto "lf"
31: cfg 1;gto 9
32: "end":spc ;prt "Subprograms from","the Directory","have been loaded"
33: ent "Record or Edit?",A$;if A$#"Edit" and A$#"EDIT" and A$#"edit";gto +3
34: spc ;prt "To remove loader","type in and","execute:"
35: prt "   del 0,56,*";spc 3;end
36: ent "What trk?",T;trk T;ent "What File?",F
37: rcf F,1*57;dsp "Done";end
38: "prog":ret "buf,1"
39: "Directory":"02Parameter Check02PC"→C$
40: "03oo03Simple Output04Serial Output04SO05Parallel Output05PO"&C$→C$
41: C$&"06Interrupt Output06IO07Simple Input07ii08Serial Input08SI"→C$
42: C$&"09Parallel Input09PI02Parameter Check02PC"→C$
43: C$&"10Clear Unit10CU"→C$
44: C$&"11Voltage Serial Output11VS12Voltage Parallel Output12VP"→C$
45: C$&"13Current Serial Output13CS14Current Parallel Output14CP"→C$
46: C$&"15Analog Input15AI16Single Ended Scan16SE"→C$
47: C$&"17Double Ended Scan17DE18External DVM Scanner18ES"→C$
48: C$&"19Motor Output19MO20Set Counter20SC21Read Counter21RC"→C$
49: C$&"22Timer Pulse22TP23Relay Output23RO24Time Interval24TI"→C$
50: C$&"25Frequency Measurement25FM26Option 40 Voltage Control26VC"→C$
51: C$&"27Option 40 Current Control27CC28Arm28AM29Serial Poll"→C$
52: C$&"29SP30Disarm30DM31Time Out31TO32Recycle32RE"→C$
53: C$&"33Reference Word33RW34T Thermocouple34TT35J Thermocouple35JT"→C$
54: C$&"36E Thermocouple36ET37R Thermocouple37RT38S Thermocouple38ST"→C$
55: C$&"39Thermocouple Reference39TR40B Thermocouple40BT"→C$
56: C$&"41K Thermocouple41KT"→C$;gto 1
*12472
```

CARD SUBPROGRAMS

The remaining pages in this section contain detailed descriptions of the card subprograms recorded on file nos. 2 through 41 of the 14556A cartridge. The format utilized in describing each subprogram is presented below. The individual card subprogram descriptions follow in file number order.

File Number: Call Labels

Abstract of subprogram.

SYNTAX

cll 'XX' (parameter 1, parameter 2, . . . , parameter N)

long or short label

PARAMETERS

Parameter List	Variable	Description	Default Value
parameter 1	p1	Description of parameter	Value assigned by subprogram is parameter not specified by calling program.

PROGRAM EXAMPLES

A typical program line using a "cll" statement to utilize the subprogram.

VARIABLE DESCRIPTION

Description of each variable used in the subprogram.

PROGRAM DESCRIPTION

Description of the function and use of the subprogram.

SPECIAL CONSIDERATIONS

List of items that may effect the use of the subprogram.

USED WITH

List of multiprogrammer plug-in cards that subprogram is designed to call.

DO NOT USE WITH

List of multiprogrammer plug-in cards that subprogram must not call. If any of the listed cards are called, the program will "hang-up."

PROGRAM LISTINGS

Both GPIO and HP-IB program listings are provided. The number of bytes used by each subprogram is given to aid the user in determining the size of the completed program.

File 2: Parameter Check (PC)

Checks if card slot address, card data, and unit number parameters are within limits.

SYNTAX

cll 'Parameter Check' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'PC' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1,p3,etc.	Card Slot Address	None
D_n	p2,p4,etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Parameter Check' (401, A, 407, G, 402, B, 0o)

or

cll 'Parameter Check' (1, 5252, 7, 2525, 2, 7777)

or

cll 'PC' (401, A, 407,7 7, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc. — All even number variables are data to be sent to output cards. If these variables are greater than 7777, the program will stop and permit entry of a new value.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

This subprogram is usually entered via a "gsb" from a Library subprogram. However, a "cll" could be made directly to this subprogram. Regardless of how the subprogram is accessed, the program examines parameters p1, p3, etc. to determine if the card slot addresses are correct. New slot addresses can be entered if an out of range number is encountered. To enter the new slot address type the correct slot address and assign it to the variable indicated.

Example: If the display shows:

"Slot address p3 out of range"
Type: 403 → p3 and press EXECUTE
Press CONTINUE

Data variables p2, p4, etc. as well as unit number (pp0) are also tested for limits. Again, if they are out of range, a new value must be assigned before the program will continue.

SPECIAL CONSIDERATIONS

1. Slot addresses must be from 400_{10} to 414_{10} or from zero to 14_{10}.
2. Only data greater than 7777 will be considered as out of range.

USED WITH

all plug-in cards.

PROGRAM LISTINGS

GPIO

```
0: "Parameter Check":
1: "PC":if fle10#1;mdec;dev "MP",9;wti 0,9;sfs 10
2: fmt 9,c,f2.0,/,c,f2.0,2/
3: if int(p0/2)=p0/2;1+p0→p0;0→pp0
4: fxd 0;mact;1→p39
5: if pp39>=400 and pp39<=414;pp39-400→pp39
6: if pp39>=0 and pp39<=14;jmp 2
7: dsp "Slot Address p",p39,"out of range";stp ;jmp -2
8: if p(p39+1)<=7777;jmp 2
9: dsp "Data, p",p39+1,"out of range";stp ;jmp -1
10: if (p39+2→p39)<=p0-1;gto -5
11: if pp0<=15;jmp 2
12: dsp "Unit no.",pp0,"out of range";stp ;jmp -1
13: ret
*21981
```

Bytes used: 466

HP-IB

```
0: "Parameter Check":
1: "PC":if fle10#1;dev "MP",723,"HP-IB",7;sfs 10
2: fmt 9,c,f2.0,/,c,f2.0,2/
3: if int(p0/2)=p0/2;1+p0→p0;0→pp0
4: 1→p39
5: if pp39>=400 and pp39<=414;pp39-400→pp39
6: if pp39>=0 and pp39<=14;jmp 2
7: dsp "Slot address p",p39," out of range";stp ;jmp -2
8: if p(p39+1)<=7777;jmp 2
9: dsp "Data,p",1+p39,"out of range";stp ;jmp -1
10: if (p39+2→p39)<=p0-1;gto -5
11: if pp0<=15;jmp 2
12: dsp "Unit no.",pp0,"out of range";stp ;jmp -1
13: fmt 1,c,z;fmt 2,c,f4.0,c,z;ret
*8261
```

Bytes used: 486

File 3: Simple Output (oo)

Sends octal data (o to 7777) to the output card (only one card per call) specified. Does not cause 6940B to wait for flag.

SYNTAX

cll 'Simple Output' (S, D, [U])

or

cll 'oo' (S, D, U)

PARAMETERS

Parameter List	Variable	Description	Default Value
S	p1	Card Slot Address	None
D	p2	Card Data	None
U	pp0	Unit No.	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Simple Output' (401, A, 0)

or

cll 'Simple Output' (1, 5252)

or

cll 'oo' (401, A)

In the first example, the long descriptive label and the complete slot address (401) is used, variable data is sent, and the unit number is included. In the next example, the shortened slot address (1) is used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label. complete slot addresses are used, variable data is sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1 — This variable is used for slot address. The value must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot address must be in decimal.

p2 — This variable is the data to be sent to the output card. If this variable is greater than 7777, the program will stop and permit entry at a new value.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. Note that only the SYE mode bit is turned on and the unit number is in octal (dto pp0). Data p2 is then sent to the slot address specified by p1. Finally, a control word with SYE mode bit on is sent to unit 0 (the 6940B) before a return is made to the main program.

SPECIAL CONSIDERATIONS

1. Data sent to the output card must be in octal.
2. Only one card can be called with this subprogram.
3. Since TME is not on, the subroutine will not wait for an external device to time out.

USED WITH

69332A Open Collector Output, 69380A Breadboard Output

PROGRAM LISTING

GPIO

```
0: "Simple Output":
1: "oo":gsb "Parameter Check"
2: if p0>3;beep;dsp "Wrong Number of Parameters";stp
3: wtb "MP";170040+dtopp0
4: wtb "MP";10000*dtop1+p2
5: wtb "MP";170040;mdec
6: ret
*5861
```

Bytes used: 172

HP-IB

```
0: "Simple Output":
1: "oo":gsb "Parameter Check"
2: if p0>3;beep;dsp "Wrong Number of Parameters";stp
3: wrt "MP.2","0",40+dtopp0,"T"
4: wrt "MP.2",char(64+p1),p2,"T"
5: wrt "MP.1","00040T"
6: ret
*12778
```

Bytes used: 186

File 4: Serial Output (SO)

Sends octal data (0 to 7777) to the output cards specified. Card outputs are updated sequentially.

SYNTAX

cll 'Serial Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'SO' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4,etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Serial Output' (401, A, 407, G, 402, B, 0)

or

cll 'Serial Output' (1, 5252, 7, 2525, 2, 7777)

or

cll 'SO' (401, A, 407, 77, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 4: Serial Output (SO)

p2, p4, etc.	All even number variables are data to be sent to output cards. If these variables are greater than 7777, the program will stop and permit entry of a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits DTE, SYE, and TME are programmed on by this control word. Therefore, the program will wait for each output to complete its time-out before the next output can be programmed. The program then loops until all slot addresses (p1, p3, etc.) have received the appropriate data (p2, p4, etc.). Finally, a control word with SYE mode bit is sent to unit 0 (the 6940B) before a return is made to the main program.

SPECIAL CONSIDERATIONS

1. Data sent to the call subprogram must be in octal.
2. With DTE and TME mode bits on, cards such as 69331A Digital Output and 69330A Relay Output will send an external gate and wait for an external flag.

USED WITH

69330A Relay Output, 69331A Digital Output, and 69433A Relay Output/Readback.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Serial Output":
1: "SO":gsb "Parameter Check"
2: wtb "MP",170160+dtopp0
3: 1→p39
4: wtb "MP",10000*dtopp39+p(p39+1)
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170040;mdec
7: ret
*12295
```

Bytes used: 180

HP-IB

```
0: "Serial Output":
1: "SO":gsb "Parameter Check"
2: wrt "MP.2","0",160+dtopp0,"T"
3: 1→p39
4: wrt "MP.2",char(64+pp39),p(p39+1),"T"
5: if (p39+2→p39)<=p0-1;gto -1
6: wrt "MP.1","00040T"
7: ret
*6293
```

Bytes used: 194

File 5: Parallel Output (PO)

Sends octal data (0 to 7777) to output cards specified. Card outputs are updated simultaneously.

SYNTAX

cll 'Parallel Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'PO' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1,p3,etc.	Card Slot Address	None
D_n	p2,p4,etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Parallel Output' (401, A, 407, G, 402, B, 0)

or

cll 'Parallel Output' (1, 5252, 7, 2525, 2, 7777)

or

cll 'PO' (401, A, 407, 77, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc.	All even number variables are data to be sent to output cards. If these variables are greater than 7777, the program will stop and permit entry of a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits DTE, SYE, and TME are programmed on by this control word. Therefore, the program will wait for each output to complete its time-out before the next output can be programmed. The program then loops until all slot addresses (p1, p3, etc.) have received the appropriate data (p2, p4, etc.). Finally, a control word with SYE mode bit is sent to unit 0 (the 6940B) before a return is made to the main program.

SPECIAL CONSIDERATIONS

1. Data sent to the call subprogram must be in octal.
2. With DTE and TME mode bits on, cards such as 69331A Digital Output and 69330A Relay Output will send an external gate and wait for an external flag.

USED WITH

69330A Relay Output, 69331A Digital Output, and 69433A Relay Output/Readback.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Parallel Output":
1: "PO":gsb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: wtb "MP",10000*dtopp39+p(p39+1)
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170160,170040;rdec
7: ret
*6433
```

Bytes used: 188

HP-IB

```
0: "Parallel Output":
1: "PO":gsb "Parameter Check"
2: wrt "MP.2","O",40+dtopp0,"T"
3: 1→p39
4: wrt "MP.2",char(64+pp39),p(p39+1),"T"
5: if (p39+2→p39)<=p0-1;gto -1
6: wrt "MP.1","O0160T00040T"
7: ret
*2202
```

Bytes used: 202

File 6: Interrupt Output (IO)

Sends octal data (0 to 7777) to the output cards specified. Uses the 9825A's interrupt capability to indicate when data transfers with external devices are completed

SYNTAX

cll 'Interrupt Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'IO' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1,p3,etc.	Card Slot Address	None
D_n	p2,p4,etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Interrupt Output' (401, A, 407, G, 402, B, 0)

or

cll 'Interrupt Output' (1, 5252, 7, 2525, 2, 7777)

or

cll 'IO' (401, A, 407, 77, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 6: Interrupt Output (IO)

p2, p4, etc.	All even number variables are data to be sent to output cards. If these variables are greater than 7777, the program will stop and permit entry of a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0 with only the SYE mode bit on. The subprogram then loops until all slot addresses (p1, p3, etc.) receive the appropriate data (p2, p4, etc.). Finally, a control word with DTE, SYE, and TME mode bits on is sent. This will permit the cards that were addressed and are timing out to keep the flag to the 9825A Desktop Computer busy until they have all timed out. Before returning to the main program, an on interrupt statement (oni) is executed which establishes a linkage between select code 9 and service routine "Done". An enable interrupt statement (eir) is also executed. When the last card has timed out, a trailing edge of flag will be sent to the 9825A, causing a branch to the specified interupt service routine ("Done").

Following is an example of an interrupt service routine:

```
"Done":

prt "All cards have", "timed out."
moct; wtb 9, 170040; mdec
iret
```

SPECIAL CONSIDERATIONS

1. Data sent to the call subprogram must be in octal.
2. The label "Done" must be used only for the interrupt service routine.

USED WITH

69330A Relay Output, 69331A Digital Output.

DO NOT USE WITH

69332A Open Collector Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Interrupt Output":
1: "IO":gsb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: wtb "MP",10000*dtopp39+p(p39+1)
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170160;oni "MP","Done";eir "MP";mdec
7: ret
8: "Done":beep;dsp "Done";wait 1000;-1→0;iret
*13917
```

Bytes used: 238

File 6: Interrupt Output (IO)

HP-IB

```
0: "Interrupt Output":
1: "IO":esb "Parameter Check"
2: rds("MP")→p(p0+3)
3: wrt "MP.2","0",40+dtopp0,"T"
4: 1→p39
5: wrt "MP.2",char(64+pp39),p(p39+1),"T"
6: if (p39+2→p39)<=p0-1;gto -1
7: wrt "MP.1","00160T";oni "HP-IB","Done";eir "HP-IB"
8: ret
9: "Done":beep;dsp "Done";wait 1000;-1→0;iret
*9065
```

Bytes used: 278

File 7: Simple Input (ii)

Reads octal data from the input card (only one card per call) specified. The data is returned in octal. Does not cause the 6940B to wait for flag.

SYNTAX

cll 'Simple Input' (S, D [U])

or

cll 'ii' (S, D, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S	p1	Card Slot Address	None
D	p2	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Simple Input' (401, A, 0)

or

cll 'Simple Input' (1, A)

or

cll 'ii' (401, A)

In the first example, the long descriptive label and the complete slot address (401) is used, the data storage location is specified, and the unit number is included. In the next example, the shortened slot address (1) is used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1 — This variable is used for slot the address. The value must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot address must be in decimal.

p2 — This variable is used to store data from the input cards. Data will be returned in octal.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits ISL and SYE are programmed on by this control word. An address word is sent without a gate and the data is read without a gate. Before returning to the main program, a control word with SYE mode bit on is programmed.

SPECIAL CONSIDERATIONS

1. Only one card will be read with this call subprogram, regardless of how many are specified in the parameter list.

USED WITH

69430A Isolated Digital Input and 69480A Breadboard Input.

PROGRAM LISTINGS

GPIO

```
0: "Simple Input":
1: "ii":gsb "Parameter Check"
2: wtb "MP",170240+dtopp0
3: wti 4,10000+dtop1;rdi 4→p2
4: wtb "MP",170040;mdec
5: ret
+5700
```

Bytes used: 130

HP-IB

```
0: "Simple Input":
1: "ii":gsb "Parameter Check"
2: wrt "MP.2","0",240+dtopp0,"T"
3: wrt "MP.2",char(64+p1),"X";red "MP",p2
4: wrt "MP.1","00040T"
5: ret
*18363
```

Bytes used: 146

File 8: Serial Input (SI)

Returns octal data (IRQ bit removed) from the input cards specified. Cards are read sequentially.

SYNTAX

cll 'Serial Input' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'SI' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Serial Input' (401, A, 407, G, 402, B, 0)

or

cll 'Serial Input' (1, A, 7, G, 2, B,)

or

cll 'SI' (401, A, 407, G, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, the data storage locations are specified, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc. — All even numbered variables are used to store data from input cards. The IRQ bit is removed in the call subprogram. Data will be returned in octal.

pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits ISL, SYE, and TME are programmed on by this control word. Therefore, the program will wait for each input to complete its time-out before the next input can be read. The subprogram then loops until all slot addresses (p1, p3, etc.) have been addressed and read with a gate. Finally, a control word with SYE mode bit is sent to unit 0 (the 6940B) before a return is made to the main program.

SPECIAL CONSIDERATIONS

1. Data returned will be in octal..
2. With TME mode bit on, the 69431A will keep the 6940B flag busy until an external flag returns a trailing edge.

USED WITH

69431A Digital Input.

DO NOT USE WITH

69430A Isolated Digital Input, 69435A Pulse Counter, 69480A Breadboard Input and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Serial Input":
1: "SI":gsb "Parameter Check"
2: wtb "MP",170260+dtopp0
3: 1→p39
4: wti 4,10000*dtopp39;rdb("MP")→p(p39+1)
5: if p(p39+1)>7777;p(p39+1)-100000→p(p39+1)
6: if (p39+2→p39)<=p0-1;gto -2
7: wtb "MP",170040;mdec
8: ret
*25248
```

Bytes used: 232

HP-IB

```
0: "Serial Input":
1: "SI":gsb "Parameter Check"
2: wrt "MP.2","0",260+dtopp0,"T"
3: 1→p39
4: wrt "MP.2",char(64+pp39),"T",char(64+pp39),"X";red "MP",p(p39+1)
5: if p(p39+1)>7777;p(p39+1)-10000→p(p39+1)
6: if (p39+2→p39)<=p0-1;gto -2
7: wrt "MP.1","00040T"
8: ret
*20115
```

Bytes used: 266

File 9: Parallel Input (PI)

Returns octal data (IRQ bit removed) from the input cards specified. The first card that has data ready is read first, followed by the next ready card, etc. The subprogram will not complete until all cards have been read.

SYNTAX

cll 'Parallel Input' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'PI' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Parallel Input' (401, A, 407, G, 402, B, 0)

or

cll 'Parallel Input' (1, A, 7, G, 2, B,)

or

cll 'PI' (401, A, 407, G, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, the data storage locations are specified, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc.	All even numbered variables are used to store data from input cards. The IRQ bit is removed in the call subprogram. Data will be returned in octal.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The ISL, SYE mode bits are programmed on by this control word. The the program then loops until all slot addresses (p1, p3, etc.) specified by the call subprogram have been addressed with a gate. Variable p (3 + p0) has a bit set for each card addressed.

The interrupt mode in the 6940B is then enabled by sending a control word with IEN, SYE and TME mode bits on. Then the interrupt linkage is established between interrupts and the program starting at label "IRQ". The subprogram now waits until all cards have interrupted at least once. Each time an interrupt occurs, the program leaves the line labeled "waiting" and branches to the line labeled "IRQ".

Once at "IRQ" a control word is sent with ISL and SYE mode bits on to the unit specified in the call. Each card is now read without gate to determine if it has interrupted (that is, if IRQ is true). If a card has interrupted, a bit is set in variable p (+ p0) and the card is recycled by addressing it with a gate and disarmed. The IRQ bit is then removed from the data.

When all cards have interrupted once, variable p (+ p0) equals variable p (+ p0). The subprogram then programs a control word with SYE mode bit on before returning to the main program.

SPECIAL CONSIDERATIONS

1. Only the first interrupt from a card will cause data to be stored in the specified variable.
2. Any 69431A, 69434A, 69436A, or 69600B card in the system with jumper W6 in (as received from the factory) will be armed and capable of generating an interrupt after the control word with IEN, SYE and TME mode bits on is programmed.
3. Data will be returned in octal with the IRQ bit removed.
4. Jumper W6 does not have to be removed for this call.

USED WITH

69431A Digital Input.

DO NOT USE WITH

69434A Event Sense, 69436A Process Interrput, 69600B Programmable Timer.

PROGRAM LISTINGS

GPIO

