

# ***JE665***

## ***RS-232C INTERFACE***

***USER'S MANUAL***  
Operation and Programming



This manual is intended as a supplement to the JE664 EPROM Programmer Operation Manual. It describes the operation and programming of the JE665 RS-232C Interface Option. Complete details are given for connection of the JE665 to any computer with a spare RS-232C port (9600 baud, 8 data bits, 2 stop bits, odd parity). A sample program is provided in Microsoft\* MBASIC for implementation on CP/M† systems.

**JAMECO ELECTRONICS** makes no warranties, expressed or implied, concerning the accuracy of this manual. Every effort is made to insure that the contents are correct but errors may occur.

Second Edition

© 1983 by Jameco Electronics  
1355 Shoreway Road  
Belmont, CA 94002  
(415) 592-8097

Written by David H. Nelson — Jameco Electronics

\* Microsoft and MBASIC are registered trademarks of Microsoft Corp.

† CP/M is a registered trademark of Digital Research, Inc.

# ***Table of Contents***

<b>INTRODUCTION</b>	<b>ii</b>
<b>1 THE RS-232C INTERFACE</b>	<b>1</b>
<b>2 THEORY OF OPERATION</b>	<b>2</b>
<b>3 JE665 PROGRAM</b>	<b>4</b>
<b>4 JE665 PROGRAM MODIFICATION</b>	<b>10</b>
<b>5 USING THE JE665 PROGRAM</b>	<b>13</b>

**APPENDIX A — JE665 SCHEMATIC DIAGRAM**

**APPENDIX B — JE665 ASSEMBLY DRAWING AND COMPONENT MATERIAL LIST**

# ***Introduction***

The JE665 option is a circuit board that implements the RS-232C standard interface to allow connection of the JE664 EPROM Programmer to a computer system. The JE665 requires a spare RS-232C port capable of 9600 baud, 8 data bits, 2 stop bits and odd parity.

A sample program is provided in Microsoft MBASIC for implementation on CP/M computers. Modifications may be necessary due to the different hardware configurations available. These changes are explained in detail.

When connection is complete, the computer will have access to the RAM inside the JE664. This allows for convenient data storage and manipulation with the computer.

# The RS-232C Interface

The RS-232C interface is an EIA (Electronic Industries Association) "Recommended Standard" interface. It transmits and receives data in serial form, one bit at a time.

There are no physical requirements for the RS-232C except that it use a 25-pin connector. Accepted practice is to use a 25-pin "D" connector (DB25). The connector on the JE665 is of the female gender and requires a male mate on the cable. Only six (6) of the 25 lines are used by the JE665. Table 1 lists these lines and indicates the direction of signal travel. They will be described in more detail later.

**Table 1. Name and Origins of Signals Used in the JE665**

PIN	NAME	DESCRIPTION	SIGNAL ORIGIN
20	DTR	Data Terminal Ready	Computer
7	GND	Signal Ground	---
6	DSR	Data Set Ready	JE665
5	CTS	Clear to Send	JE665
3	TXD	Transmit Data	JE665
2	RXD	Receive Data	Computer

RS-232C voltage levels are different than standard logic levels. A voltage between +3 and +15V indicates a "High" (logic "0" on the JE665) and a voltage between -3 and -15V indicates a "Low" (logic "1" on the JE665). Notice that these signals are "inverted" (i.e. negative voltage equals logic 1) on the JE665. This is normal but may not be the case on some computers.

A cable is required to connect the JE664/JE665 to your computer. It is recommended that you use a cable with removable pins so that lines may be swapped if necessary. The cable must have at least the six indicated lines, but may have all 25.

In order to transmit data over the RS-232C link, there must be some way to indicate to each device that a data transfer will occur. These protocols help assure that both devices are ready and that the data you wish to transfer does, in fact, get transferred.