```
0: "Parallel Input":
1: "PI":gsb "Parameter Check"
2: "arm":wtb "MP",170240+dtopp0
3: 1→p39
4: 2↑pp39+p(3+p0)→p(3+p0)
5: wtb "MP",10000*dtopp39
6: if (p39+2→p39)<=p0-1;gto -2
7: wtb "MP",170460;oni "MP","IRQ"
8: "waiting":if p(3+p0)#p(2+p0);eir "MP";gto +0
9: "ret":wti 5,40;wtb "MP",170040;mdec;ret
10: "IRQ":wtb "MP",170240+dtopp0
11: 1→p39
12: mdec;if bit(pp39,p(2+p0))=1;gto "increment"
13: moct;wti 4,10000*dtopp39;rdi 4→p(p39+1)
14: if p(p39+1)<10000;gto "increment"
15: mdec;p(p39+1)-100000→p(p39+1);ior(2↑pp39,p(2+p0))→p(2+p0);moct
16: "disarm":wtb "MP",170040+dtopp0,10000*dtopp39
17: "increment":if (p39+2→p39)<=p0-1;gto -5
18: moct;wtb "MP",170460;iret
*18480
```

Bytes used: 644

HP-IB

```
0: "Parallel Input":
1: "PI":gsb "Parameter Check"
2: "arm":wrt "MP.2","0",240+dtopp0,"T"
3: 1→p39
4: 2↑pp39+p(3+p0)→p(3+p0)
5: wrt "MP.2",char(64+pp39),"T"
6: if (p39+2→p39)<=p0-1;gto -2
7: rds("MP")→p(p0+5);wrt "MP.1","00460T";oni "HP-IB","IRQ"
8: "waiting":if p(3+p0)#p(2+p0);eir "HP-IB";gto +0
9: "ret":wrt "MP.1","X00040T";iret
10: "IRQ":rds("MP")→p(p0+5);wrt "MP.2","0",240+dtopp0,"T"
11: 1→p39
12: if bit(pp39,p(2+p0))=1;gto "increment"
13: wrt "MP.2",char(64+pp39),"X";red "MP",p(p39+1)
14: if p(p39+1)<10000;gto "increment"
15: p(p39+1)-10000→p(p39+1);ior(2↑pp39,p(2+p0))→p(2+p0)
16: "disarm":wrt "MP.2","0",40+dtopp0,"T";wrt "MP.1",char(64+pp39),"T"
17: "increment":if (p39+2→p39)<=p0-1;gto -5
18: wrt "MP.1","00460T";iret
*14637
```

Bytes used: 714

File 10: Clear Unit (CU)

Sends logic zeroes to clear all cards in unit (only one unit per call) specified. For example, logic zeroes will clear a Relay Output Card (open all 12 relays).

SYNTAX

cll 'Clear Unit' (U)

or

cll 'CU' (U)

PARAMETERS

Parameter List	Variable	Description	Default Value
p1		Unit Number Address	None

PROGRAM EXAMPLES

cll 'Clear Unit' (4)

or

cll 'Clear Unit' (P)

or

cll 'CU' (P)

The first example uses the long descriptive label to call "Clear Unit". The unit number is given as the constant 4, for the fourth 6941B. The second and third examples use variable P to designate which unit will be cleared. A short label is used in the third example.

VARIABLE DESCRIPTION

p1	Unit number for unit to be cleared (zero data sent to all cards).

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Paramater Check', this subprogram sends a control word to the unit designated by pp0. In this case, pp0 will equal p1, since only one parameter was passed. Next the subprogram sends zero data to all card slots in the unit specified. The subprogram does not wait for any time outs and sends two control words before returning to the main program. The first control word has only the DTE mode bit programmed on. This control word is used to force the 69321B Voltage D/A and 69370A Current D/A to update their second rank of storage to zero. The second control word has only the SYE mode bit on.

File 10: Clear Unit (CU)

SPECIAL CONSIDERATIONS

1. Only one unit can be cleared per call.

USED WITH

All cards.

PROGRAM LISTINGS

GPIO

```
0: "Clear Unit":
1: "CU":ssb "Parameter Check"
2: wti 5,40;wtb "MP",170000+dtop0
3: 0→p39
4: wtb "MP",10000*dtop39
5: if (p39+1→p39)<=14;sto -1
6: wtb "MP",170100,170040
7: mdec;ret
*5907
```

Bytes used: 174

HP-IB

```
0: "Clear Unit":
1: "CU":ssb "Parameter Check"
2: wrt "MP.2","X0",dtop0,"T"
3: 0→p39
4: wrt "MP.1",char(64+p39),"T"
5: if (p39+1→p39)<=14;sto -1
6: wrt "MP.1","0100T040T";ret
*32748
```

Bytes used: 172

File 11: Voltage Serial Output (VS)

Sends desired output voltage (−10.240 to +10.235V) in decimal form (e.g., −6.5 for −6.5 volts) to the D/A cards specified. Card outputs are updated sequentially. Note that a 69322A D/A card has 4 output channels which can be individually updated.

SYNTAX

cll 'Voltage Serial Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'VS' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Voltage Serial Output' (401, A, 407, G, 402.3, B, 0)

or

cll 'Voltage Serial Output' (1, −10, 7, 5, 2.3, 4.75)

or

cll 'VS' (401, A, 407, 5, 402.3, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. Note that slot address 402.3 specifies output no. 3 of a 69322A Quad D/A card in slot 2 (see variable description).In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

File 11: Voltage Serial Output (VS)

VARIABLE DESCRIPTION

p1, p3, etc.

All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

Slot addresses for the 69322A Quad D/A Voltage Converter card must have one digit to the right of the decimal point. This digit will specify the output that is to receive new data.

Example:

```
cll 'Voltage Serial Output (401.1, 10)
                            ↗
                  indicates Output No. 1 (terminal A)
```

Slot Address	Output No.	Output Terminal
XXX.1	1	A
XXX.2	2	B
XXX.3	3	D
XXX.4	4	E

Where XXX can be any slot address from 400 to 414.

p2, p4, etc.

All even number variables are data to be sent to output cards. If these variables are greater than 7777, the program will stop and permit entry at a new value.

Data variables are to be expressed in volts. Maximum limits are:

	Negative Output	Positive Output
69321B	−10.240V	+10.235V
69322A	−10.240V	+10.220V

pp0

If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits DTE, SYE, and TME are programmed on by this control word. The program then addresses each card specified, sending it data equivalent to the desired output voltage. Since the DTE mode bit is on, the output is updated immediately. After each card times-out, the next card is sent the appropriate data. Quad D/A Voltage Converters, 69322A, are identified by the non-zero fraction in the slot address variable. A branch to "QUAD" occurs when a 69322A address is sent. This part of the program combines the output data and output number information into one variable (p(p39+41)) for output. After the last card specified has timed-out, the program sends a control word with SYE on to unit 0, the 6940B, before returning to the main program.

SPECIAL CONSIDERATIONS

1. Cards such as 69331A, 69330A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subroutine.
2. Slot addresses that are the result of calculations may have a non-zero fraction that would make them appear to be for 69322A Quad D/A Voltage converters. To prevent this, use the power of ten rounding function (prnd) to remove the fraction (e.g., prnd (A, 0).
3. Current D/A's (69370A's) as well as Voltage D/A's (69321B's) that were previously addressed when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.

USED WITH

69321B D/A Voltage Converter, 69322A Quad D/A Voltage Converter.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Voltage Serial Output":
1: "VS":ssb "Parameter Check"
2: wtb "MP";170160+dtop(p0
3: 1→p39
4: pp39→p(p39+40);if frc(pp39)#0;sto "QUAD"
5: if p(p39+1)>10.235 or p(p39+1)<-10.24;beep;fxd 0;sto +10
6: p(p39+1)/.005→p(p39+41);if p(p39+41)<0;p(p39+41)+4096→p(p39+41)
7: wtb "MP";10000*dtop(p39+40)+dtop(p39+41)
8: if (p39+2→p39)<=p0-1;sto -4
9: wtb "MP";170040;mdec
10: ret
11: "QUAD":if p(p39+1)>10.22 or p(p39+1)<-10.24;sfg 9;sto +4
12: p(p39+1)/.02→p(p39+41);if p(p39+41)<0;p(p39+41)+1024→p(p39+41)
13: (10*frc(p(p39+40))-1)*1024+p(p39+41)→p(p39+41)
14: prnd(pp39,0)→p(p39+40);jmp -7
15: dsp "Data, p";p39+1;"out of range";stp ;sto -10;if fla9;cfg 9;sto -4
*9219
```

Bytes used: 656

HP-IB

```
0: "Voltage Serial Output":
1: "VS":ssb "Parameter Check"
2: wrt "MP.2","0";160+dtop(p0,"T"
3: 1→p39
4: pp39→p(p39+40);if frc(pp39)#0;sto "QUAD"
5: if p(p39+1)>10.235 or p(p39+1)<-10.24;beep;fxd 0;sto +10
6: p(p39+1)/.005→p(p39+41);if p(p39+41)<0;p(p39+41)+4096→p(p39+41)
7: wrt "MP.2";char(64+p(p39+40));dtop(p39+41),"T"
8: if (p39+2→p39)<=p0-1;sto -4
9: wrt "MP.1","00040T"
10: ret
11: "QUAD":if p(p39+1)>10.22 or p(p39+1)<-10.24;sfg 9;sto +4
12: p(p39+1)/.02→p(p39+41);if p(p39+41)<0;p(p39+41)+1024→p(p39+41)
13: (10*frc(p(p39+40))-1)*1024+p(p39+41)→p(p39+41)
14: prnd(p(p39+40),0)→p(p39+40);jmp -7
15: dsp "Data, p";p39+1;"out of range";stp ;sto -10;if fla9;cfg 9;sto -4
*22743
```

Bytes used: 674

File 12: Voltage Parallel Output (VP)

Sends desired output voltage (−10.240V to +10.235V) in decimal form (e.g., 7.5 for +7.5 volts) to the cards specified. 69322A Quad D/A card outputs are updated when the applicable outputs are addressed (e.g., to address channel 2 of 69322A in slot 3 use address 3.2). The 69321B cards are simultaneously updated to their new output voltages.

SYNTAX

cll 'Voltage Parallel Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'VP' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Voltage Parallel Output' (401, A, 407, G, 402.3, B, 0)

or

cll 'Voltage Parallel Output' (1, −10, 7, 5, 2.3, 4.75)

or

cll 'VP' (401, A, 407, 5, 402.3, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. Note that slot address 402.3 specifies output No. 3 of a 69322A Quad D/A card in slot 2 (see variable description). In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

File 12: Voltage Parallel Output (VP)

VARIABLE DESCRIPTION

p1, p3, etc.

All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

Slot addresses for the 69322A Quad D/A Voltage Converter card must have one digit to the right of the decimal point. This digit will specify the output that is to receive new data.

Example:

```
cll 'Voltage Serial Output' (401.1, 10)
                             ↑
                             Indicates Output No. 1 (terminal A)
```

Slot Address	Output No.	Output Terminal
XXX.1	1	A
XXX.2	2	B
XXX.3	3	D
XXX.4	4	E

Where XXX can be any slot address from 400 to 414.

p2, p4, etc.

All even number variables are data to be sent to output cards. If these variables are greater than 7777, the program will stop and permit entry at a new value.

Data variables are to be expressed in volts. Maximum limits are:

	Negative Output	Positive Output
69321B	−10.240V	+10.235V
69322A	−10.240V	+10.220V

pp0

If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bit SYE is programmed on by this control word. The program will address each card specified, sending it the data equivalent to the desired output voltage. Since the DTE mode bit is off, the outputs of only the 69321B cards are not updated immediately. Outputs of the 69322A cards, however, are updated immediately. After all cards have received data, another control word is sent with DTE, SYE, and TME mode bits on. The TME mode bit forces the program to wait for all cards to time out before sending the final control word with SYE mode bit on and returning to the main program. The DTE mode bit causes all 69321B outputs to update simultaneously.

File 12: Voltage Parallel Output (VP)

Quad D/A Voltage Converters, 69322A, are identified by the non-zero fraction in the slot address variable. A branch to "QUAD1" occurs when a 69322A address is sent. This part of the program combines the output data and output number information into one variable p(p39+41) for output.

SPECIAL CONSIDERATIONS

1. Cards such as 69331A, 69330A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subroutine.
2. Slot addresses that are the result of calculations may have a non-zero fraction that would make them appear to be for 69322A Quad D/A Voltage converters. To prevent this, use the power of ten rounding function (prnd) to remove the fraction (e.g., prnd (A, 0).
3. Current D/A's (69370A's) as well as Voltage D/A's (69321B's) that were previously addressed when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.

USED WITH

69321A D/A Voltage Converter, 69322A Quad D/A Voltage Converter.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

```
0: "Voltage Parallel Output":
1: "VP":gsb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: pp39→p(p39+40);if frc(pp39)#0;gto "QUAD1"
5: if p(p39+1)>10.235 or p(p39+1)<-10.24;beep;fxd 0;gto +10
6: p(p39+1)/.005→p(p39+41);if p(p39+41)<0;p(p39+41)+4096→p(p39+41)
7: wtb "MP",10000*dtop(p39+40)+dtop(p39+41)
8: if (p39+2→p39)<=p0-1;gto -4
9: wtb "MP",170160,170040;mdec
10: ret
11: "QUAD1":if p(p39+1)>10.22 or p(p39+1)<-10.24;sfg 9;gto +4
12: p(p39+1)/.02→p(p39+41);if p(p39+41)<0;p(p39+41)+1024→p(p39+41)
13: (10*frc(p(p39+40))-1)*1024+p(p39+41)→p(p39+41)
14: prnd(pp39,0)→p(p39+40);jmp -7
15: dsp "Data; p";p39+1," out of range";stp ;gto -10;if flg9;cfg 9;gto -4
*439
```

Bytes used: 668

HP-IB

```
0: "Voltage Parallel Output":
1: "VP":gsb "Parameter Check"
2: wrt "MP.2","0",40+dtopp0,"T"
3: 1→p39
4: pp39→p(p39+40);if frc(pp39)#0;gto "QUAD1"
5: if p(p39+1)>10.235 or p(p39+1)<-10.24;beep;fxd 0;gto +10
6: p(p39+1)/.005→p(p39+41);if p(p39+41)<0;p(p39+41)+4096→p(p39+41)
7: wrt "MP.2",char(64+p(p39+40)),dtop(p39+41),"T"
8: if (p39+2→p39)<=p0-1;gto -4
9: wrt "MP.1","00160T00040T"
10: ret
11: "QUAD1":if p(p39+1)>10.22 or p(p39+1)<-10.24;sfg 9;gto +4
12: p(p39+1)/.02→p(p39+41);if p(p39+41)<0;p(p39+41)+1024→p(p39+41)
13: (10*frc(p(p39+40))-1)*1024+p(p39+41)→p(p39+41)
14: prnd(p(p39+40),0)→p(p39+40);jmp -7
15: dsp "Data; p",p39+1,"out of range";stp ;gto -10;if flg9;cfg 9;gto -4
*29019
```

Bytes used: 684

File 13: Current Serial Output (CS)

Sends desired output current (0 to 20.475mA) in decimal form (e.g., 4.5 for +4.5mA) to the cards specified. Cards are updated sequentially to their new output currents.

SYNTAX

cll 'Current Serial Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'CS' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Current Serial Output' (401, A, 407, G, 402, B, 0)

or

cll 'Current Serial Output' (1, 10, 7, 5, 2, 4.75)

or

cll 'CS' (401, A, 407, 5, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 13: Current Serial Output (CS)

p2, p4, etc.	All even number variables are data to be sent to output cards. Data variables are in milliamps. If these variables are greater than 20.475 or less than 0, the program will stop and permit entry at a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits DTE, SYE, and TME are programmed on by this control word. The program then addresses each card specified, sending it data equivalent to the desired output current. Since the DTE mode bit is on, the output is updated immediately. After each card times-out, the next card is sent the appropriate data. After the last card specified has timed-out, the program sends a control word with SYE on to unit 0, the 6940B, before returning to the main program.

SPECIAL CONSIDERATIONS

1. Cards such as 69331A, 69330A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subroutine.
2. Negative current values may cause data to be sent to cards not specified in the call sequence.
3. Voltage D/A's (69321B's) as well as Current D/A's (69370A's) that were previously addressed when DTE was off will update their second rank of storage and their output when DTE programmed on in this call subprogram.

USED WITH

69370A D/A Current Converter.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Current Serial Output":
1: "CS":gsb "Parameter Check"
2: wtb "MP",170160+dtopp0
3: 1→p39
4: if p(p39+1)<=20.475 and p(p39+1)>=0;gto +2
5: beep;fxd 0;dsp "Data p",p39+1,"out of range";stp ;gto -1
6: wtb "MP",10000*dtopp39+dto(p(p39+1)/.005)
7: if (p39+2→p39)<=p0-1;gto -3
8: wtb "MP",170040;mdec
9: ret
*30318
```

Bytes used: 292

File 13: Current Serial Output (CS)

HP-IB

```
0: "Current Serial Output":
1: "CS":esb "Parameter Check"
2: wrt "MP.2","O",160+dtopp0,"T"
3: 1→p39
4: if p(p39+1)<=20.475 and p(p39+1)>=0;eto +2
5: beep;fxd 0;dsp "Data p",p39+1,"out of range";stp ;eto -1
6: wrt "MP.2",char(64+pp39),dto(p(p39+1)/.005),"T"
7: if (p39+2→p39)<=p0-1;eto -3
8: wrt "MP.1","Q0040T"
9: ret
*8576
```

Bytes used: 306

File 14: Current Parallel Output (CP)

Sends desired output current (0 to 20.475mA) in decimal form (e.g., 4.5 for +4.5mA) to the cards specified. The cards are simultaneously updated to their new output currents.

SYNTAX

cll 'Current Parallel Output' (S_0, D_0, S_1, D_1, ---- S_n, D_n, [U])

or

cll 'CP' (S_0, D_0, S_1, D_1, ---- S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Current Parallel Output' (401, A, 407, G, 402, B, 0)

or

cll 'Current Parallel Output' (1, 10, 7, 5, 2, 4.7)

or

cll 'CP' (401, A, 407, 5, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used forthe "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 14: Current Parallel Output (CP)

p2, p4, etc.	All even number variables are data to be sent to output cards. Data variables are in milliamps. If these variables are greater than 20.475 or less than 0, the program will stop and permit entry at a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits DTE, and SYE are programmed on by this control word. The program then addresses each card specified, sending it the data equivalent to the desired output current. Since the DTE mode bit is on, the output is updated immediately. After all cards have received data, another control word is sent with DTE, SYE and TME mode bits on. The TME mode bits forces the program to wait for all cards to time out before sending the final control word with SYE mode bit on and returning to the main program.

SPECIAL CONSIDERATIONS

1. Cards such as 69331A, 69330A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subroutine.
2. Slot addresses that are the result of calculations may have a non-zero fraction that would make them appear to be for 69322A Quad D/A Voltage converters. To prevent this, use the power of ten rounding function (prnd) to remove the fraction (e.g., prnd (A, 0).
3. Voltage D/A's (69321B's) as well as Current D/A's (69370A's) that were previously addressed when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.

USED WITH

69370A D/A Current Converter.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

PROGRAM LISTINGS

GPIO

```
0: "Current Parallel Output":
1: "CP":gsb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: if p(p39+1)<=20.475 and p(p39+1)>=0;gto +2
5: beep;fxd 0;dsp "Data p",p39+1,"out of range";stp ;gto -1
6: wtb "MP",10000*dtopp39+dto(p(p39+1)/.005)
7: if (p39+2→p39)<=p0-1;gto -3
8: wtb "MP",170160,170040;mdec
9: ret
*16696
```

Bytes used: 300

File 14: Current Parallel Output (CP)

HP-IB

```
0: "Current Parallel Output":
1: "CP":esb "Parameter Check"
2: wrt "MP.2","0",40+dtopp0,"T"
3: 1→p39
4: if p(p39+1)<=20.475 and p(p39+1)>=0;gto +2
5: beep;fxd 0;dsp "Data p",p39+1,"out of range";stp ;gto -1
6: wrt "MP.2",char(64+pp39),dto(p(p39+1)/.005),"T"
7: if (p39+2→p39)<=p0-1;gto -3
8: wrt "MP.1","00160T00040T"
9: ret
*20323
```

Bytes used: 314

File 15: Analog Input (AI)

Reads voltage data (−10 to +10V) from the cards specified. Returned data is in decimal form (e.g., −5.5 for −5.5 volts). Cards are read sequentially.

SYNTAX

cll 'Analog Input' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'AI' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Analog Input' (401, A, 407, G, 402, B, 0)

or

cll 'Analog Input' (1, A, 7, G, 2, B,)

or

cll 'AI' (401, A, 407, G, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, the data storage locations are specified, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc. — All even number variables are used to store data from input cards. Data will be returned in decimal and will indicate the voltage read.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits ISL, SYE, and TME are programmed on by this control word. The program then reads each card speficied, one at a time, storing the data (after conversion to volts) in the variable specified. A control word is sent, after the last card is read, that has SYE mode bit on, to unit 0. The call subroutine then returns to the main program.

SPECIAL CONSIDERATIONS

1. The call subroutine assumes the 69421A is a standard ±10 volt card. To modify the program for other options, line 7 must be changed as follows:

(for option 102) $0.05p(p39+1) \rightarrow p(p39+1)$
(for option 001) $0.0005p(p39+1) \rightarrow p(p39+1)$

USED WITH

69421A Voltage Monitor, 69422A High Speed A/D.

DO NOT USE WITH

69423A Low Level A/D and Scanner.

PROGRAM LISTINGS

GPIO

```
0: "Analog Input":
1: "AI":gsb "Parameter Check"
2: wtb "MP",170260+dtopp0
3: 1→p39
4: wti 4,10000+dtopp39;rdb("MP")→p(p39+1)
5: if p(p39+1)>7777;p(p39+1)-100000→p(p39+1)
6: otdp(p39+1)→p(p39+1);if p(p39+1)>2047;p(p39+1)-4096→p(p39+1)
7: .005p(p39+1)→p(p39+1)
8: if (p39+2→p39)<=p0-1;gto -4
9: wtb "MP",170040;mdec
10: ret
+21145
```

Bytes used: 334

HP-IB

```
0: "Analog Input":
1: "AI":gsb "Parameter Check"
2: wrt "MP.2","0",260+dtopp0,"T"
3: 1→p39
4: wrt "MP.2",char(64+pp39),"T",char(64+pp39),"X";red "MP",p(p39+1)
5: if p(p39+1)>7777;p(p39+1)-10000→p(p39+1)
6: otdp(p39+1)→p(p39+1);if p(p39+1)>2047;p(p39+1)-4096→p(p39+1)
7: .005p(p39+1)→p(p39+1)
8: if (p39+2→p39)<=p0-1;gto -4
9: wrt "MP.1","00040T"
10: ret
+6040
```

Bytes used: 368

File 16: Single Ended Scan (SE)

Scans and read voltages from multiplexed dc sources. Measured voltages in decimal form are read from 69421A or 69422A cards. The 69330A or 69433A cards are used to scan the dc voltages to be measured. All readings are stored in r-variables. The voltages scanned must share a single common or ground wire.

SYNTAX

cll 'Single Ended Scan' (Sv, R, C, [U])

or

cll 'SE' (Sv, R, C, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
Sv	p1	Voltage Monitor Slot Address	None
R	p2	No. of Readings	None
C	p3	Start Channel	Unit 0
U	p4	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Single Ended Scan' (403, 98, 14, 0)

or

cll 'Single Ended Scan' (3, 98, 14)

cll 'SE' (403, 98, 14)

In the first example, the long descriptive label and the complete slot addresses (403) is used, the number of readings and the starting channel are specified, and the unit number is included. In the next example, the shortened slot address (3) is used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1	Card slot address for the Voltage Monitor Card, 69421A or 69422A High Speed A/D card. The value must be from 400 to 413 or from 0 (zero) to 13, where 1 is understood to be slot address 401, 2 is 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot address must be in decimal. The first Relay Output card, 69330A or Relay Output/Readback Card, 69433A must be in the next higher slot address. All relay cards used for scanning must be next to one another.
p2	The total number of readings must not exceed 168, since that is the maximum obtainable from one mainframe (14 cards). Minimum value is one reading.
p3	Start channel number can be any value from zero to 168.

Note: Program will adjust maximum limit for start channel plus number of readings to reflect slot address for first relay card.

PROGRAM DESCRIPTION

The subprogram starts with a "cll" to 'Parameter Check'. Parameters sent are the card slot address for the Voltage Monitor, a dummy variable (zero) and the unit address. If either the slot address (p1) or the unit number (p4) are out of limits, the program will stop and wait for correct values to be entered.

Upon returning from 'Parameter Check', the program checks the maximum number of readings registered. If more than the available number of readings are requested, the program stops and waits until the number of readings requested is corrected. Next, the starting channel is checked and if it is not between zero and the available number of channels, the program again stop and waits until the start channel number is corrected. The final check is on the total of start channel plus total number of readings. If this number exceeds the available number of channels, the program again stops and waits until the total is corrected. The available number of channels is determined by assuming there are relay cards in all card slots to the right of the Voltage Monitor card.

All relays are now programmed open to prevent two sources from being shorted together. Scanning is then begun by calculating the first card and first relay to be closed. Since a control word with DTE, SYE and TME mode bits has been programmed, the program will wait for each relay to settle before programming the control word with ISL, SYE, and TME mode bits on. Once this control word is programmed, the Voltage Monitor card is then addressed and read with a gate. The reading is converted to volts and stored in an r-variable, rp39; where p39 represents the channel number. The relay that was closed for the last reading is now opened by sending a control word with DTE, SYE and TME mode bits and sending zero data to the appropriate relay card. The program then repeats the relay closure, voltage measurement, relay open cycle until all channels specified have been read. After the final reading, a control word with SYE mode bit on is sent to unit 0, the 6940B, before returning to the main program.

SPECIAL CONSIDERATIONS

1. All relay cards used for scanning must be mounted in the card slots immediately to the right of the Voltage Monitor card.
2. Cards such as 69330A, 69331A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subprogram.
3. The call subprogram assumes the 69421A is a standard ±10 volt card. To modify the program for other options, line 19 must be changed as follows:

(for option 102) .05rp39 → rp39
(for option 001) .0005rp39 → rp39

4. Voltage D/A's (69321B's) and Current D/A's (69370A's) that were previously addressd when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.

USED WITH (See Figure 3-1)

A/D Cards: 69421A Voltage Monitor, 69422A High Speed A/D

Scanner Cards: 69330A Relay Output, 69433A Relay Output/Readback

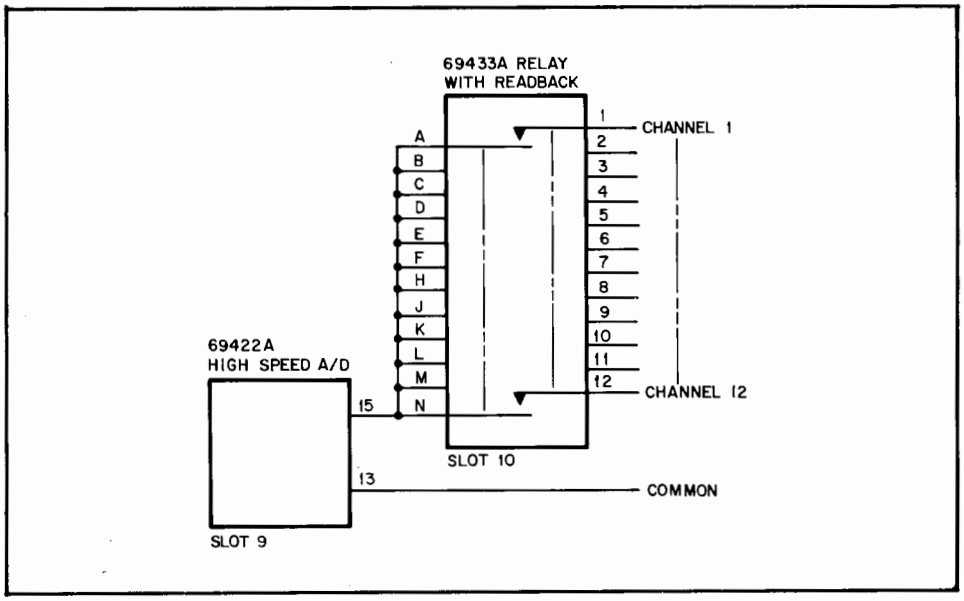


Figure 3-1. Single Ended Scan Connection Diagram

File 16: Single Ended Scan (SE)

PROGRAM LISTINGS

GPIO

```
0: "Single Ended Scan":
1: "SE";cll 'Parameter Check'(p1,0,p4);12*(p1+1)→p5
2: if p2<1 or p2>168-p5;prt "Total readings,";p2;"out of range";stp ;jmp 0
3: if p3<0 or p3>168-p5;prt "Starting channel,";p3;"out of range";stp ;jmp 0
4: if p2+p3>168-p5;prt "Last channel,";p2+p3;">168";stp ;jmp -2
5: wtb "MP";170140+dtop4
6: prnd(.49+(p2+p3)/12,0)→p6
7: 1→p39
8: wtb "MP";10000*dto(p1+p39)
9: if (p39+1→p39)<=p6;sto -1
10: wtb "MP";170160+dtop4
11: p3→p39
12: prnd(.51+p39/12,0)→p5
13: wtb "MP";10000*dto(p1+p5)+dto(2↑(p39mod12))
14: wtb "MP";170260+dtop4
15: wti 4;10000*dtop1;rdb("MP")→rp39
16: wtb "MP";170160+dtop4;10000*dto(p1+p5)
17: if rp39>7777;rp39-100000→rp39
18: otdrp39→rp39;if rp39>2047;rp39-4096→rp39
19: .005rp39→rp39
20: if (p39+1→p39)<p2+p3;sto -8
21: wtb "MP";170040;mdec
22: ret
*19031
```

Bytes used: 816

HP-IB

```
0: "Single Ended Scan":
1: "SE";cll 'Parameter Check'(p1,0,p4);12*(p1+1)→p5
2: if p2<1 or p2>168-p5;prt "Total readings,";p2;"out of range";stp ;jmp 0
3: if p3<0 or p3>168-p5;prt "Starting channel,";p3;"out of range";stp ;jmp 0
4: if p2+p3>168-p5;prt "Last channel,";p2+p3;">168";stp ;jmp -2
5: wrt "MP.2","0",140+dtop4,"T"
6: prnd(.49+(p2+p3)/12,0)→p6
7: 1→p39
8: wrt "MP.2";char(64+p1+p39),"T"
9: if (p39+1→p39)<=p6;sto -1
10: wrt "MP.2","0",160+dtop4,"T"
11: p3→p39
12: prnd(.51+p39/12,0)→p5
13: wrt "MP.2";char(64+p1+p5),dto(2↑(p39mod12)),"T"
14: wrt "MP.2","0",260+dtop4,"T"
15: wrt "MP.2";char(64+p1),"TX";red "MP";rp39
16: wrt "MP.2","0",160+dtop4,"T";char(64+p1+p5),"T"
17: if rp39>7777;rp39-10000→rp39
18: otdrp39→rp39;if rp39>2047;rp39-4096→rp39
19: .005rp39→rp39
20: if (p39+1→p39)<p2+p3;sto -8
21: wrt "MP.1","00040T"
22: ret
*29652
```

Bytes used: 872

File 17: Double Ended Scan (DE)

Scans and reads voltages from multiplexed DC sources. Measured voltages in decimal form are read from 69421A or 69422A cards. The 69330A or 69433A cards are used to scan the DC voltages to be measured. The voltages scanned need not share a single common or ground wire. All readings are stored in r-variables.

SYNTAX

cll 'Double Ended Scan' (Sv, R, C,)

or

cll 'DE' (Sv, R, C, U)

PARAMETERS

Parameter List	Variable	Description	Default Value
Sv	p1	Voltage Monitor Slot Address	None
R	p2	No. of Readings	None
C	p3	Start Channel	None
U	p4	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Double Ended Scan' (403, 98, 14, 0)

or

cll 'Double Ended Scan' (3, 98, 14)

or

cll 'DE' (403, 98, 14)

In the first example, the long descriptive label and the complete slot address (403) is used, the number of readings and the starting channel are speficied, and the unit number is included. In the next example, the shortened slot address (3) is used and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

File 17: Double Ended Scan (DE)

VARIABLE DESCRIPTION

p1	Card slot address for the Voltage Monitor Card, 69421A or 69422A High Speed A/D card. The value must be from 400 to 413 or from 0 (zero) to 13, where 1 is understood to be slot address 401, 2 is 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot address must be in decimal. The first Relay Output card, 69330A or Relay Output/Readback Card, 69433A must be in the next higher slot address. All relay cards used for scanning must be next to one another.
p2	The total number of readings must not exceed 84, since that is the maximum obtainable from one mainframe (14 cards). Minimum value is one reading.
p3	Start channel number can be any value from zero to 84.

Note: Program will adjust maximum limit for start channel plus number of readings to reflect slot address for first relay card.

PROGRAM DESCRIPTION

The subprogram starts with a "cll" to 'Parameter Check'. Parameters sent are the card slot address for the Voltage Monitor, a dummy variable (zero) and the unit address. If either the slot address (p1) or the unit number (p4) are out of limits, the program will stop and wait for correct values to be entered.

Upon returning from 'Parameter Check', the program checks the maximum number of readings requested. If more than the available number of readings are requested, the program stops and waits until the number of readings requested is corrected. Next, the starting channel is checked and if it is not between zero and the available number of channels, the program again stop and waits until the start channel number is corrected. The final check is on the total of start channel plus total number of readings. If this number exceeds the available number of channels, the program again stops and waits until the total is corrected. The available number of channels is determined by assuming there are relay cards in all card slots to the right of the Voltage Monitor card.

All relays are now programmed open to prevent two sources from being shorted together. Scanning is then begun by calculating the first card and first relay to be closed. Since a control word with DTE, SYE and TME mode bits has been programmed, the program will wait for each relay to settle before programming the control word with ISL, SYE, and TME mode bits on. Once this control word is programmed, the Voltage Monitor card is then addressed and read with a gate. The reading is converted to volts and stored in an r-variable, rp39; where p39 represents the channel number. The relay that was closed for the last reading is now opened by sending a control word with DTE, SYE and TME mode bits and sending zero data to the appropriate relay card. The program then repeats the relay closure, voltage measurement, relay open cycle until all channels specified have been read. After the final reading, a control word with SYE mode bit on is sent to unit 0, the 6940B, before returning to the main program.

SPECIAL CONSIDERATIONS

1. All relay cards used for scanning must be mounted in the card slots immediately to the right of the Voltage Monitor card.
2. Cards such as 69330A, 69331A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subprogram.
3. The call subprogram assumes the 69421A is a standard ±10 volt card. To modify the program for other options, line 20 must be changed as follows:

(for option 102) .05rp39 → rp39
(for option 001) .0005rp39 → rp39

4. Voltage D/A's (69321B's) and Current D/A's (69370A's) that were previously addressd when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.

USED WITH (See Figure 3-2)

A/D Cards: 69421A Voltage Monitor, 69422A High Speed A/D

Scanner Cards: 69330A Relay Output, 69433A Relay Output/Readback

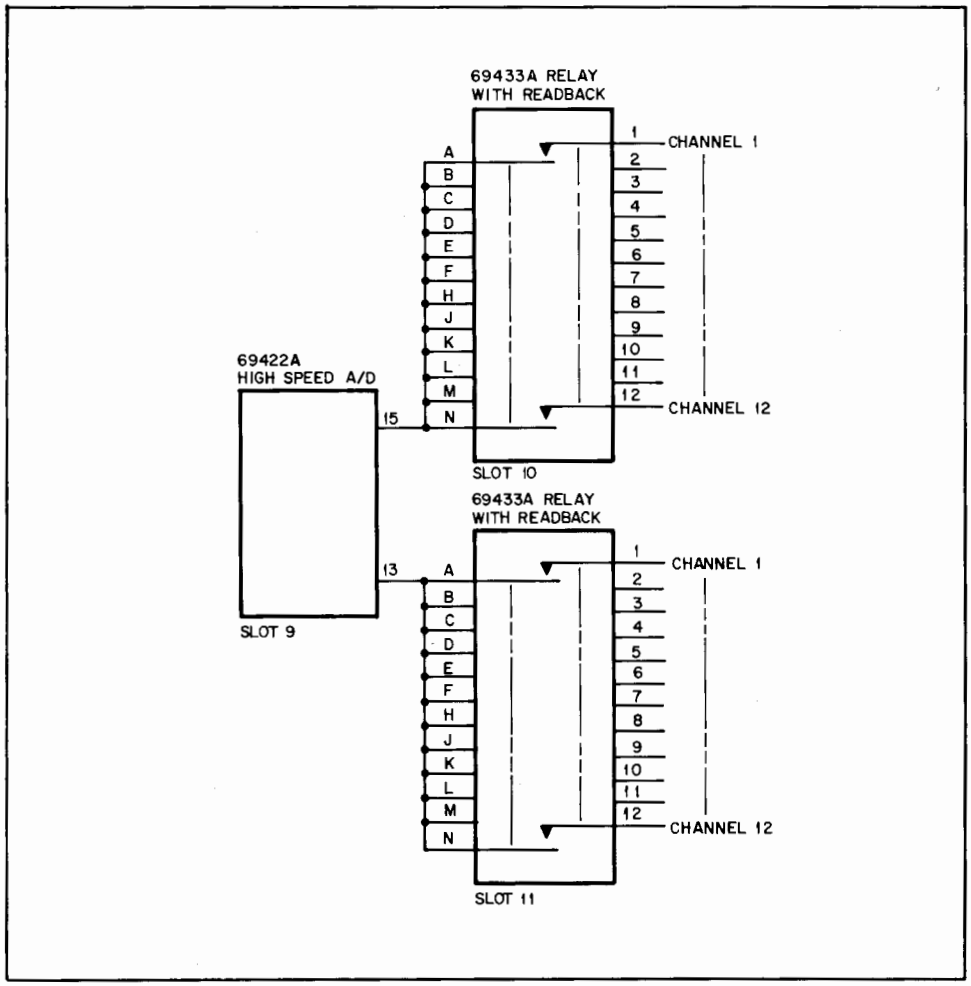


Figure 3-2. Double Scan Connection Diagram

File 17: Double Ended Scan (DE)

PROGRAM LISTINGS

GPIO

```
0: "Double Ended Scan":
1: "DE":cll 'Parameter Check'(p1;0;p4);12*int((14-p1)/12)→p5
2: if p2<1 or p2>84-p5;prt "Total readings;";p2;"out of range";stp ;jmp 0
3: if p3<0 or p3>84-p5;prt "Starting channel;";p3;"out of range";stp ;jmp 0
4: if p2+p3>84-p5;prt "Last channel;";p2+p3;">84";stp ;jmp -2
5: wtb "MP";170140+dtop4
6: 2*prnd(.49+(p2+p3)/12;0)→p6
7: 2*prnd(.49+p3/12;0)→p7
8: 1→p39
9: wtb "MP";10000*dto(p1+p39)
10: if (p39+1→p39)<=p6;gto -1
11: wtb "MP";170160+dtop4
12: p3→p39
13: 2*prnd(.51+p39/12;0)-1→p5;dto(2↑(p39mod12))→p8
14: wtb "MP";10000*dto(p1+p5)+p8;10000*dto(p1+p5+1)+p8
15: wtb "MP";170260+dtop4
16: wti 4;10000*dtop1;rdb("MP")→rp39
17: wtb "MP";170140+dtop4;10000*dto(p1+p5);10000*dto(p1+p5+1);170160+dtopp0
18: if rp39>7777;rp39-100000→rp39
19: otdrp39→rp39;if rp39>2047;rp39-4096→rp39
20: .005rp39→rp39
21: if (p39+1→p39)<p2+p3;gto -8
22: wtb"MP";170040;mdec
23: ret
*26992
```

Bytes used: 934

HP-IB