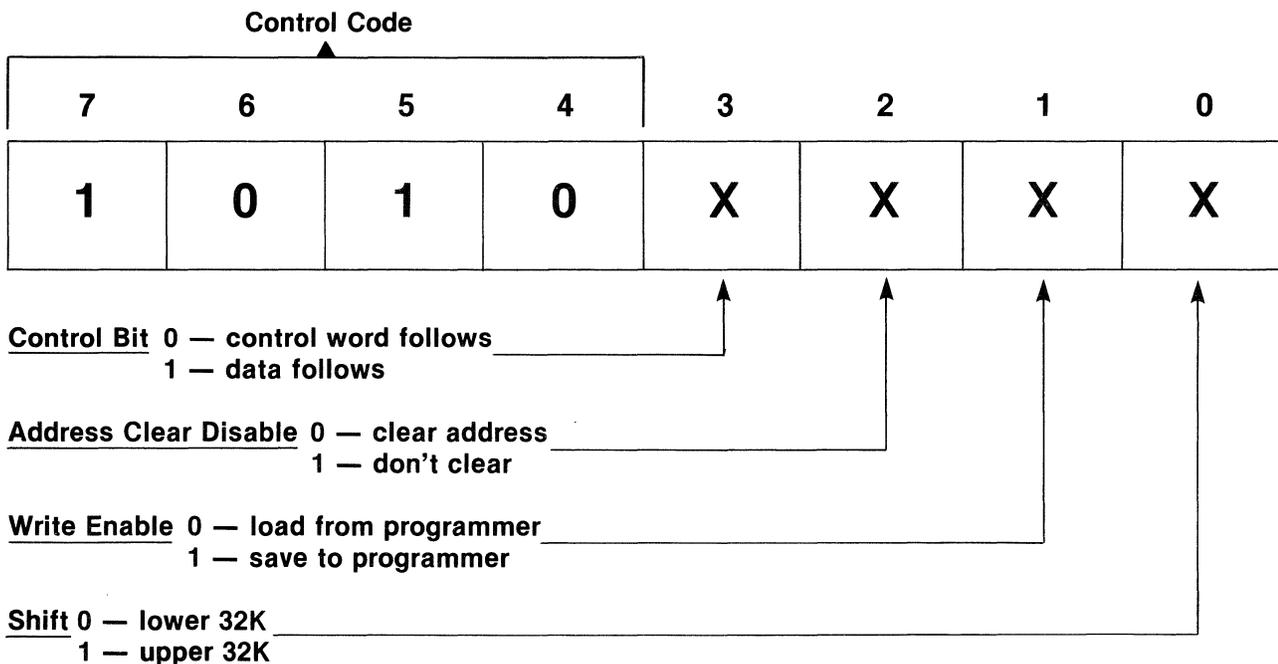
Data accuracy is another important aspect of the RS-232C link. Each byte is checked for proper transfer to and from the JE665. Hardware parity checking is implemented.

# Theory of Operation

To access the JE665 from your computer, certain events must occur in the correct order. As long as the proper signals are present the JE665 will communicate properly. But if they are not you may have unexpected difficulties.

The first signal to occur comes from the JE665. This is the DSR (Data Set Ready, pin 6) line. It will be held high (RS-232C +3 to +15V) by the JE665 when the cable is connected and the PRGM PULSES/RS 232 switch is in the RS 232 position. It indicates that the programmer is ready.

After the computer determines that the programmer is ready it will set the DTR (Data Terminal Ready, pin 20) line low (-3 to -15V) and then back high (+3 to +15V). This will reset the JE665 and tell it to expect a control word next. It will also set the CTS (Clear to Send, pin 5) line low (-3 to -15V) signalling that the JE665 expects a control word.



*Figure 1 — Control Word Detail*

The JE665 uses control words to tell it what to do and what to expect next. The control words are sent in the exact same way that the other data is sent, but there is one difference. The JE665 expects the upper four bits of the control word to be 1010 (A Hex). The lower four bits are the actual commands to the programmer. These commands can select the upper or lower 32K of the programmers RAM, clear the address, select the read or write option, and tell whether or not another control word follows (Figure 1 and Table 2). It should be noted that more than one control word can be sent in succession. The only requirement is that the last control word must have the control bit (bit 3) set. At this point no further control words are accepted until 256 data bytes have been transferred.

After the correct control word(s) are sent, the last one having the control bit (bit 3) set, the actual data will follow. A block of 256 bytes will be transferred. If the transfer is **to** the programmer then each byte will be sent and immediately read back. This will verify that the transfer was correct. If the transfer is **from** the programmer then a dummy byte will be sent and the byte returned will be the desired data.

After the 256 byte block has been transferred, the CTS line will again be set low (-3 to -15V) acknowledging that the block was accepted and indicating that the JE665 expects another control word.

The proper control word(s) are sent followed by another 256 byte data block. This cycle is repeated until all of the data is transferred or until the end of the lower 32K bits is reached. If the lower 32K is filled then two new control words are sent that select the upper 32K. The cycle of 256 byte blocks and control words is then repeated until all of the data has been transferred. (Note: The UPPER 32K LED on the JE664 front panel will NOT change when the RS232 is active even though the 32K segment may change. When returning to manual control of the programmer be sure that you are in the upper or lower 32K as necessary).

**Table 2 — Control Word Definitions**

<b>Control word Hex (Binary)</b>	<b>Will Follow</b>	<b>Clear Address</b>	<b>Read Write</b>	<b>U/L 32K</b>
A0(1010 0000)	control word	yes	---	lower
A1(1010 0001)	control word	yes	---	upper
A4(1010 0100)	control word	no	---	lower
A5(1010 0101)	control word	no	---	upper
A8(1010 1000)	data	yes	read	lower
A9(1010 1001)	data	yes	read	upper
AA(1010 1010)	data	yes	write	lower
AB(1010 1011)	data	yes	write	upper
AC(1010 1100)	data	no	read	lower
AD(1010 1101)	data	no	read	upper
AE(1010 1110)	data	no	write	lower
AF(1010 1111)	data	no	write	upper

# JE665 Program

The sample program of this chapter is provided to help you make the connection from the JE665 to your computer. It has all the features necessary for data transfer, data storage and data entry. You may choose to use it or you may write your own.