```
0: "Double Ended Scan":
1: "DE":cll 'Parameter Check'(p1;0;p4);12*int((14-p1)/12)→p5
2: if p2<1 or p2>84-p5;prt "Total readings;";p2;"out of range";stp ;jmp 0
3: if p3<0 or p3>84-p5;prt "Starting channel;";p3;"out of range";stp ;jmp 0
4: if p2+p3>84-p5;prt "Last channel;";p2+p3;">84";stp ;jmp -2
5: wrt "MP.2","0";140+dtop4,"T"
6: 2*prnd(.49+(p2+p3)/12;0)→p6
7: 2*prnd(.49+p3/12;0)→p7
8: 1→p39
9: wrt "MP.2";char(64+p1+p39),"T"
10: if (p39+1→p39)<=p6;gto -1
11: wrt "MP.2","0";160+dtop4,"T"
12: p3→p39
13: 2*prnd(.51+p39/12;0)-1→p5;dto(2↑(p39mod12))→p8
14: wrt "MP.2";char(64+p1+p5),p8,"T";char(64+p1+p5+1),p8,"T"
15: wrt "MP.2","0";260+dtop4,"T"
16: wrt "MP.2";char(64+p1),"TX";red "MP";rp39
17: wrt "MP.2","0";160+dtop4,"T";char(64+p1+p5),"T";char(64+p1+p5+1),"T"
18: if rp39>7777;rp39-10000→rp39
19: otdrp39→rp39;if rp39>2047;rp39-4096→rp39
20: .005rp39→rp39
21: if (p39+1→p39)<p2+p3;gto -8
22: wrt "MP.1","00040T"
23: ret
*7677
```

Bytes used: 984

File 19: Motor Output (MO)

Sends signed decimal data to the card specified. Positive numbers will cause stepping motors to rotate in one direction, and negative numbers in the reverse direction. Data greater than ±2047 steps will cause a stop.

SYNTAX

cll 'Motor Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'MO' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data in Signed Decimal	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Motor Output' (401, A, 407, G, 402, B, 0)

or

cll 'Motor Output' (1, 100, 7, 500, 2, −1000)

or

cll 'MO' (401, A, 407, 500, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc.	All even number variables are data (signed decimal)to be sent to output cards. Data variables are in milliamps. If these variables are more positive than +2047 or more negative than −2047, the program will stop and permit entry at a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram checks the data for sign and magnitude and converts the data to signed octal. A control word is sent to the unit designated by pp0 after pp0 is converted to octal (dto pp0). The converted data is then sent to the appropriate card. After all cards specified in the call have been addressed and given new data, the call subprogram sends a control word with SYE mode bit on to unit zero. This call program does not wait for all cards to time out prior to returning to the main program.

SPECIAL CONSIDERATIONS

1. Other call subprograms that turn on the TME mode bit (e.g., Serial Output) will not complete until the stepper motors programmed with Motor Output have timed out.
2. "Time Out" routine may be used to have system await motor completion.
3. Parameters out of range (e.g., more than 2047 steps programmed) will cause a halt. The program will not continue until the data is within range.

USED WITH

69335A Stepping Motor Control Card.

PROGRAM LISTINGS

GPIO

```
0: "Motor Output":
1: "MO":
2: gsb "Parameter Check"
3: 1→p39
4: if abs(p(p39+1))>2047;fxd 0;dsp "Data;p";p39+1;" out of range";stp ;jmp 0
5: if p(p39+1)<0;dtoabs(p(p39+1))+4000→p42;jmp 2
6: if p(p39+1)>=0;dtop(p39+1)→p42
7: wtb "MP";170040+dtopp0
8: wtb "MP";10000*dtopp39+p42;if (p39+2→p39)<=p0-1;gto -4
9: wtb "MP";170040;mdec
10: ret
*28807
```

Bytes used: 330

HP-IB

```
0: "Motor Output":
1: "MO":
2: gsb "Parameter Check"
3: 1→p39
4: if abs(p(p39+1))>2047;fxd 0;dsp "Data;p";p39+1;";out of range";stp ;jmp 0
5: if p(p39+1)<0;dtoabs(p(p39+1))+4000→p42;jmp 2
6: if p(p39+1)>=0;dtop(p39+1)→p42
7: wrt "MP.2";"O";40+dtopp0;"T"
8: wrt "MP.2";char(64+pp39);p42;"T";if (p39+2→p39)<=p0-1;gto -4
9: wrt "MP.1";"00040T"
10: ret
*20291
```

Bytes used: 344

File 20: Set Counter (SC)

Presets specified counters to the decimal count sent as data. Maximum count that card can be set to is 4095.

SYNTAX

cll 'Set Counter' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'SC' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data in Decimal	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Set Counter' (401, A, 407, G, 402, B, 0)

or

cll 'Set Counter' (1, 1000, 7, 2000, 2, 3000)

or

cll 'SC' (401, A, 407, 2000, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc.	All even number variables are decimal data to be sent to output cards. If these variables are greater than 4095, the program will stop and permit entry at a new value.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The specified counters are set to the value specified by the data parameters. The data is converted from decimal to octal before being sent to the card. After all cards have been set, the program sends a control word with SYE mode bit on to unit zero, the 6940B. The program then returns to the main program.

SPECIAL CONSIDERATIONS

1. Maximum number that the counters can be set to is 4095 (decimal).

USED WITH

69435A Pulse Counter.

PROGRAM LISTINGS

GPIO

```
0: "Set Counter":
1: "SC":gsb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: if p(p39+1)>=0 and p(p39+1)<4096;ato +2
5: beep;fxd 0;dsp "data p",p39,"out of range";stp ;ato -1
6: wtb "MP",10000*dtopp39+dtop(p39+1)
7: if (p39+2→p39)<=p0-1;ato -3
8: wtb "MP",170040
9: mdec;ret
*18986
```

Bytes used: 270

HP-IB

```
0: "Set Counter":
1: "SC":gsb "Parameter Check"
2: wrt "MP.2","0",40+dtopp0,"T"
3: 1→p39
4: if p(p39+1)>=0 and p(p39+1)<4096;ato +2
5: beep;fxd 0;dsp "data p",p39+1,"out of range";stp ;ato -1
6: wrt "MP.2",char(64+pp39),dtop(p39+1),"T"
7: if (p39+2→p39)<=p0-1;ato -3
8: wrt "MP.1","00040T"
9: ret
*23927
```

Bytes used: 290

File 21: Read Counter (RC)

Reads the specified counter cards. Returned data is in decimal form and is stored in the specified variables.

SYNTAX

cll 'Read Counter' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'RC' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Read Counter' (401, A, 407, G, 402, B, 0)

or

cll 'Read Counter' (1, A, 7, G, 2, B)

or

cll 'RC' (401, A, 407, G, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, the data storage locations are specified, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc. — All even numbered variables are used to store data from input cards. Data will be returned in decimal.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits ISL and SYE are programmed on by this control word. The cards specified are read. Their data is converted to decimal and stored in the variables specified by the call. After the last data item has been converted and stored, the program sends a control word with SYE mode bit on to unit zero, the 6940B, before returning to the main program.

SPECIAL CONSIDERATIONS

1. The 69435A Counter Card is read "on-the-fly", therefore, there is a finite possibility that the data will be read as the count is being updated.

USED WITH

69435A Pulse Counter Card.

PROGRAM LISTINGS

GPIO

```
0: "Read Counter":
1: "RC":gsb "Parameter Check"
2: wtb "MP",170240+dtopp0
3: 1→p39
4: wti 4,10000*dtopp39;otdrdb("MP")→p(p39+1)
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170040;mdec
7: ret
*11761
```

Bytes used: 186

HP-IB

```
0: "Read Counter":
1: "RC":gsb "Parameter Check"
2: wrt "MP.2","0";240+dtopp0,"T"
3: 1→p39
4: wrt "MP.2";char(64+pp39),"X";red "MP",p(p39+1);otdp(p39+1)→p(p39+1)
5: if (p39+2→p39)<=p0-1;gto -1
6: wrt "MP.1","00040T"
7: ret
*31288
```

Bytes used: 224

File 22: Timer Pulse (TP)

Sends the time (in seconds) to the card specified. Time interval resolution is 0.001 seconds.

SYNTAX

cll 'Timer Pulse' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'TP' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data in Seconds	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Timer Pulse' (401, A, 407, G, 402, B, 0)

or

cll 'Timer Pulse' (1, 1.50, 7, 3.75, 2, 4.00)

or

cll 'TP' (401, A, 407, 3.75, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 22: Timer Pulse (TP)

p2, p4, etc.	All even number variables are data in seconds, to be sent to output cards. If these variables are greater than 4.095, the program will stop and permit entry at a new value. Resolution is one millisecond.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sets the range parameter, p38, to 0.001. This corresponds to inserting jumper W2-D or connecting pins 11 and 4 on the card edge connector. To change resolution to 100 microseconds, change the range parameter, p38, to 0.0001 and connect W2-E on pins 11 and 3 on the edge connector.

A control word with DTE and SYE mode bits is then sent to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Each card specified is then programmed for the appropriate time interval. After the last card has been programmed and without waiting for any cards to time out, a control word with SYE mode bit on is sent to unit zero, the 6940B, before returning to the main program.

SPECIAL CONSIDERATIONS

1. Maximum time interval is 4.095 seconds.
2. Cards such as 69331A, 69330A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subprogram.
3. Voltage D/A cards (69321B's) and current D/A cards (69370A's) that were previously addressed when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.
4. The system can be forced to wait for the time interval pulse to complete if jumper W3 is installed on 69600B card and the "Time Out" subprogram is called immediately after the "Timer Pulse" call.

USED WITH

69600B Programmable Timer.

PROGRAM LISTINGS

GPIO

```
0: "Timer Pulse":
1: "TP":gsb "Parameter Check"
2: .001→p38;1→p39
3: if p(p39+1)>=.001 and p(p39+1)<=4.095;gto +2
4: fxd 0;beep;dsp "Data p";p39+1;"out of range";stp ;gto -1
5: wtb "MP",170140+dtopp0
6: wtb "MP",10000+dtopp39+int(dto(p(p39+1)/p38))
7: if (p39+2→p39)<=p0-1;gto -4
8: wtb "MP",170040;mdec;ret
+32236
```

Bytes used: 292

File 22: Timer Pulse (TP)

HP-IB

```
0: "Timer Pulse":
1: "TP":gsb "Parameter Check"
2: .001→p38;1→p39
3: if p(p39+1)>=.001 and p(p39+1)<=4.095;gto +2
4: fxd 0;beep;dsp "Data p";p39+1;"out of range";stp ;gto -1
5: wrt "MP.2","O";140+dtopp0;"T"
6: wrt "MP.2";char(64+pp39);int(dto(p(p39+1)/p38));"T"
7: if (p39+2→p39)<=p0-1;gto -4
8: wrt "MP.1","040T";ret
*4688
```

Bytes used: 306

File 23: Relay Output (RO)

Changes the state of the desired relay (1 through 12) on the specified card. Data must be in the range of 1 to 12 or −12 to −1. Relays are closed for positive numbers and opened for negative numbers.

SYNTAX

cll 'Relay Output' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'RO' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Relay Output' (401, A, 407, G, 402, B, 0)

or

cll 'Relay Output' (1, 4, 7, −3, 2, 6)

or

cll 'RO' (401, A, 407, −3, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 23: Relay Output (RO)

p2, p4, etc.	All even number variables are data to be sent to output cards. The data specifies which individual relay is to change state. Positive data will cause that relay to close, while negative data will cause that relay to open. Data must be in the range of −12 to −1 or 1 to 12.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). Mode bits ISL and SYE are programmed on by this control word. Each card is read to determine the present status of relays. Then, using the present status and the relay to be changed, the relay card is updated. After all cards have been updated and have timed out, the subprogram sends a control word with SYE mode bit on. A return is then made to the main program.

SPECIAL CONSIDERATIONS

1. Data represents the connector pin associated with a specific relay and not the bit number (e.g., data = 2 will close a relay between connector pins 2 and B).
2. Cards such as 69331A, 69330A that have previously been addressed when DTE was off will send an external gate when DTE is programmed on in this call subprogram.
3. Voltage D/A's (69321B's) and Current D/A cards (69370A's) that were previously addressed when DTE was off will update their second rank of storage and their outputs when DTE is programmed on in this call subprogram.

USED WITH

69433A Relay/Readback.

PROGRAM LISTINGS

GPIO

```
0: "Relay Output":
1: "RO":gsb "Parameter Check"
2: 1→p39
3: if abs(p(p39+1))>0 and abs(p(p39+1))<13;gto +2
4: fxd 0;dsp "Data p",p39+1,"out of range";stp ;gto -1
5: wtb "MP",170240+dtopp0
6: wti 4,10000*dtopp39;rdi 4→p38
7: shf(1,1-abs(p(p39+1)))→p37
8: if p(p39+1)>0;band(7777,ior(p37,p38))→p37;gto +2
9: band(band(p38,cmp37),7777)→p37
10: wtb "MP",170160+dtopp0,10000*dtopp39+p37
11: if (p39+2→p39)<=p0-1;gto -8
12: wtb "MP",170040;mdec;ret
+10222
```

Bytes used: 430

File 23: Relay Output (RO)

HP-IB

```
0: "Relay Output":
1: "RO":gsb "Parameter Check"
2: 1→p39
3: if abs(p(p39+1))>0 and abs(p(p39+1))<13;gto +2
4: fxd 0;beep;dsp "Data p",p39+1,"out of range";stp ;gto -1
5: wrt "MP.2","O",240+dtopp0,"T";moct
6: wrt "MP.2",char(64+pp39),"X";red 723,p38
7: shf(1,1-abs(p(p39+1)))→p37
8: if p(p39+1)>0;band(7777,ior(p37,p38))→p37;gto +2
9: band(band(p38,cmpp37),7777)→p37
10: wrt "MP.2","O",160+dtopp0,"T",char(64+pp39),p37,"T"
11: if (p39+2→p39)<=p0-1;gto -8
12: wrt "MP.1","040T";mdec;ret
*365
```

Bytes used: 466

File 24: Time Interval (TI)

Returns the decimal value of an external pulse. The time base (69601B) is assumed to be 100kHz.

SYNTAX

cll 'Time Interval' (S_0, D_0, [U])

or

cll 'TI' (S_0, D_0, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1	Card Slot Address	None
D_n	p2	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None
	p4	Reference Frequency	100000

PROGRAM EXAMPLES

cll 'Time Interval' (401, A, 0)

or

cll 'Time Interval' (1, A)

or

cll 'TI' (401, A)

In the first example, the long descriptive label and the complete slot address (401)is used, the data storage location is specified, and the unit number is included. In the next example, the shortened slot address (1) is used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1 — Used for slot address. The value must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2 — Used to store data from input card. Data is returned in decimal (seconds).

p4	Must equal the external frequency reference applied to pin E of the 69435A edge connector.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram requires two cards: the 69435A Pulse Counter and 69601B Frequency Reference Cards (see the connection diagram of Figure 3-3). The purpose of this subprogram is to time a pulse applied to the 69435A count-up enable input, pin A of the edge connector. The width of the pulse can vary from microseconds to about one hour (4000 seconds). See the chart on Figure 3-3 for appropriate connections.

The subprogram starts with a go sub (gsb) to 'Parameter Check'. The Parameter Check subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check'. this subprogram sends a control word, with SYE mode bit on, to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram also checks the number of parameters passed. If less than three are passed, the program displays: "No return variable in cll statement", and then stops.

If enough parameters are passed, the call subprogram sends a control word with ISL and SYE mode bits on. Then it addresses and reads the 69435A Pulse Counter card. While waiting for a non-zero reading, the program reads the 69435A and displays:

"Waiting for pulse"

A wait of approximately 50 milliseconds occurs after the first non-zero reading from the 69435A. This wait is to ensure that the pulse being measured has completed. The wait (approximately 50msec) is based on the assumption that the reference frequency from the 69601B is 100kHz. Therefore, p4 is 100000 as indicated by the chart in the Time Interval Connection Diagram. Longer pulses require two changes:

1. Connection of a different reference frequency.
2. Change in the variable p4.

For example:

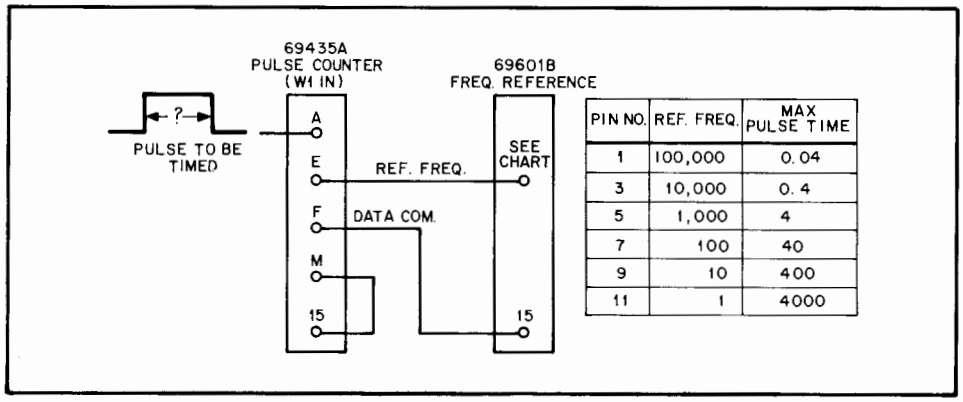
A time interval of approximately 0.6 seconds must be timed. The pulse is connected to pin A of the 69435A Counter Card edge connector. Then, pin E of the 69435A is connected to pin 5 of the 69601B Frequency Reference card edge connector. This connection will give a reference frequency of 1000Hz and therefore a resolution of one millisecond. Line 2 in the subprogram must be changed from 100000 → p4 to 1000 → p4. With this connection, pulses from milliseconds to 4 seconds can be measured.

SPECIAL CONSIDERATIONS

1. Reference Frequncy parameter, p4, is 100000, therefore the reference frequency is assumed to be 100kHz. If a different reference frequency is desired, p4 can be modified to correspond with the proper frequency.
2. Pulse to be measured is TTL compatible.

USED WITH

69435A Pulse Counter, 69601B Frequency Reference.



PIN NO.	REF. FREQ.	MAX PULSE TIME
1	100,000	0.04
3	10,000	0.4
5	1,000	4
7	100	40
9	10	400
11	1	4000

Figure 3-3. Time Interval Connection Diagram

PROGRAM LISTINGS

GPIO

```
0: "Time Interval":
1: "TI":gsb "Parameter Check"
2: 100000→p4
3: if p0<3;dsp "No return variable in cll statement";beep;end
4: wtb "MP",170040+dtopp0,10000*dtop1,170240+dtopp0,10000*dtop1
5: dsp "Waiting for pulse";if rdi 4=0;gto +0
6: dsp ""
7: wait 10+5e6/p4;rdi 4→p2
8: otdp2/p4→p2
9: wtb "MP",170040;mdec;ret
*9706
```

Bytes used: 296

HP-IB

```
0: "Time Interval":
1: "TI":gsb "Parameter Check"
2: 100000→p4
3: if p0<3;dsp "No return variable in cll statement";beep;end
4: wrt "MP.2","0",40+dtopp0,"T";char(64+p1),"T"
5: wrt "MP.2","0",240+dtopp0,"T";char(64+p1),"Z"
6: dsp "Waiting for pulse";red "MP",p2;if p2=0;gto +0
7: dsp ""
8: wait 10+5e6/p4;red "MP",p2;otdp2/p4→p2
9: wrt "MP.1","040T";ret
*19515
```

Bytes used: 340

File 25: Frequency Measurement (FM)

Returns the decimal value of an external frequency source. The program adjusts the sample interval (from 69600B) until the best resolution is achieved.

SYNTAX

cll 'Frequency Measurement' (S_0, D_0, [U])

or

cll 'FM' (S_0, D_0, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1	Card Slot Address	None
D_n	p2	Storage Location for Frequency Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Frequency Measurement' (401, A, 0)

or

cll 'Frequency Measurement' (1, A)

or

cll 'FM' (401, A)

In the first example, the long descriptive label and the complete slot address (401) is used, the data storage location is specified, and the unit number is included. In the next example, the shortened slot address (1) is used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1 — Used for slot address of 69435A Pulse Counter card. The value must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot address must be in decimal.

p2 — Used to store the results of the frequency measurement. Data is in decimal representing the number of hertz.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

p40 — Sample time interval for the 69600B Programmable Timer card. Value is stepped from 10 to 100 to 1000 milliseconds by the subprogram.

PROGRAM DESCRIPTION

This subprogram requires two cards: the 69435A Pulse Counter and 69600B Programmable Timer card (see the connection diagram of Figure 3-4). The purpose of this subprogram is to measure the frequency (in hertz) of an external frequency source connected to the 69435A card.

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word, with SYE mode bit on, to the unit designated by pp0. The pulse counter card is then preset to zero and the 69600B Programmable Timer card programmed for a 10 millisecond pulse. The 69600B is assumed to be in the slot immediately to the right of the 69435A card. If the data read back from the 69435A is less than 144 octal (100 decimal) and if the gate time variable, p40, is less than 1000 milliseconds, the program will repeat the measurement. Repeating the measurement improves the resolution and accuracy of the results.

Once the reading from the 69435A has enough resolution, the data is converted to frequency. Gate time variable, p40, is converted to seconds (0.001 * p40) and the count, in decimal, is divided by the gate time. Before returning to the main program, the subprogram sends a control word, with SYE mode bit on, to unit zero, the 6940B.

SPECIAL CONSIDERATIONS

1. The 69600B must be located in the slot immediately to the right of the 69435A.
2. Connections are those given in the Frequency Measurement Connection Diagram.
3. The 69600B must be modified by the user to have jumper W6 out and jumper W3 in.
4. Unknown frequency source is capable of providing drive current (5-10 milliamps) for count inputs.

USED WITH

69435A Pulse Counter Card, 69600B Programmable Timer.

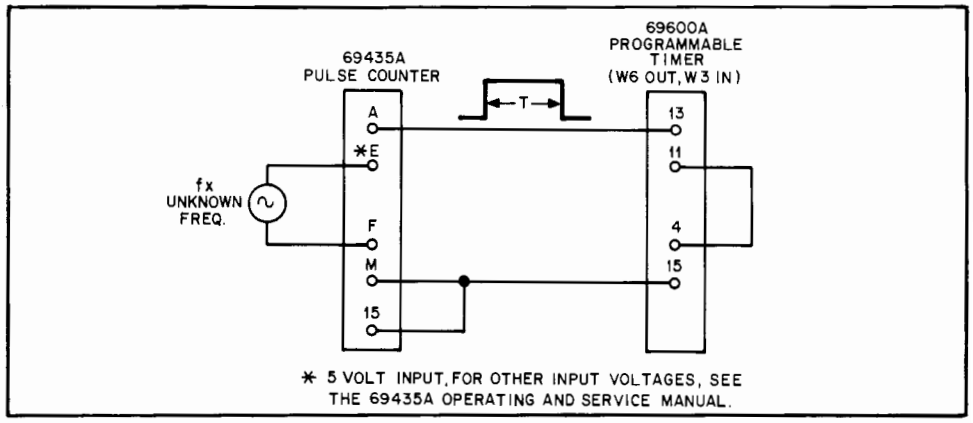


Figure 3-4. Frequency Measurement Connection Diagram

File 25: Frequency Measurement (FM)

PROGRAM LISTINGS

GPIO

```
0: "Frequency Measurement":
1: "FM":esb "Parameter Check"
2: 10→p40
3: wtb "MP",170040+dtopp0,10000*dtop1
4: wtb "MP",170160+dtopp0,10000*(dtop1+1)+dtop40,170240
5: wti 4,10000*dtop1;rdi 4→p2
6: if p2<144 and p40<1000;10p40→p40;gto -3
7: otdp2/(.001*p40)→p2
8: wtb "MP",170040;mdec;ret
*20794
```

Bytes used: 292

HP-IB

```
0: "Frequency Measurement":
1: "FM":esb "Parameter Check"
2: 10→p40
3: wrt "MP.2","0",40+dtopp0,"T",char(64+p1),"T"
4: wrt "MP.2","0",160+dtopp0,"T",char(64+p1+1),dtop40,"T","0",r240+dtopp0,"T"
5: wrt "MP",char(64+p1),"T";red 723,p2
6: if p2<144 and p40<1000;10p40→p40;gto -3
7: otdp2/(.001*p40)→p2
8: wrt "MP.1","040T";ret
*2300
```

Bytes used: 336

File 26: Option 40 Voltage Control (VC)

Sends desired output voltage (0 to 100V) in decimal form to the option 40 power supply programming cards specified.

SYNTAX

cll 'Option 40 Voltage Control' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'VC' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Option 40 Voltage Control' (401, A, 407, G, 402, B, 0)

or

cll 'Option 40 Voltage Control' (1, 10, 7, 20, 2, 30)

or

cll 'VC' (401, A, 407, 20, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 26: Option 40 Voltage Control (VC)

p2, p4, etc.	All even number variables are data to be sent to output cards. The data is in decimal (0 to 100) and represents the desired output voltage in volts.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p38	Power supply LSB: 0.01 for power supplies of up to 40 volts output. 0.025 for power supplies of 41 to 100 volts maximum output.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The data (voltages) are converted to LSB's and programmed to the resistance programming card specified. For power supplies in the range of 0 to 40 volts, the LSB is 0.01 volts. For power supplies in the range of 41 to 100 volts, the LSB is 0.025 volts.

This program waits until each resistance programming card completes before programming the next card. When all the cards have been programmed, this routine turns on SYE at the Multiprogrammer and returns to the main program.

SPECIAL CONSIDERATIONS

The proper resistance programming card must be used to program the power supply being used (see Table 3-1).

USED WITH

69500A to 69506A Resistance Output Cards.

PROGRAM LISTINGS

GPIO

```
0: "Option 40 Voltage Control":
1: "VC":gsb "Parameter Check"
2: wtb "MP";170160+dtopp0
3: 1→p39
4: if p(p39+1)>-.001 and p(p39+1)<100;gto +2
5: beep;prt "Data p";p39+1;"out of range";stp ;gto -1
6: if p(p39+1)>40;.025→p38;gto +2
7: .01→p38
8: dtop(p39+1)/p38→p38
9: wtb "MP";10000dtopp39+p38
10: if (p39+2→p39)<=p0-1;gto -6
11: wtb "MP";170040;mdec
12: ret
+993
```

Bytes used: 362

File 26: Option 40 Voltage Control (VC)

HP-IB

```
0: "Option 40 Voltage Control":
1: "VC":esb "Parameter Check"
2: wrt "MP.2","O",160+dtopp0,"T"
3: 1→p39
4: if p(p39+1)>-.001 and p(p39+1)<100;sto +2
5: beep;prt "Data p",p39+1,"out of range";stp ;sto -1
6: if p(p39+1)>40;.025→p38;sto +2
7: .01→p38
8: dtop(p39+1)/p38→p38
9: wrt "MP.2",char(64+pp39),p38,"T"
10: if (p39+2→p39)<=p0-1;sto -6
11: wrt "MP.1","040T"
12: ret
*25715
```

Bytes used: 374

Table 3-1.
HP Power Supply (Equipped with Option 040) and Resistance Output Card Selection Table

HP POWER SUPPLY			VOLTAGE PROGRAMMING			CURRENT PROGRAMMING §		
Volts	Amps	Model	Resistance Card	Volt/Step	Accuracy	Resistance Card	Amp/Step	Accuracy ‡
0-7.5	0-5	6281A	69501A	10mV	1% ± 20mV	69501A	10mA	10% ± 65mA
0-8	0-1000	6464C	69501A	10mV	1.0% ± 5mV	69501A	2A	1.0% ± 2A
0-10	0-2	6113A	69504A	10mV	0.1% ± 2mV	69512A	40mA	◇
0-10	0-10	6282A	69501A	10mV	1% ± 20mV	69501A	20mA	10% ± 65mA
0-10	0-20	6256B	69501A	10mV	1.0% ± 10mV	69510A	0.4A	10% ± 85mA
0-10	0-50	6259B	69501A	10mV	1.0% ± 8mV	69510A	1A	10% ± 65mA
0-10	0-100	6260B	69501A	10mV	1.0% ± 10mV	69510A	2A	10% ± 205mA
0-16 or 0-18	0-600 or 0-500	6466C	69501A	10mV	1.0% ± 5mV	69501A	3A	1.0% ± 2A
0-20	0-0.6	6205B	2 Each 69501A	10mV	1% ± 20mV	Current Limit Not Adjustable		
0-40	0-0.3							
Dual Output; Dual Range								
0-20	0-1	6101A	69504A	10mV	0.1% ± 2mV	69512A	20mA	◇
0-20	0-1	6111A †	69504A	10mV	0.1% ± 2mV	69512A	20mA	◇
0-20	0-2	6104A ††	69505A	10mV	.025% ± 1mV	69512A	40mA	2% ± 5mA
0-40	0-1	6114A ††						
0-20	0-3	6284A	69501A	10mV	1% ± 20mV	69513A	60mA	10% ± 60mA
0-20	0-3	6253A *	2 Each 69501A	10mV	1% ± 20mV	69513A	60mA	10% ± 60mA
0-20	0-10	6263B	69501A	10mV	1.0% ± 9mV	69512A	0.2A	10% ± 50mA
0-20	0-20	6264B	69501A	10mV	1.0% ± 10mV	69510A	0.4A	10% ± 85mA
0-20	0-50	6261B	69501A	10mV	1.0% ± 8mV	69510A	1.0A	10% ± 65mA
0-24	0-3	6224B	69501A	10mV	1.0% ± 15mV	69513A	60mA	10% ± 35mA
0-25	0-1	6220B	69502A	25mV	1% ± 50mV	69512A	20mA	10% ± 20mA
0-25	0-2	6227B *	2 Each 69501A	10mV	1% ± 25mV	69512A	40mA	10% ± 15mA
0-36	0-300	6469C	69501A	10mV	1.0% ± 5mV	69501A	0.6A	1.0% ± 1A
0-40	0-0.5	6102A	69504A	10mV	0.1% ± 2mV	69512A	10mA	◇
0-40	0-0.5	6112A †	69504A	10mV	0.1% ± 2mV	69512A	10mA	◇
0-40	0-1.5	6289A	69501A	10mV	1% ± 20mV	69511A	30mA	10% ± 30mA
0-40	0-1.5	6255A*	2 Each 69501A	10mV	1% ± 20mV	69511A	30mA	10% ± 30mA
0-40	0-3	6265B	69501A	10mV	1.0% ± 7mV	69512A	60mA	10% ± 12mA
0-40	0-5	6266B	69501A	10mV	1.0% ± 8mV	69512A	0.1A	10% ± 20mA
0-40	0-10	6267B	69501A	10mV	1.0% ± 8mV	69512A	0.2A	10% ± 30mA
0-40	0-30	6268B	69501A	10mV	1.0% ± 8mV	69510A	0.60A	10% ± 40mA
0-40	0-50	6269B	69501A	10mV	1.0% ± 8mV	69510A	1.0A	10% ± 65mA
0-50	0-0.5	6220B	69502A	25mV	1% ± 25mV	69512A	10mA	10% ± 10mA
0-50	0-0.8	6105A ††	69506A	25mV	.025% ± 1mV	69511A	15mA	2% ± 5mA
0-100	0-0.4	6115A ††						
0-50	0-1	6228B *	2 Each 69502A	25mV	1% ± 25mV	69512A	20mA	10% ± 15mA
0-50	0-1.5	6226B	69502A	25mV	1.0% ± 15mV	69511A	30mA	10% ± 18mA
0-60	0-1	6294A	69502A	25mV	1% ± 50mV	69512A	20mA	10% ± 20mA
0-60	0-3	6271B	69502A	25mV	1.0% ± 6mV	69512A	60mA	10% ± 12mA
0-60	0-15	6274B	69502A	25mV	1.0% ± 7mV	69512A	0.3A	10% ± 50mA
0-64	0-150	6472C	69502A	25mV	1.0% ± 25mV	69501A	0.3A	1.0% ± 1A
0-100	0-0.2	6106A	69506A	50mV	.1% ± 5mV	69512A	4mA	◇
0-100	0-0.75	6299A	69502A	25mV	1% ± 15mV	69511A	15mA	10% ± 15mA

§ Each current programming card (models 69510A-69513A) provides two independent outputs, and thus can program the current output of two power supplies. The 69501A card has only one output and is used for current programming of certain high current supplies.

‡ The accuracy specification refers to the combination of the power supply and the programming card; it includes the effects of load related internal temperature changes in the supply.

◇ These supplies are constant voltage/current limit, rather than constant voltage/constant current.

† The 6111A and 6112A supplies are identical to the 6101A and 6102A supplies respectively, except for the addition of front panel thumbwheel voltage controls.

†† No special option 040 modifications are necessary for these dual range 6104A, 6105A, 6114A, and 6115A.

* Dual Output.

File 27: Option 40 Current Control (CC)

Sends desired current output (0 to 1.000 amps) in decimal form (e.g., 0.55 for 0.55A) to the option 40 power supply programming cards specified. Each card has two output channels (see program examples).

SYNTAX

cll 'Option 40 Current Control ' (S_0, D_0, S_0, D_1, S_1, D_2, S_1, D_3, - - - - S_n, D_n, [U])

or

cll 'CC' (S_0, D_0, S_0, D_1, S_1, D_2, S_1, D_3, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Option 40 Current Control' (401, A, 401, G, 407, B, 407, C, 0)

or

cll 'Option 40 Current Control' (1, .100, 1, .200, 7, .300, 7, .400)

or

cll 'CC' (401, A, 401, .200, 407, B, 407, C)

In the first example, the long descriptive label and the complete slot addresses (401, 407) are used, variable data is sent, and the unit number is included. Note that each slot address is repeated. The first data item (A) is used to program channel 1 of the card in slot 401. The second data item (G) is for channel 2 of the card in slot 401. The third data item (B) programs channel 1 of the next card specified, and so forth. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

File 27: Option 40 Current Control (CC)

VARIABLE DESCRIPTION

p1, p3, etc.	All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal and are repeated (see description below).
p2, p4, etc.	All even number variables are data to be sent to output cards. The data is in decimal (0.000 to 1.000) and represents the desired output current in amperes. The parameters are arranged in sets of four. First the card slot address, then the data to program channel 1, followed by a repeat of the card slot address and then the data to program channel 2.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The data (currents) are converted to LSB's and programmed to the resistance programming card specified.

The routine as written is intended for 1 amp supplies. A change must be made to program power supplies of other output ratings. Lines 4 and 8 contain the LSB of the power supply. For 1 amp power supplies this LSB is 0.02. For other power supplies, the "1" on lines 4 and 8 should be changed to the maximum rated output current. For example, to program a 2 amp supply, the "1" on lines 4 and 8 should be changed to a "2".

This program waits until each card has completed before programming the next card. When all cards have been programmed, this routine turns on SYE at the multiprogrammer and returns to the main program.

SPECIAL CONSIDERATIONS

1. The proper resistance programming card must be used to program the power supply being used (see Table 3-1).
2. Lines 4 and 8 must be modified if the supply is not rated for one amp: the "1" in these lines should be changed to correspond to the maximum output current of the particular supply.
3. The resistance cards are programmed by a pair of data: S_0, D_1, S_0, D_2. The first Address-Data pair for chanel one, the second pair for channel 2. This requires that each address be repeated twice: once for the channel one of the card and the second time for channel two. If only one channel of a card is to be programmed, a zero must be programmed to the other channel.

USED WITH

69510A to 69351A Resistance Output Cards.

File 27: Option 40 Current Control (CC)

PROGRAM LISTINGS

GPIO

```
0: "Option 40 Current Control":
1: "CC":esb "Parameter Check"
2: wtb "MP",170160+dtopp0
3: 1→p39
4: p(p39+1)/(1*.02)→p38
5: if p(p39+1)>-.001 and p(p39+1)<50;eto +2
6: beep;prt "Data p",p39+1,"out of range";stp ;eto -2
7: if p39+3>p0;0→p37;eto +4
8: p(p39+3)/(1*.02)→p37
9: if p37>-.001 and p37<=50;eto +2
10: beep;prt "Data p",p39+3,"out of range";stp ;eto -2
11: wtb "MP",10000dtopp39+dtop38+100dtop37
12: if (p39+4→p39)<=p0-1;eto -8
13: wtb "MP",170040;mdec
14: ret
*12388
```

Bytes used: 470

HP-IB

```
0: "Option 40 Current Control":
1: "CC":esb "Parameter Check"
2: wrt "MP.2","O",160+dtopp0,"T"
3: 1→p39
4: p(p39+1)/(1*.02)→p38
5: if p(p39+1)>-.001 and p(p39+1)<50;eto +2
6: beep;prt "Data p",p39+1,"out of range";stp ;eto -2
7: if p39+3>p0;0→p37;eto +4
8: p(p39+3)/(1*.02)→p37
9: if p37>-.001 and p37<=50;eto +2
10: beep;prt "Data p",p39+3,"out of range";stp ;eto -2
11: wrt "MP.1",char(64+pp39),dtop38+100dtop37,"T"
12: if (p39+4→p39)<=p0-1;eto -8
13: wrt "MP.1","040T"
14: ret
*17510
```

Bytes used: 482

File 28: Arm (AM)

Arms cards specified to permit an interrupt whenever any card detects an interrupt condition. (Assumes all W6 jumpers have been removed.)

SYNTAX

cll 'Arm' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'AM' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Dummy Variables	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Arm' (401, 0, 407, 0, 402, 0, 0)

or

cll 'Arm' (1, 0, 7, 0, 2, 0)

or

cll 'AM' (401, 0, 407, 0, 402, 0)

In the first example, the long descriptive label is used, the complete slot addresses (401, 407, etc.) of the cards to be armed are specified, dummy data values (zeros) are included, and the unit number is given. A dummy data value in the range from 0 to 7777 must be included after each slot address. If the dummy data is omitted, the specified cards will not be armed.

In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc.	All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even number variables are dummy variables, and must be in the range of zero through 7777.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The control word programs the ISL and SYE mode bits on. All cards specified are addressed and gated. After the last card is gated, a control word is sent with IEN, SYE and TME mode bits on. This control word places all 6940B/6941B mainframes in the "Interrupt Mode". All cards selectively armed with this call, are then capable of generating an interrupt which the 9825A can detect. The program then returns to the main program, after enabling the interrupt system of the 9825A (eir "MP"). Figures 3-5 (page 3-90) and 3-6 (page 3-91) are interrupt mode flow charts that illustrate the programming sequence for the "Arm" subprogram.

SPECIAL CONSIDERATIONS

1. Dummy data variables (0 to 7777) must be used in this subprogram.
2. These cards will become armed, unless jumper W6 is removed from all, when this call sends a control word with IEN, SYE and TME mode bits on.
3. Cards previously addressed when TME was off will generate an interrupt if they have not completed their time out prior to this call.

USED WITH

69431A Digital Input, 69434A Event Sense, 69436A Process Interrupt, 69600B Programmable Timer.

File 28: Arm (AM)

PROGRAM LISTINGS

GPIO

```
0: "Arm":
1: "AM":esb "Parameter Check"
2: wtb "MP",170240+dtopp0
3: 1→p39
4: wtb "MP",10000*dtopp39
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170460;mdec
7: eir "MP";ret
*4661
```

Bytes used: 166

HP-IB

```
0: "Arm":
1: "AM":esb "Parameter Check"
2: wrt "MP.2","0",240+dtopp0,"T"
3: 1→p39
4: wrt "MP.1",char(pp39+64),"T"
5: if (p39+2→p39)<=p0-1;gto -1
6: wrt "MP.1","00460T"
7: eir "HP-IB";ret
*20408
```

Bytes used: 184

File 29: Serial Poll (SP)

Performs a Serial Poll on cards specified. The data returned is in octal and includes an IRQ bit if the card generated an interrupt.

SYNTAX

cll 'Serial Poll' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'SP' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Serial Poll' (401, A, 407, G, 402, B, 0)

or

cll 'Serial Poll' (1, A, 7, G, 2, B)

or

cll 'SP' (401, A, 407, G, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, the data storage locations are specified, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 29: Serial Poll (SP)

p2, p4, etc.	All even number variables are used to store data from input cards. The IRQ bit is not removed in the call subroutine. Data will be returned in octal.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word, with ISL and SYE mode bits on, to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The program then addresses and reads (without a gate) all cards specified. The data, as well as any IRQ bits, are stored in the variables specified. When present, the IRQ bit indicates that the card interrupted. After the last card is polled, a control word with SYE mode bit on is sent to the unit zero. The program then returns to the main program. Figures 3-5 (page 3-90) and 3-6 (page 3-91) are interrupt mode flow charts that illustrate the programming sequence for the "Serial Poll" subprogram.

SPECIAL CONSIDERATIONS

The user program must be written to determine which return data contains an input request (IRQ) bit and therefore caused an interrupt. The following is a sample program using the interrupt mode subprograms.


```
0: oni 9;"ISR"
1: cll 'Recycle'(1;0;2;0)
2: cll 'Arm'(1;0;2;0)
                •
                •
                •
13: gto +0
14: "ISR":
15: cll 'Serial Poll'(1;A;2;B)
16: if A>7777;gto "N"
17: if B>7777;gto "P"
18: iret
19: "N":cll 'Disarm'(1;0)
20: prt A-100000
21: iret
22: "P":cll 'Disarm'(2;0)
23: prt B-100000
24: iret
```

For HP-IB change lines 0, 20, and 23 to:

```
0: oni 7, "ISR"; rds (723)-C
20: prt (A-10000)
23: prt (B-10000)
```

USED WITH

69431A Digital Input, 69434A Event Sense, 69436A Proccess Interrupt, 69600B Programmable Timer.

PROGRAM LISTINGS

GPIO

```
0: "Serial Poll":
1: "SP":esb "Parameter Check"
2: wtb "MP";170240+dtopp0
3: 1→p39
4: wti 4;10000*dtopp39
5: rdi 4→p(p39+1)
6: if (p39+2→p39)<=p0-1;sto -2
7: wtb "MP";170040;mdec
8: ret
*3710
```

Bytes used: 186

HP-IB

```
0: "Serial Poll":
1: "SP":esb "Parameter Check"
2: wrt "MP.2","O";240+dtopp0;"T"
3: 1→p39
4: wrt "MP.1";char(pp39+64);"X"
5: red "MP";p(p39+1)
6: if (p39+2→p39)<=p0-1;sto -2
7: wrt "MP.1","O0040T"
8: ret
*28995
```

Bytes used: 202

File 30: Disarm (DM)

Selectively disarms cards specified. Assumes all W6 jumpers have been removed.

SYNTAX

cll 'Disarm' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'DM' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Dummy Variables	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Disarm' (401, 0, 407, 0, 402, 0, 0)

or

cll 'Disarm' (1, 0, 7, 0, 2, 0)

or

cll 'DM' (401, 0, 407, 0, 402, 0)

In the first example, the long descriptive label is used, the complete slot addresses (401, 407, etc.) of the cards to be disarmed are specified, dummy data values (zeros) are included, and the unit number is given. A dummy data value in the range from 0 to 7777 must be included after each slot address. If the dummy data is omitted, the specified cards will not be disarmed.

In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc.	All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even number variables are dummy variables, and must be between zero and 7777.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The control word program the SYE mode bit on. All cards specified are sent zero data. The data is sent with a gate signal. A return to the main program is made after the last card is disarmed and a control word with SYE mode bit on is sent to unit zero, the 6940B. Figures 3-5 (page 3-90) and 3-6 (page 3-91) are interrupt mode flow charts that illustrate the programming sequence for the "Disarm" subprogram.

SPECIAL CONSIDERATIONS

1. Removal of jumper W6 from 69431A, 69434A, 69436A and 69600B cards is recommended. These cards will become armed when a control word with IEN, SYE and TME mode bits on is programmed, unless their W6 jumpers are removed.

USED WITH

69431A Digital Iput, 69434A Event Sense, 69436A Process Interrupt, 69600B Programmable Timer.

File 30: Disarm (DM)

PROGRAM LISTINGS

GPIO

```
0: "Disarm":
1: "DM":esb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: wtb "MP",10000*dtopp39
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170040
7: ret
*4890
```

Bytes used: 158

HP-IB

```
0: "Disarm":
1: "DM":esb "Parameter Check"
2: wrt "MP.2","0",40+dtopp0,"T"
3: 1→p39
4: wrt "MP.1",char(pp39+64),"T"
5: if (p39+2→p39)<=p0-1;gto -1
6: wrt "MP.1","00040T"
7: ret
*31345
```

Bytes used: 176

File 31 Time Out (TO)

Halts system until all output cards complete their operations.

SYNTAX

cll 'Time Out'

or

cll 'TO'

PARAMETERS

None required.

PROGRAM EXAMPLES

cll 'Time Out'

or

cll 'TO'

The first example uses the long descriptive title to call 'Time Out'. The second example uses the short label.

PROGRAM DESCRIPTION

The subprogram calls 'Parameter Check' to set up formats and modes necessary to "talk" to the Multiprogrammer. The subprogram then sends out a control word to the Multiprogrammer with TME, DTE, and SYE on. The system will now await any card, such as the Stepping Motor Card, that is in the middle of an operation to complete. When all cards finish their previously programmed operation, this subprogram turns SYE on and returns.

SPECIAL CONSIDERATIONS

1. Any card timing out in any mainframe will cause the system to wait when this subprogram is called.
2. This subprogram will work especially well if the user wishes to wait while a stepping motor is turning. First call 'Motor Output', then call 'Time Out' to wait for the motor to stop.

USED WITH

All output cards.

File 31: Time Out (TO)

PROGRAM LISTINGS

GPIO

```
0: "Time Out":
1: "TO":cll 'PC'(0)
2: wtb "MP",170160,170040
3: mdec;ret
*24250
```

Bytes used: 68

HP-IB

```
0: "Time Out":
1: "TO":cll 'PC'(0)
2: wrt "MP.1","00160T00040T"
3: ret
*18026
```

Bytes used: 68

File 32: Recycle (RE)

Recycles specified cards, before they are armed by the subprogram 'ARM' (File 28).

SYNTAX

cll 'Recycle' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'RE' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Card Data In Octal	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Recycle' (401, A, 407, G, 402, B, 0)

or

cll 'Recycle' (1, 5252, 7, 2525, 2, 7777)

or

cll 'RE' (401, A, 407, 77, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, variable data is sent, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent,and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, complete slot addresses are used, variable data and constant data are sent, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

File 32: Recycle (RE)

VARIABLE DESCRIPTION

p1, p3, etc.	All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even number variables are data to be sent to the cards. If these variables are greater than 7777, the program will stop and permit entry of a new value. Data must be in octal.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The data parameters represent the following:

69436A: Data is decimal representation of those bits to be cleared

69600B : Data is decimal representation of next time period. Note that the "Timer Pulse" subprogram may be used to recycle the 69600B.

SPECIAL CONSIDERATIONS

1. Removal of jumper W6 is recommended to enable individual arming of cards. See the proper Operating and Service Manual for the W6 location on the card.
2. At start up the 69436A must be initialized to reset any bits set on power up. To accomplish this, data of 4095 may be sent to all 69436A cards using this subprogram: "Recycle"

USED WITH

69436A Process Interrupt Card, 69600B Programmable Timer Card.

DO NOT USE WITH

69332A Open Collector Output, 69380A Breadboard Output, and 69601B Frequency Reference.

File 32: Recycle (RE)

PROGRAM LISTINGS

GPIO

```
0: "Recycle":
1: "RE":esb "Parameter Check"
2: wtb "MP",170040+dtopp0
3: 1→p39
4: wtb "MP",10000*dtopp39+dtop(p39+1)
5: if (p39+2→p39)<=p0-1;gto -1
6: wtb "MP",170040;mdec
7: ret
*16432
```

Bytes used: 176

HP-IB

```
0: "Recycle":
1: "RE":esb "Parameter Check"
2: fmt 1,c,z;fmt 2,c,f4.0,c,z;wrt "MP.2","0",40+dtopp0,"T"
3: 1→p39
4: wrt "MP.2",char(pp39+64),dtop(p39+1),"T"
5: if (p39+2→p39)<=p0-1;gto -1
6: wrt "MP.1","00040T"
7: ret
*11538
```

Bytes used: 216

File 33: Reference Word (RW)

Reads back the reference words from the desired Event Sense cards and sends new reference words to these cards.

SYNTAX

cll 'Reference Word' (S_0, D_0, S_0, D_1, S_1, D_2, S_1, D_3, - - - - S_n, D_n, [U])

or

cll 'RW' (S_0, D_0, S_0, D_1, D_2, S_1, D_3, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Reference Word	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Reference Word' (401, A, 401, G, 407, B, 407, C, 0)

or

cll 'Reference Word' (1, A, 1, 7777, 7, B, 7, 5252)

or

cll 'RW' (401, A, 401, G, 407, B, 407, C)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, and the unit number is included. Note that each slot address is repeated. The first data item (A) is the storage location which will receive the reference word read from the card in slot 401. The second data item (G) programs the new reference word to the card in slot 401. In the next example, shortened slot addresses (1, 7, etc.) are used, constant data is sent, for the new reference words, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The sub-program will automatically assign a value of zero to pp0 when the unit number is omitted.

File 33: Reference Word (RW)

VARIABLE DESCRIPTION

p1, p3, etc.	All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal and are repeated (see description below).
p2, p4, etc.	All even numbered parameters are data which represent the reference word in octal. The parameters are arranged in sets of four: First the card slot address, then the variable to receive the reference word read from the card, followed by a repeat of the card slot address and then the data to program the new reference word. If it is desired to retain the old reference word, the variable which receives the old reference word may be used to program the new one as: cll 'RW' (1, A, 1, A) In this example "A" will receive the old reference word, which will be returned to the card.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The first data parameter receives the old reference word, the second data parameter will become the card's new reference word. These can both be the same if the same variables are repeated in both locations.

If it is desired to change the reference word to equal the external word, the following procedure should be followed.

(1) Read the card using "Serial Poll" as in: cll 'SP' (1,A)
(2) Call 'Reference Word as in: cll 'RW' (1, X, 1, A-IRQ)
where, X is a dummy variable to receive the old reference word, 1 is the card address, A is the external word, and "IRQ" is used to subtract IRQ (HP-IB: IRQ = 10000, GPIO: IRQ = 100000)

Note that the external word is lost once "Reference Word" has been called. Figure 3-6 (page 3-91) is an interrupt mode flow chart that illustrates the programming sequence for the Reference Word subprogram.

SPECIAL CONSIDERATIONS

The "Reference Word" subprogram uses two address-data pairs for each card as: "cll 'RW' (1, A, 1, B)" where "1" is the card address, "A" is the variable to receive the old reference word and "B" is the new reference word to be programmed.

USED WITH

69434A Event Sense Card.

PROGRAM LISTINGS

GPIO

```
0: "Reference Word":
1: "RW":ssb "Parameter Check"
2: 1→p39
3: wtb "MP",170240+dtopp0;wti 4,10000*dtopp39
4: rdb("MP")→p(p39+1)
5: if p(p39+1)>7777;p(p39+1)-100000→p(p39+1)
6: if p39+3>p0;sto +2
7: wtb "MP",170040+dtopp0,10000*dtop(p39+2)+p(p39+3)
8: if (p39+4→p39)<p0-1;sto -5
9: wtb "MP",170040;mdec;ret
*8492
```

Bytes used: 306

HP-IB

```
0: "Reference Word":
1: "RW":ssb "Parameter Check"
2: 1→p39
3: wrt "MP.2","O";240+dtopp0,"T";char(64+pp39),"X"
4: red "MP";p(p39+1)
5: if p(p39+1)>7777;p(p39+1)-10000→p(p39+1)
6: if p39+3>p0;sto +2
7: wrt "MP.2","O";40+dtopp0,"T";char(p(p39+2)),p(p39+3),"T"
8: if (p39+4→p39)<p0-1;sto -5
9: wrt "MP.1","00040T";ret
*17881
```

Bytes used: 322

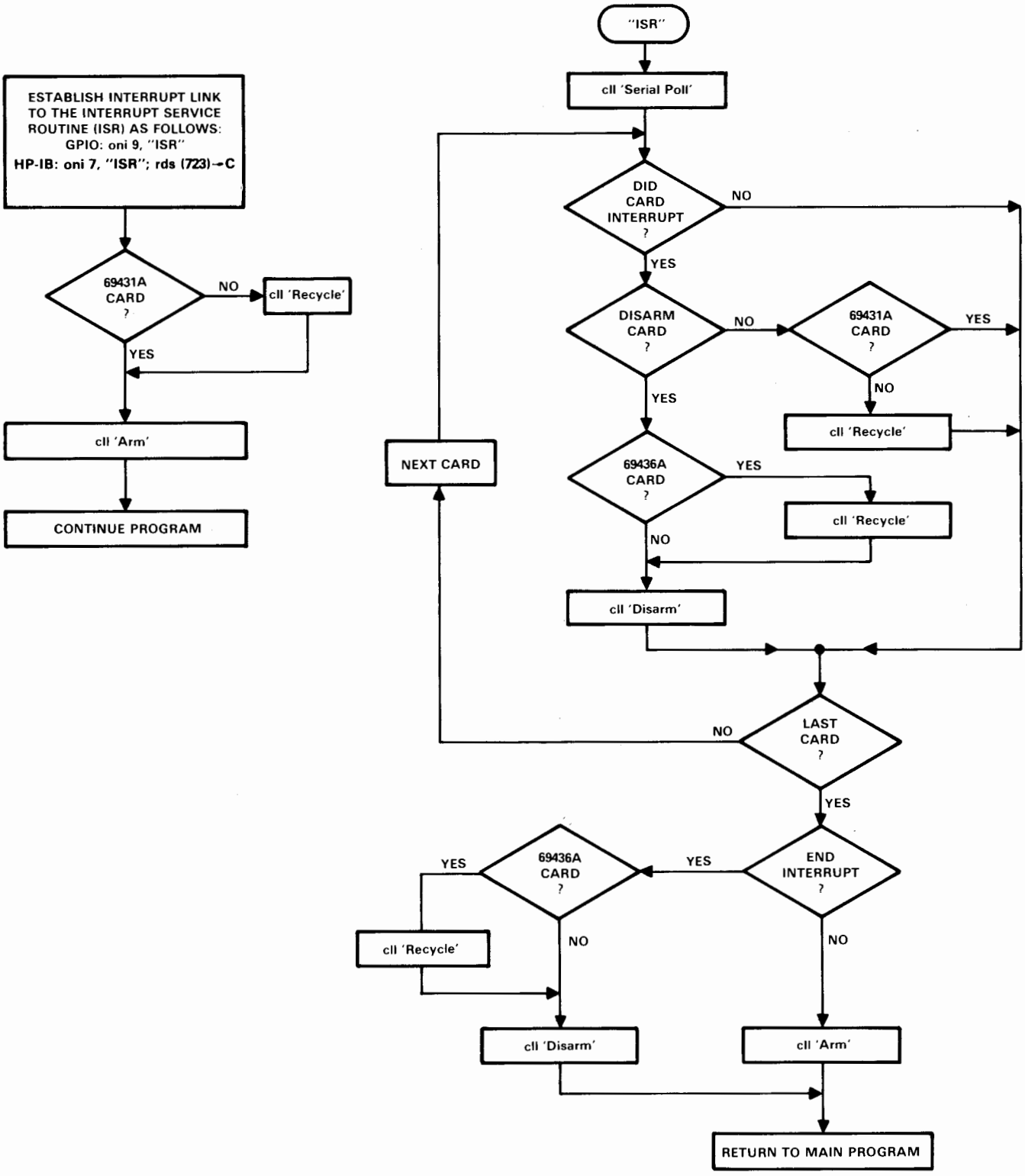


Figure 3-5. Interrupt Mode Flow Chart for the 69431A Digital Input, 69436A Process Interrupt, and 69600B Programmable Timer Cards.

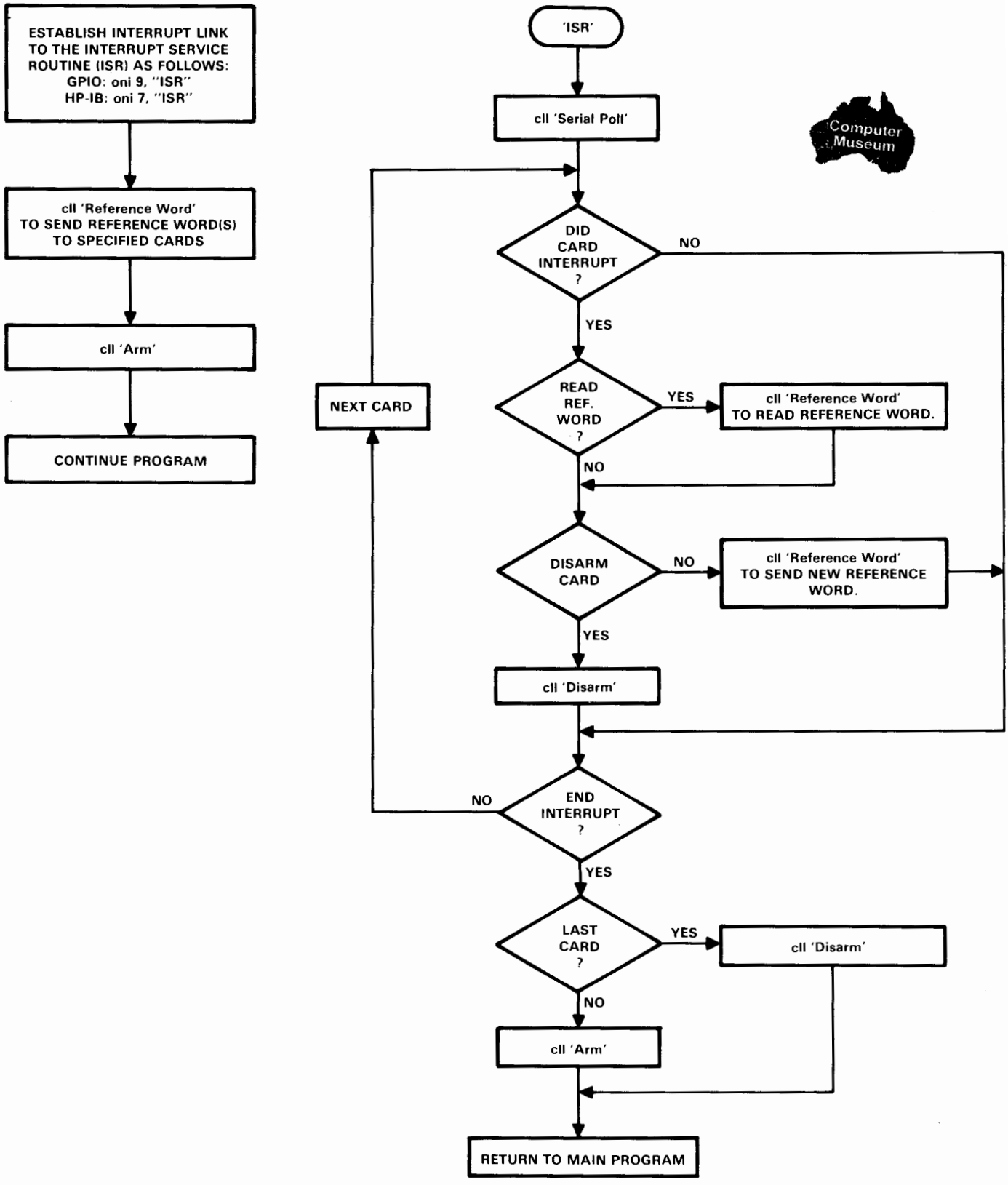


Figure 3-6. Interrupt Mode Flow Chart for the 69434A Event Sense Card.

File 34: T Thermocouple (TT)

Reads the temperature in °C from 69423A cards and type T thermocouples (copper versus copper-nickel alloy). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'T Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'TT' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'T Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'T Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'TT' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to the left of the decimal point and a channel to the right. The slot address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers run from 1 to 6. Thus, in order to read channel

	5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even numbered variables are used to store data from the input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p60-62	Constants to convert from reference junction voltage to °C.
p50-53	Constants to convert from reference junction °C to correction voltage of type T thermocouples.
p40-44	Constants to convert from type T thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type T thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form:

$p50 + X\,(p51 + X\,(p52 + X\,p53)) = \mu V$

where, $p50 = a_0$, $p51 = a_1$, $p52 = a_2$, $p53 = a_3$.

The conversion equation from volts from the thermocouple junction to °C is of the form:

$p40 + X\,(p41 + X\,(p42 + X\,(p43 + p44))) = °C$

where, $p40 = a_0$, $p41 = a_1$, $p42 = a_2$, $p43 = a_3$, $p44 = a_4$.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS — 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
0-400°C	−.15 to .17°C

For other ranges, change the p40-p44 constants to the values given in Table 3-2 for the applicable temperature range.

Table 3-2. Type T Thermocouples — Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0, a_1E, + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (°C)	a_0		a_1		a_2		a_3		a_4		Error Range (°C)
	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Error-Approx.
-270 to 0	. . . . .	. .	4.3553379	-3	-2.0325426	-5	-5.4720813	-9	-5.0865527	-13	-8 to 6
-200 to 0	. . . . .	. .	2.3837090	-2	-2.9878839	-6	-7.1945810	-10	-1.0041943	-13	-.3 to .3
-200 to 400	. . . . .	. .	2.6792411	-2	-1.0370271	-6	6.1330327	-11	-1.3988385	-15	-6 to 5
0 to 400	. . . . .	. .	2.5661297	-2	-6.1954869	-7	2.2181644	-11	-3.5500900	-16	-.15 to .17

File 34: T Thermocouple (TT)

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type T thermocouples (Copper vs. Copper-Nickel Alloy)

PROGRAM LISTINGS

GPIO

```
0: "T Thermocouple":
1: "TT":ssb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 38.709457→p51;3.7085566e-2→p52;5.649552e-5→p53
4: 2.5661297e-2→p41;-6.1954869e-7→p42;2.2181644e-11→p43;-3.55009e-16→p44
5: 1→p39
6: wtb "MP";170160+dtopp0
7: if (10frc(pp39)→p38)<7;sto +2
8: beep;prt "Address p";p39;"out of range";stp ;sto -1
9: wtb "MP";10000dtoint(pp39)+7;170260+dtopp0
10: wti 4,10000dtoint(pp39);rdi 4→p37
11: (otdp37-4096)*10/4352→p37
12: p60/(p61+ln(-p37))-p62→p37
13: p37(p51+p37(p52+p37p53))1e-6→p37
14: wtb "MP";170160+dtopp0;10000dtoint(pp39)+p38;170260+dtopp0
15: wti 4,10000dtoint(pp39);rdi 4→p36
16: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
17: (p37+p36)1e6→p36
18: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
19: if (p39+2→p39)<=p0-1;sto -13
20: wtb "MP";170040;mdec
21: ret
+18707
```

Bytes used: 810

HP-IB

```
0: "T Thermocouple":
1: "TT":ssb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 38.709457→p51;3.7085566e-2→p52;5.649552e-5→p53
4: 2.5661297e-2→p41;-6.1954869e-7→p42;2.2181644e-11→p43;-3.55009e-16→p44
5: 1→p39
6: wrt "MP.2","0";160+dtopp0;"T"
7: if (10frc(pp39)→p38)<7;sto +2
8: beep;prt "Address p";p39;"out of range";stp ;sto -1
9: wrt "MP.2";char(64+int(pp39));7;"T","0";260+dtopp0;"T"
10: wrt "MP.1";char(64+int(pp39));"X";red "MP";p37
11: (otdp37-4096)*10/4352→p37
12: p60/(p61+ln(-p37))-p62→p37
13: p37(p51+p37(p52+p37p53))1e-6→p37
14: wrt "MP.2","0";160+dtopp0;"T";char(int(pp39)+64);p38;"T"
15: wrt "MP.2","0";260+dtopp0;"T";char(64+int(pp39));"X";red "MP";p36
16: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
17: (p37+p36)1e6→p36
18: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
19: if (p39+2→p39)<=p0-1;sto -13
20: wrt "MP.1","040T"
21: ret
+12725
```

Bytes used: 862

File 35: J Thermocouple (JT)

Reads the temperature in °C from 69423A cards and type J thermocouples (iron versus copper-nickel alloy). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'J Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'JT' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'J Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'J Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'JT' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to the left of the decimal point and a channel to the right. The slot address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers run from 1 to 6. Thus, in order to read channel 5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot

address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc. — All even numbered variables are used to store data from the input cards.

pp0 — If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

p60-62 — Constants to convert from reference junction voltage to °C.

p50-53 — Constants to convert from reference junction °C to correction voltage of type J thermocouples.

p40-44 — Constants to convert from type J thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type J thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form: μV

$p50 + X\,(p51 + X\,(p52 + X\,p53)) = mV$

where, $p50 = a_0$, $p51 = a_1$, $p52 = a_2$, $p53 = a_3$.

The conversion equation from volts from the thermocouple junction to °C is of the form:

$p40 + X\,(p41 + X\,(p42 + X\,(p43 + p44))) = °C$

where, $p40 = a_0$, $p41 = a_1$, $p42 = a_2$, $p43 = a_3$, $p44 = a_4$.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS — 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
−20 to 500°C	−.07 to .06°C

For other ranges, change the p40-p44 constants to the values given in Table 3-3 for the applicable temperature range.

Table 3-3. Type J Thermocouples — Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0 + a_1E + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (C°)	a_0		a_1		a_2		a_3		a_4		Error Range (C°)
	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Exact-Approx.
-200 to 0	. . . . .		1.8843850	-2	-1.2029733	-6	-2.5278593	-10	-2.5849263	-14	-.4 to .5
-200 to 760	. . . . .		2.1155170	-2	-3.3513149	-7	1.2443997	-11	-1.5227150	-16	-6 to 7
-200 to 1200	. . . . .		2.1676850	-2	-2.1844464	-7	3.9094347	-12	-2.4303017	-17	-14 to 10
- 20 to 500	. . . . .		1.9745056	-2	-1.8094256	-7	7.8777919	-12	-1.1897222	-16	-.07 to .06
0 to 400	. . . . .		1.9750953	-2	-1.8542600	-7	8.3683958	-12	-1.3280568	-16	-.03 to .05
0 to 760	. . . . .		1.9323799	-2	-1.0306020	-7	3.7084018	-12	-5.1031937	-17	-.9 to .7
0 to 1200	. . . . .		1.8134974	-2	-5.6495930	-8	-2.4644023	-12	2.1141718	-17	-3 to 4
400 to 760	9.2808351	+1	5.4463817	-3	6.5254537	-7	-1.3987013	-11	9.9364476	-17	-.03 to .03
400 to 1200	-1.1075293	+2	2.8651303	-2	-2.9758175	-7	2.5945419	-12	-4.9012035	-18	-1.3 to 1.6
600 to 760	1.8020713	+2	-4.5284199	-3	1.0769294	-6	-2.1962321	-11	1.5521511	-16	-.001 to .001
760 to 1200	-6.3828680	+2	7.4068749	-2	-1.7177773	-6	2.1771293	-11	-9.9502571	-17	-.15 to .11

File 35: J Thermocouple (JT)

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type J thermocouples (Iron vs. Copper-Nickel Alloy)

PROGRAM LISTINGS

GPIO

```
0: "J Thermocouple":
1: "JT":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 50.373743→p51;3.0167011e-2→p52;-7.4293513e-5→p53
4: 1.97450056e-2→p41;-1.8094256e-7→p42;7.8777919e-12→p43;-1.1897222e-16→p44
5: 1→p39
6: wtb "MP",170160+dtopp0
7: if (10frc(pp39)→p38)<7;gto +2
8: beep;prt "Address p",p39,"out of range";stp ;gto -1
9: wtb "MP",10000dtoint(pp39)+7,170260+dtopp0
10: wti 4,10000dtoint(pp39);rdi 4→p37
11: (otdp37-4096)*10/4352→p37
12: p60/(p61+ln(-p37))-p62→p37
13: p37(p51+p37(p52+p37p53))1e-6→p37
14: wtb "MP",170160+dtopp0,10000dtoint(pp39)+p38,170260+dtopp0
15: wti 4,10000dtoint(pp39);rdi 4→p36
16: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
```

Bytes used: 814

HP-IB

```
0: "J Thermocouple":
1: "JT":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 50.373743→p51;3.0167011e-2→p52;-7.4293513e-5→p53
4: 1.97450056e-2→p41;-1.8094256e-7→p42;7.8777919e-12→p43;-1.1897222e-16→p44
5: 1→p39
6: wrt "MP.2","O",160+dtopp0,"T"
7: if (10frc(pp39)→p38)<7;gto +2
8: beep;prt "Address p",p39,"out of range";stp ;gto -1
9: wrt "MP.2",char(64+int(pp39)),7,"T","O",260+dtopp0,"T"
10: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
11: (otdp37-4096)*10/4352→p37
12: p60/(p61+ln(-p37))-p62→p37
13: p37(p51+p37(p52+p37p53))1e-6→p37
14: wrt "MP.2","O",160+dtopp0,"T",char(int(pp39)+64),p38,"T"
15: wrt "MP.2","O",260+dtopp0,"T",char(64+int(pp39)),"X";red "MP",p36
16: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
17: (p37+p36)1e6→p36
18: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
```

Bytes used: 864

File 36: E Thermocouple (ET)

Reads the temperature in °C from 69423A cards and type E thermocouples (nickel-chromium alloy versus copper-nickel alloy). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'E Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'ET' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'E Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'E Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'ET' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to the left of the decimal point and a channel to the right. The slot address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers run from 1 to 6. Thus, in order to read channel

	5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even numbered variables are used to store data from the input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p60-62	Constants to convert from reference junction voltage to °C.
p50-53	Constants to convert from reference junction °C to correction voltage of type E thermocouples.
p40-44	Constants to convert from type E thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type E thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form:

$p50 + X\ (p51 + X\ (p52 + X\ p53)) = \mu V$

where, $p50 = a_0$, $p51 = a_1$, $p52 = a_2$, $p53 = a_3$.

The conversion equation from volts from the thermocouple junction to °C is of the form:

$p40 + X\ (p41 + X\ (p42 + X\ (p43 + p44))) = °C$

where, $p40 = a_0$, $p41 = a_1$, $p42 = a_2$, $p43 = a_3$, $p44 = a_4$.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS – 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
−20 to 500°C	−.18 to .12°C

For other ranges, change the p40-p44 constants to the values given in Table 3-4 for the applicable temperature range.

Table 3-4. Type E Thermocouples – Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0, a_1E, + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (°C)	a_0		a_1		a_2		a_3		a_4		Error Range (°C)
	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Exact-Approx.
-270 to 0	. . . . .	. .	2.8168878	-3	-8.5940057	-6	-1.4930918	-9	-8.7987588	-14	-9 to 6
-200 to 0	. . . . .	. .	1.5726646	-2	-1.2102152	-6	-1.9577799	-10	-1.6696298	-14	-.3 to .3
-200 to 800	. . . . .	. .	1.8432856	-2	-3.2311582	-7	6.9795810	-12	-5.1106852	-17	-8 to 7
- 20 to 500	. . . . .	. .	1.6970287	-2	-2.0830603	-7	4.6512717	-12	-4.1805785	-17	-.18 to .12
0 to 400	. . . . .	. .	1.7022525	-2	-2.2097240	-7	5.4809314	-12	-5.7669892	-17	-.05 to .04
0 to 1000	. . . . .	. .	1.6410783	-2	-1.3560189	-7	1.8600342	-12	-8.5537337	-18	-.9 to 1.4
400 to 1000	1.9669452	+1	1.4207735	-2	-5.1844510	-8	5.6361365	-13	-1.5646343	-18	-.03 to .03
600 to 800	2.5192188	+1	1.3909529	-2	-4.7201133	-8	5.5638718	-13	-1.7775228	-18	-.0005 to .0005
850 to 1000	-7.1102114	+2	5.6554599	-2	-9.7013068	-7	9.3938146	-12	-3.3333675	-17	-.001 to .001

File 36: E Thermocouple (ET)

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type E thermocouples (Nickel-Chromium Alloy vs. Copper-Nickel Alloy)

PROGRAM LISTINGS

GPIO

```
0: "E Thermocouple":
1: "ET":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 58.6537565→p51;4.6720025e-2→p52;-1.4438022e-5→p53
4: 1.6970287e-2→p41;-2.0830603e-7→p42;4.6512717e-12→p43;-4.1805785e-17→p44
5: 1→p39
6: wtb "MP",170160+dtopp0
7: if (10frc(pp39)→p39)<7;gto +2
8: beep;prt "Address p",p39,"out of range";stp ;gto -1
9: wtb "MP",10000dtoint(pp39)+7,170260+dtopp0
10: wti 4,10000dtoint(pp39);rdi 4→p37
11: (otdp37-4096)*10/4352→p37
12: p60/(p61+ln(-p37))-p62→p37
13: p37(p51+p37(p52+p37p53))1e-6→p37
14: wtb "MP",170160+dtopp0,10000dtoint(pp39)+p38,170260+dtopp0
15: wti 4,10000dtoint(pp39);rdi 4→p36
16: ((otdp36→p36)-(p36)2047)4096)1e-5→p36
17: (p37+p36)1e6→p36
18: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
19: if (p39+2→p39)<=p0-1;gto -13
20: wtb "MP",170040;mdec
21: ret
*31034
```

Bytes used: 812

HP-IB

```
0: "E Thermocouple":
1: "ET":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 58.6537565→p51;4.6720025e-2→p52;-1.4438022e-5→p53
4: 1.6970287e-2→p41;-2.0830603e-7→p42;4.6512717e-12→p43;-4.1805785e-17→p44
5: 1→p39
6: wrt "MP.2","O";160+dtopp0;"T"
7: if (10frc(pp39)→p38)<7;gto +2
8: beep;prt "Address p",p39,"out of range";stp ;gto -1
9: wrt "MP.2",char(64+int(pp39)),7,"T","O";260+dtopp0;"T"
10: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
11: (otdp37-4096)*10/4352→p37
12: p60/(p61+ln(-p37))-p62→p37
13: p37(p51+p37(p52+p37p53))1e-6→p37
14: wrt "MP.2","O",160+dtopp0,"T",char(int(pp39)+64),p38,"T"
15: wrt "MP.2","O",260+dtopp0,"T",char(64+int(pp39)),"X";red "MP",p36
16: ((otdp36→p36)-(p36)2047)4096)1e-5→p36
17: (p37+p36)1e6→p36
18: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
19: if (p39+2→p39)<=p0-1;gto -13
20: wrt "MP.1","040T"
21: ret
*25052
```

Bytes used: 864

File 37: R Thermocouple (RT)

Reads the temperature in °C from 69423A cards and type R thermocouples (platinum-13% rhodium alloy versus platinum). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'R Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'RT' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'R Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'R Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'RT' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to the left of the decimal point and a channel to the right. The slot address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers run from 1 to 6. Thus, in order to read channel 5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

File 37: R Thermocouple (RT)

p2, p4, etc.	All even numbered variables are used to store data from the input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p60-62	Constants to convert from reference junction voltage to °C.
p50-53	Constants to convert from reference junction °C to correction voltage of type R thermocouples.
p40-44	Constants to convert from type R thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type R thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form:

$p50 + X (p51 + X (p52 + X p53)) = \mu V$

where, $p50 = a_0$, $p51 = a_1$, $p52 = a_2$, $p53 = a_3$.

The conversion equation from volts from the thermocouple junction to °C is of the form:

$p40 + X (p41 + X (p42 + X (p43 + p44))) = °C$

where, $p40 = a_0$, $p41 = a_1$, $p42 = a_2$, $p43 = a_3$, $p44 = a_4$.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS — 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
400-1100°C	−.04 to .04°C

For other ranges, change the p40-p44 constants to the values given in Table 3-5 for the applicable temperature range.

Table 3-5. Type R Thermocouples — Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0, a_1E, + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (°C)	a_0		a_1		a_2		a_3		a_4		Error Range (°C)
	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Exact-Approx.
- 50 to 900	. . . . .	. .	1.6251434	-1	-2.0454379	-5	2.5404935	-9	-1.1767904	-13	-13 to 3
0 to 1100	. . . . .	. .	1.5239494	-1	-1.3755675	-5	1.2610922	-9	-4.4281251	-14	- 4 to 7
0 to 1400	. . . . .	. .	1.4441607	-1	-9.5014952	-6	6.2073358	-10	-1.5622497	-14	- 6 to 10
0 to 1650	. . . . .	. .	1.3944190	-1	-7.4485484	-6	3.8266182	-10	-7.4517277	-15	- 7 to 13
0 to 1768	. . . . .	. .	1.3752883	-1	-6.7651171	-6	3.1420473	-10	-5.4254872	-15	- 7 to 14
400 to 1100	4.5509556	+1	1.1284875	-1	-2.8603978	-6	8.5173702	-11	-1.1440038	-15	-.04 to .04
400 to 1400	4.9160016	+1	1.1054589	-1	-2.3559046	-6	3.9276248	-11	3.3369324	-16	-.08 to .09
400 to 1650	4.8343651	+1	1.1098270	-1	-2.4353890	-6	4.5164488	-11	1.8172612	-16	-.10 to .12
1050 to 1400	-4.1134459	+0	1.2738464	-1	-4.3132296	-6	1.3863582	-10	-1.5283798	-15	-.002 to .002
1050 to 1650	3.7487318	+1	1.1519304	-1	-2.9827002	-6	7.4538667	-11	-3.7809957	-16	-.011 to .011
1400 to 1550	8.0559850	+1	1.0442877	-1	-1.9827500	-6	3.3603790	-11	2.4513433	-16	-.0005 to .0005
1400 to 1650	1.4180146	+2	9.0181346	-2	-7.4068329	-7	-1.4487255	-11	9.4290495	-16	-.0005 to .0005
1400 to 1768	3.1759093	+3	-5.8922431	-1	5.6190639	-5	-2.1303241	-9	3.0369250	-14	-.11 to .08
1666 to 1768	1.2883437	+4	-2.6747958	+0	2.2334214	-4	-8.0565860	-9	1.0882779	-13	-.0007 to .0007

File 37: R Thermocouple (RT)

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type R thermocouples (platinum-13% rhodium alloy vs. platinum)

PROGRAM LISTINGS

GPIO

```
0: "R Thermocouple":
1: "RT":esb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 5.2891411→p51;1.3844426e-2→p52;-2.0889531e-5→p53
4: 4.5509556→p40;.11284875→p41;-2.8603978e-6→p42;8.5173702e-11→p43
5: -1.1440038e-15→p44
6: 1→p39
7: wtb "MP";170160+dtopp0
8: if (10frc(pp39)→p38)<7;sto +2
9: beep;prt "Address p",p39,"out of range";stp ;sto -1
10: wtb "MP";10000dtoint(pp39)+7;170260+dtopp0
11: wti 4;10000dtoint(pp39);rdi 4→p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wtb "MP";170160+dtopp0;10000dtoint(pp39)+p38;170260+dtopp0
16: wti 4;10000dtoint(pp39);rdi 4→p36
17: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;sto -13
21: wtb "MP";170040;mdec
22: ret
*21837
```

Bytes used: 826

HP-IB

```
0: "R Thermocouple":
1: "RT":esb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 5.2891411→p51;1.3844426e-2→p52;-2.0889531e-5→p53
4: 4.5509556→p40;.11284875→p41;-2.8603978e-6→p42;8.5173702e-11→p43
5: -1.1440038e-15→p44
6: 1→p39
7: wrt "MP.2","0";160+dtopp0,"T"
8: if (10frc(pp39)→p38)<7;sto +2
9: beep;prt "Address p",p39,"out of range";stp ;sto -1
10: wrt "MP.2",char(64+int(pp39)),7,"T","0";260+dtopp0,"T"
11: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wrt "MP.2","0";160+dtopp0,"T",char(int(pp39)+64),p38,"T"
16: wrt "MP.2","0";260+dtopp0,"T",char(64+int(pp39)),"X";red "MP",p36
17: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;sto -13
21: wrt "MP.1","040T"
22: ret
*14758
```

Bytes used: 878

File 38: S Thermocouple (ST)

Reads the temperature in °C from 69423A cards and type S thermocouples (platinum - 10% rhodium alloy vs. platinum). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'S Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'ST' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'S Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'S Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'ST' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to the left of the decimal point and a channel to the right. The slot

address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers rum from 1 to 6. Thus, in order to read channel 5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.

p2, p4, etc.	All even numbered variables are used to store data from the input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p60-62	Constants to convert from reference junction voltage to °C.
p50-53	Constants to convert from reference junction °C to correction voltage of type S thermocouples.
p40-44	Constants to convert from type S thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type S thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form:
p50 + X (p51 + X (p52 + X p53)) = μV
where, p50 = a_0, p51 = a_1, p52 = a_2, p53 = a_3.

The conversion equation from volts from the thermocouple junction to °C is of the form:
p40 + X (p41 + X (p42 + X (p43 + p44))) = °C
where, p40 = a_0, p41 = a_1, p42 = a_2, p43 = a_3, p44 = a_4.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS — 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
400 to 1400°C	−.08 to .08°C

For other ranges, change the p40-p44 constants to the values given in Table 3-6 for the applicable temperature range.

Table 3-6. Type S Thermocouples — Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0, a_1E, + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (°C)	a_0		a_1		a_2		a_3		a_4		Error Range (°C)
	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Exact-Approx.
- 50 to 900	. . . . .	. .	1.6414048	-1	-2.0241757	-5	2.7849728	-9	-1.4172102	-13	-11 to 3
0 to 1100	. . . . .	. .	1.5445376	-1	-1.3349067	-5	1.3626587	-9	-5.3270847	-14	-3 to 6
0 to 1400	. . . . .	. .	1.4713897	-1	-9.0783455	-6	6.5660913	-10	-1.8499175	-14	-5 to 9
0 to 1650	. . . . .	. .	1.4260554	-1	-7.0073775	-6	3.8981279	-10	-8.3047780	-15	-6 to 11
0 to 1768	. . . . .	. .	1.4087955	-1	-6.3195007	-6	3.1267454	-10	-5.7422562	-15	-6 to 12
400 to 1100	4.1137317	+1	1.1599785	-1	-1.8642979	-6	1.2643267	-11	8.4828836	-16	-.05 to .07
400 to 1400	4.4507730	+1	1.1373998	-1	-1.3349811	-6	-3.9224680	-11	2.6563405	-15	-.08 to .08
400 to 1650	4.1670535	+1	1.1543356	-1	-1.6782780	-6	-1.0845801	-11	1.8379726	-15	-.2 to .2
1050 to 1400	-3.0938374	+1	1.4106560	-1	-4.9794442	-6	1.7334256	-10	-1.9262160	-15	-.003 to .003
1050 to 1650	1.2226507	+1	1.2706383	-1	-3.2873314	-6	8.3038098	-11	-1.3019379	-16	-.010 to .010
1400 to 1550	1.3866867	+2	9.3486676	-2	4.8592708	-8	-6.3885209	-11	2.2896541	-15	-.0005 to .0005
1400 to 1650	1.3923740	+2	9.3267401	-2	7.7266682	-8	-6.5458208	-11	2.3208160	-15	-.0005 to .0005
1400 to 1768	4.5133695	+3	-1.0046437	+0	1.0322002	-4	-4.3637046	-9	6.9361610	-14	-.13 to .10
1666 to 1768	2.3131446	+4	-5.4122671	+0	4.9347196	-4	-1.9681943	-8	2.9430179	-13	-.0005 to .0005

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type S thermocouples (platinum - 10% rhodium alloy vs. platinum)

PROGRAM LISTINGS

GPIO

```
0: "S Thermocouple":
1: "ST":esb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 5.3994446→p51;1.2467754e-2→p52;-1.9934168e-5→p53
4: 44.50779→p40;.11373998→p41;-1.3349811e-6→p42;-3.922468e-11→p43
5: 2.6563405e-15→p44
6: 1→p39
7: wtb "MP",170160+dtopp0
8: if (10frc(pp39)→p38)<7;gto +2
9: beep;prt "Address p",p39,"out of range";stp ;gto -1
10: wtb "MP",10000dtoint(pp39)+7,170260+dtopp0
11: wti 4,10000dtoint(pp39);rdi 4→p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wtb "MP",170160+dtopp0,10000dtoint(pp39)+p38,170260+dtopp0
16: wti 4,10000dtoint(pp39);rdi 4→p36
17: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;gto -13
21: wtb "MP",170040;mdec
22: ret
*13815
```

Bytes used: 826

File 38: S Thermocouple (ST)

HP-IB

```
0: "S Thermocouple":
1: "ST":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 5.3994446→p51;1.2467754e-2→p52;-1.9934168e-5→p53
4: 44.50779→p40;.11373998→p41;-1.3349811e-6→p42;-3.922468e-11→p43
5: 2.6563405e-15→p44
6: 1→p39
7: wrt "MP.2","0",160+dtopp0,"T"
8: if (10frc(pp39)→p38)<7;gto +2
9: beep;prt "Address p",p39,"out of range";stp ;gto -1
10: wrt "MP.2",char(64+int(pp39)),7,"T","0",260+dtopp0,"T"
11: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wrt "MP.2","0",160+dtopp0,"T",char(int(pp39)+64),p38,"T"
16: wrt "MP.2","0",260+dtopp0,"T",char(64+int(pp39)),"X";red "MP",p36
17: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;gto -13
21: wrt "MP.1","040T"
22: ret
*6736
```

Bytes used: 878

File 39: Thermocouple Reference (TR)

Reads the temperature in °C of the reference on the 69423A card.

SYNTAX

cll 'Thermocouple Reference' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'TR' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot Address	None
D_n	p2, p4, etc.	Storage Location for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'Thermocouple Reference' (401, A, 407, G, 402, B, 0)

or

cll 'Thermocouple Reference' (1, A, 7, G, 2, B)

or

cll 'TR' (401, A, 407, G, 402, B)

In the first example, the long descriptive label and the complete slot addresses (401, 407, etc.) are used, the data storage locations are specified, and the unit number is included. In the next example, shortened slot addresses (1, 7, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

File 39: Thermocouple Reference (TR)

VARIABLE DESCRIPTION

p1, p3, etc.	All odd numbered variables are used for slot addresses. Their values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. If this parameter is not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even number variables are used to store data from input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0).

This subprogram reads the reference on the specified 69423A and returns the temperature in °C. The reference junction is channel 7 on the 69423A.

SPECIAL CONSIDERATIONS

The temperature is returned in °C. To convert to °C use the following equation;

$$9/5 \times C + 32 = F$$

USED WITH

69423A Low Level Analog Input Card.

PROGRAM LISTINGS

GPIO

```
0: "Thermocouple Reference":
1: "TR":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 1→p39
4: wtb "MP",170160+dtopp0
5: wtb "MP",10000dtoint(pp39)+7,170260+dtopp0
6: wti 4,10000dtoint(pp39);rdi 4→p37
7: (otdp37-4096)*10/4352→p37
8: p60/(p61+ln(-p37))-p62→p(p39+1)
9: if (p39+2→p39)<=p0-1;gto -5
10: wtb "MP",170040;mdec
11: ret
*19655
```

Bytes used: 350

File 39: Thermocouple Reference (TR)

HP-IB

```
0: "Thermocouple Reference":
1: "TR":esb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 1→p39
4: wrt "MP.2","0",160+dtopp0,"T"
5: wrt "MP.2",char(64+int(pp39)),7,"T","0",260+dtopp0,"T"
6: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
7: (ot(dp37-4096)*10/4352→p37
8: p60/(p61+ln(-p37))-p62→p(p39+1)
9: if (p39+2→p39)<=p0-1;eto -5
10: wrt "MP.1","040T"
11: ret
*9872
```

Bytes used: 376

File 40: B Thermocouple (BT)

Reads the temperature in °C from 69423A cards and type B thermocouples (platinum - 30% rhodium alloy vs. platinum - 6% rhodium alloy). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'B Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'BT' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'B Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'B Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'BT' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to

	the left of the decimal point and a channel to the right. The slot address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers run from 1 to 6. Thus, in order to read channel 5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even numbered variables are used to store data from the input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p60-62	Constants to convert from reference junction voltage to °C.
p50-53	Constants to convert from reference junction °C to correction voltage of type B thermocouples.
p40-44	Constants to convert from type B thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type B thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form:
p50 + X (p51 + X (p52 + X p53)) = μV
where, p50 = a_0, p51 = a_1, p52 = a_2, p53 = a_3.

The conversion equation from volts from the thermocouple junction to °C is of the form:
p40 + X (p41 + X (p42 + X (p43 + p44))) = °C
where, p40 = a_0, p41 = a_1, p42 = a_2, p43 = a_3, p44 = a_4.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS – 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
400 to 1100°C	–.09 to 1.0°C

For other ranges, change the p40-p44 constants to the values given in Table 3-7 for the applicable temperature range.

File 40: B Thermocouple (BT)

Table 3-7. Type B Thermocouples — Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0, a_1E, + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (°C)	a_0	a_1	a_2	a_3	a_4	Error Range (°C)
	Argument Exp.	Argument Exp.	Argument Exp.	Argument Exp.	Argument Exp.	Exact-Approx.
0 - 900	. . . . .	8.9244743 -1	-5.7447033 -4	1.8053618 -7	-1.9719121 -11	-30 to 75
0 - 1100	. . . . .	7.2874066 -1	-3.1771931 -4	6.8254996 -8	-5.1002233 -12	-35 to 90
0 - 1400	. . . . .	5.7822214 -1	-1.6039309 -4	2.2187592 -8	-1.0678514 -12	-45 to 110
0 - 1650	. . . . .	4.9929130 -1	-1.0349686 -4	1.0792281 -8	-3.9111456 -13	-50 to 120
0 - 1820	. . . . .	4.6255054 -1	-8.2176262 -5	7.3717195 -9	-2.2913665 -13	-50 to 130
400 - 1100	1.8946288 +2	3.0966136 -1	-5.8100680 -5	8.2483967 -9	-4.7591774 -13	-.09 to 1.0
400 - 1400	2.0949015 +2	2.7222162 -1	-3.6930932 -5	3.6830239 -9	-1.4483702 -13	-3 to 3
400 - 1650	2.2354664 +2	2.4988761 -1	-2.7160312 -5	2.1299660 -9	-6.4220755 -14	-5 to 5
1050 - 1400	3.2188156 +2	1.8282378 -1	-1.1561743 -5	6.4320083 -10	-1.4544375 -14	-.003 to .003
1050 - 1650	3.4418084 +2	1.7031473 -1	-8.9696912 -6	4.0789445 -10	-6.6410259 -15	-.025 to .020
1400 - 1550	3.7140306 +2	1.5828913 -1	-7.0050689 -6	2.6714849 -10	-2.9082072 -15	-.001 to .001
1400 - 1650	3.9253848 +2	1.4979551 -1	-5.7276293 -6	1.8192801 -10	-7.8042686 -16	-.001 to .001

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type B thermocouples (platinum - 30% rhodium alloy vs. platinum - 6% rhodium alloy.

PROGRAM LISTINGS

GPIO

```
0: "B Thermocouple":
1: "BT":ssb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: -.24673839→p51;5.9050303e-3→p52;-1.226718e-6→p53
4: 1.8946288e2→p40;.30966136→p41;-5.810068e-5→p42
5: 8.2483967e-9→p43;-4.7591774e-13→p44
6: 1→p39
7: wtb "MP",170160+dtopp0
8: if (10frc(pp39)→p38)<7;sto +2
9: beep;prt "Address p";p39;"out of range";stp ;sto -1
10: wtb "MP",10000dtoint(pp39)+7,170260+dtopp0
11: wti 4,10000dtoint(pp39);rdi 4→p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wtb "MP",170160+dtopp0,10000dtoint(pp39)+p38,170260+dtopp0
16: wti 4,10000dtoint(pp39);rdi 4→p36
17: ((otdp36→p36)-(p36)2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;sto -13
21: wtb "MP",170040;mdec
22: ret
*31134
```

Bytes used: 830

File 40: B Thermocouple (BT)

HP-IB

```
0: "B Thermocouple":
1: "BT":gsb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: -.24673839→p51;5.9050303e-3→p52;-1.226718e-6→p53
4: 1.8946288e2→p40;.30966136→p41;-5.810068e-5→p42
5: 8.2483967e-9→p43;-4.7591774e-13→p44
6: 1→p39
7: wrt "MP.2","O",160+dtopp0,"T"
8: if (10frc(pp39)→p38)<7;gto +2
9: beep;prt "Address ",p39,"out of range";stp ;gto -1
10: wrt "MP.2",char(64+int(pp39)),7,"T","O",260+dtopp0,"T"
11: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wrt "MP.2","O",160+dtopp0,"T",char(int(pp39)+64),p38,"T"
16: wrt "MP.2","O",260+dtopp0,"T",char(64+int(pp39)),"X";red "MP",p36
17: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;gto -13
21: wrt "MP.1","O40T"
22: ret
*24055
```

Bytes used: 882

File 41: K Thermocouple (KT)

Reads the temperature in °C from 69423A cards and type K thermocouples (nickel - chromium alloy vs. nickel - aluminum alloy). Each 69423A card has six input channels (see program examples.)

SYNTAX

cll 'K Thermocouple' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

or

cll 'KT' (S_0, D_0, S_1, D_1, - - - - S_n, D_n, [U])

PARAMETERS

Parameter List	Variable	Description	Default Value
S_n	p1, p3, etc.	Card Slot and Channel Address	None
D_n	p2, p4, etc.	Storage Locations for Data	None
U	pp0	Unit Number	Unit 0
	p0	Number of Parameters Passed	None

PROGRAM EXAMPLES

cll 'K Thermocouple' (401.2, A, 401.3, G, 402.5, B, 0)

or

cll 'K Thermocouple' (1.2, A, 1.3, G, 2.5, B)

or

cll 'KT' (1.2, A, 1.3, G, 2.5, B)

In the first example, the long descriptive label and the complete slot addresses (401.2, 401.3, etc.) are used, the data storage locations are specified, and the unit number is included. Note that the number to the right of the decimal point specifies the input channel (see variable description). In the next example, shortened slot addresses (1.2, 1.3, etc.) are used, and the optional unit number is omitted. Finally, a short two letter abbreviation is used for the "cll" label, and the optional unit number is omitted. The subprogram will automatically assign a value of zero to pp0 when the unit number is omitted.

VARIABLE DESCRIPTION

p1, p3, etc. — All odd numbered variables are used for slot addresses and channel numbers. The slot address is expressed as a slot number to

	the left of the decimal point and a channel to the right. The slot address values must be from 400 to 414 or from 0 (zero) to 14, where 1 is understood to mean 401, 2 is used for 402, etc. The channel numbers run from 1 to 6. Thus, in order to read channel 5 of the 69423A card in slot 2 use: "402.5" or "2.5". If the slot address or channel numbers are not within range, the program will stop and permit entry of a new value. Slot addresses must be in decimal.
p2, p4, etc.	All even numbered variables are used to store data from the input cards.
pp0	If the unit number is greater than 15, the program will stop and permit entry of a new value. Unit number must be in decimal.
p60-62	Constants to convert from reference junction voltage to °C.
p50-53	Constants to convert from reference junction °C to correction voltage of type K thermocouples.
p40-44	Constants to convert from type K thermocouple voltage to °C.

PROGRAM DESCRIPTION

The subprogram starts with a "gsb" to 'Parameter Check'. The 'Parameter Check' subprogram checks each parameter to ensure that they are within limits. Parameters out of limits will cause the program to stop and wait for a new value.

Upon returning from 'Parameter Check', this subprogram sends a control word to the unit designated by pp0. The unit number, however, is first converted to octal (dto pp0). The subprogram first reads the reference junction voltage and converts this to °C. This value in °C is then converted to a correction voltage for type T thermocouples. The specified channel of the addressed 69423A card is then read and converted to °C.

The conversion equation from °C of reference temperature to voltage correction factor is of the form:
$p50 + X\ (p51 + X\ (p52 + X\ p53)) = \mu V$
where, $p50 = a_0$, $p51 = a_1$, $p52 = a_2$, $p53 = a_3$.

The conversion equation from volts from the thermocouple junction to °C is of the form:
$p40 + X\ (p41 + X\ (p42 + X\ (p43 + p44))) = °C$
where, $p40 = a_0$, $p41 = a_1$, $p42 = a_2$, $p43 = a_3$, $p44 = a_4$.

The constants are taken from the Thermocouple Reference Tables Based on the IPTS — 68, U. S. Department of Commerce (March 1974). The constants chosen for this subprogram are for the following temperature range:

Temperature Range	Error Range
−20 to 500°C	−1.2 to 0.6°C

For other ranges, change the p40-p44 constants to the values given in Table 3-8 for the applicable temperature range.

Table 3-8. Type K Thermocouples — Quartic Approximations to the data as a function of voltage in selected temperature ranges (°C). The expansion is of the form $T = a_0, a_1E, + a_2E^2 + a_3E^3 + a_4E^4$ where E is in microvolts and T is in degrees Celsius.

Temperature Range (°C)	a_0		a_1		a_2		a_3		a_4		Error Range (°C)
	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Argument	Exp.	Exact-Approx.
-270 to 0	. . . . .	. .	1.2329875	-2	-1.4434305	-5	-4.2824995	- 9	-4.2028679	-13	-11 to 8
-200 to 0	. . . . .	. .	2.3783697	-2	-2.4382217	-6	-6.8203073	-10	-9.4854031	-14	-.5 to .5
-200 to 800	. . . . .	. .	2.8346886	-2	-5.8008526	-7	2.5720615	-11	-3.6813679	-16	-8 to 10
- 20 to 500	. . . . .	. .	2.4363851	-2	5.6206931	-8	-3.8825620	-12	3.9120208	-17	-1.2 to .6
0 to 400	. . . . .	. .	2.4383248	-2	9.7830251	-9	3.6276965	-12	-2.5756438	-16	-.5 to .6
0 to 1370	. . . . .	. .	2.5132785	-2	-6.0883423	-8	5.5358209	-13	9.3720918	-18	-2.4 to 1.2
400 to 1000	-2.4707112	+1	2.9465633	-2	-3.1332620	-7	6.5075717	-12	-3.9663834	-17	-.02 to .02
400 to 1370	6.2300671	+0	2.4955374	-2	-7.8788333	-8	1.3269743	-12	1.5580541	-18	-.3 to .3
600 to 800	-3.9480992	+1	3.1425797	-2	-4.0905633	-7	8.5482602	-12	-5.5696636	-17	-.001 to .001
850 to 1000	-3.1617495	+0	2.7115517	-2	-2.1941995	-7	4.8782826	-12	-2.9316611	-17	-.0012 to .0012
1050 to 1150	2.3615582	+2	1.1066277	-3	8.2516607	-7	-1.3558849	-11	9.1638500	-17	-.001 to .001

SPECIAL CONSIDERATIONS

This subprogram returns temperature in °C. To convert to °F, use the following equation:

$$9/5 \times C + 32 = F$$

USED WITH

69423A card and type K thermocouples (nickel - chromium alloy vs. nickel - aluminum alloy).

PROGRAM LISTINGS

GPIO

```
0: "K Thermocouple":
1: "KT":esb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 39.448872→p51;2.4548362e-2→p52;-9.0918433e-6→p53
4: 2.4363851e-2→p41;5.6206931e-8→p42
5: -3.882562e-12→p43;3.9120208e-17→p44
6: 1→p39
7: wtb "MP",170160+dtopp0
8: if (10frc(pp39)→p38)<7;eto +2
9: beep;prt "Address p";p39;"out of range";stp ;eto -1
10: wtb "MP",10000dtoint(pp39)+7,170260+dtopp0
11: wti 4,10000dtoint(pp39);rdi 4→p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wtb "MP",170160+dtopp0,10000dtoint(pp39)+p38,170260+dtopp0
16: wti 4,10000dtoint(pp39);rdi 4→p36
17: ((otdp36→p36)-(p36)2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;eto -13
21: wtb "MP",170040;mdec
22: ret
*9396
```

Bytes used: 814

File 41: K Thermocouple (KT)

HP-IB

```
0: "K Thermocouple":
1: "KT":esb "Parameter Check"
2: 3923.7225→p60;13.158→p61;273.2→p62
3: 39.448872→p51;2.4548362e-2→p52;-9.0918433e-6→p53
4: 2.4363851e-2→p41;5.6206931e-8→p42
5: -3.882562e-12→p43;3.9120208e-17→p44
6: 1→p39
7: wrt "MP.2","O",160+dtopp0,"T"
8: if (10frc(pp39)→p38)<7;gto +2
9: beep;prt "Address ",p39,"out of range";stp ;gto -1
10: wrt "MP.2",char(64+int(pp39)),7,"T","O",260+dtopp0,"T"
11: wrt "MP.1",char(64+int(pp39)),"X";red "MP",p37
12: (otdp37-4096)*10/4352→p37
13: p60/(p61+ln(-p37))-p62→p37
14: p37(p51+p37(p52+p37p53))1e-6→p37
15: wrt "MP.2","O",160+dtopp0,"T";char(int(pp39)+64),p38,"T"
16: wrt "MP.2","O",260+dtopp0,"T";char(64+int(pp39)),"X";red "MP",p36
17: ((otdp36→p36)-(p36>2047)4096)1e-5→p36
18: (p37+p36)1e6→p36
19: p40+p36(p41+p36(p42+p36(p43+p36p44)))→p(p39+1)
20: if (p39+2→p39)<=p0-1;gto -13
21: wrt "MP.1","O40T"
22: ret
*2317
```

Bytes used: 866

APPENDIX A

CHANGING THE INTERFACE SELECT CODES

As shipped from the factory, the interfaces for the 6940B Multiprogrammer have the following select codes:

Interface	Select Code
GPIO (Model 98032A-Opt.040)	9
HP-IB (Model 98034A)	7 (and bus address 23)

If a conflict arises whereby the select codes must be changed, the following procedure should be followed.

First the rotary switch on the interface is changed to the new desired select code. The second step is a software modification. Line 1 in the library subprogram entitled "Parameter Check" (file 2) associates "MP" with the proper select code (and bus address if (HP-IB). As stated above, the normal select codes are 9 for GPIO and 723 for HP-IB. This number must be changed to correspond with the new select code. For example, if a new select code of 6 is desired for the HP-IB, the new line 1 of "Parameter Check" on trk 1 becomes:

"PC": if flg 10#1; dev "MP", 623, "HP-IB", 6;sfg 10

To change the GPIO select code to 8, change line 1 of "Parameter Check" on trk 0 to:

"PC": if flg 10#1; mdec; dev "MP", 8;wti 0,8;sfg 10

The corrected "Parameter Check" should be stored back onto the Software Library Cartridge in file 2 on the proper trk (0 for GPIO, 1 for HP-IB).

APPENDIX B

ADDING CALL SUBPROGRAMS

Programs written using call subprograms not contained in the 14556A Software Library will not run until the call subprograms have been added to the program prepared by the Auto Loader. The subprograms can be added using one of the following methods:

1. Enter the required call subprogram as part of the original user application program.

OR

2. Load the required call subprograms into memory (at the end of the existing program lines) from data cartridge. See example below.

Example: To load a call subprogram from file 50, track 0 to the end of a program already stored in 9825A memory, perform the following steps.

a. Execute the following statement: fetch 9999

b. Press the display arrow, ↑, to view the last line of the program.

c. Note the line number of the last line of program (e.g., 72).

d. Insert the data cartridge containing the call subprogram that you want to load.

e. Execute the following statement: ldf 50, —

Insert a number equal to one more than that noted in step c (e.g., 1 + 72 = 73).

OR

3. Update the 14556A Software Library Directory prior to running the Auto-Loader. The Auto Loader will then automatically load the required call subprogram(s). The directory is part of the Auto Loader program (File No. 1) and contains the file number and call labels of all subprograms contained in the library. Directory updates can be accomplished quite easily. Basically, the string variable C$, that contains the directory is updated and then, if necessary, the variable S in line 0 is updated. To update the directory, proceed as follows:

a. Insert 14556A cartridge.

1. Select track 0 for GPIO or track 1 for HP-IB interface.
2. Load the Auto Loader from file 1 of the library.
3. Execute the following statement: fetch 9999

4. Press the display arrow, ↑, to view the last line of the program.

b. Update String Variable C$.

1. Delete last statement: gto 1.
2. Concatenate the update information with the existing string C$. Each entry must start with a two digit number indicating file number, and end with the label for the call subprogram.
3. Add the following statement: gto 1.

c. Update Variable S.

1. Note the line number of your last line of Auto Loader.
2. Fetch line 0.
3. Change variable S to a number that is one greater than the last line number of the Auto Loader.
4. Change the print statement of line 35 to: del 0, ——— *

last line number of Auto Loader

d. Record the updated Auto Loader in file 1 by executing: rcf 1

e. Record new call subprograms on appropriate track and file.