The program was written in Microsoft MBASIC version 4.51. If you have MBASIC you can enter the program into your computer and save it as "JE665." The next chapter explains how to customize it to your specific computer.

If you choose to write your own program, carefully follow the sample program and **THEORY OF OPERATION**. You may use another BASIC or a different language. You will also need to read **JE665 PROGRAM MODIFICATION**.

The following is a breakdown of the lines in the JE665 program. Beginning and ending line numbers are shown as well as a description of the function of each routine.

Line Numbers	Description
100 280	Initialization, credits and start up message
300 470	Select EPROM size to work with
500 700	Main program menu
1000 1230	Get data from disk
2000 2240	Put data on disk
3000 3370	Load data from programmer
4000 4390	Save data to programmer
5000 5340	View data (ASCII and Hex)
6000 6240	Enter data
7000 7060	Exit from the program
8000 8040	Clear the screen
8100 8260	Get a file name
8300 8350	Handle any errors when opening files
8400 8470	Convert hex to decimal (SA\$ to SA)
8500 8520	Handle "NO EPROM FILES"
9000 9070	Set up the RS-232C port 9600 baud, 8 data bits, 2 stop bits, odd parity
9100 9130	Check the DSR line (bit) If DSR is on return F = 1 otherwise F = 0
9200 9230	Toggle the DTR line (bit) off then back on
9300 9330	Check the CTS line (bit). If CTS is off then F = 1 otherwise F = 0
9400 9460	Send a data byte from DS and get echo in DR
9500 9530	Send a control word from D

# JE665 Program

```
100 'JE665 RS-232C SERIAL COMMUNICATION PROGRAM
110 'BY DAVID H NELSON
120 'SEPTEMBER 14, 1983
130 GOSUB 8000 'CLEAR THE SCREEN
140 CLEAR 1000
150 PRINT TAB(26);"JAMECO ELECTRONICS":PRINT
160 PRINT TAB(20);"JE665 RS-232C INTERFACE OPTION":PRINT
170 PRINT TAB(21);"SERIAL COMMUNICATION PROGRAM":PRINT:PRINT
180 PRINT TAB(26);"SEPTEMBER 14, 1983"
190 PRINT : PRINT : PRINT : PRINT : PRINT
200 PRINT "INITIALIZING . . ."
210 '***** INITIALIZATION *****
220 F$="GPLSVECX"
230 CS=2:SI$="16Kbit (2K x 8) EPROM active"
240 BA=28672
250 FOR X=BA TO BA+8192
260 POKE X,255
270 NEXT X
280 DEF FNH$(X) = RIGHT$("00"+HEX$(PEEK(BA+X)),2)
300 'SELECT EPROM SIZE
310 GOSUB 8000 'CLEAR SCREEN
320 PRINT TAB(10);SI$:PRINT
330 PRINT TAB(20);"1 - 8Kbit (1K x 8)"
340 PRINT TAB(20);"2 - 16Kbit (2K x 8)"
350 PRINT TAB(20);"3 - 32Kbit (4K x 8)"
360 PRINT TAB(20);"4 - 64Kbit (8K x 8)"
370 PRINT : PRINT : PRINT
380 PRINT " SELECT EPROM SIZE >";
390 A$=INPUT$(1)
400 IF A$=CHR$(13) THEN 500
410 CS=VAL(A$)
420 IF CS<1 OR CS>4 THEN 390
430 IF CS=1 THEN SI$="8Kbit (1K x 8) EPROM active"
440 IF CS=2 THEN SI$="16Kbit (2K x 8) EPROM active"
450 IF CS=3 THEN SI$="32Kbit (4K x 8) EPROM active"
460 IF CS=4 THEN SI$="64Kbit (8K x 8) EPROM active"
470 CS=INT((2^CS)*2+.5)
500 '***** MAIN PROGRAM *****
510 GOSUB 8000 'CLEAR SCREEN
520 PRINT TAB(20);"<< JE665 MAIN MENU >>":PRINT
530 PRINT TAB(20);"G - GET DATA FROM DISK"
540 PRINT TAB(20);"P - PUT DATA ON DISK"
550 PRINT TAB(20);"L - LOAD DATA FROM PROGRAMMER"
560 PRINT TAB(20);"S - SAVE DATA TO PROGRAMMER"
570 PRINT TAB(20);"V - VIEW DATA"
580 PRINT TAB(20);"E - ENTER DATA"
590 PRINT TAB(20);"C - CHANGE EPROM SIZE"
600 PRINT TAB(20);"X - EXIT THIS PROGRAM"
610 PRINT : PRINT : PRINT TAB(10);SI$ : PRINT : PRINT
620 PRINT TAB(10);"ENTER SELECTION >";
630 S$=INPUT$(1)
640 IF S$="C" THEN 300
650 FOR S=1TO8
660 IF S$=MID$(F$,S,1) THEN 690
670 NEXT S
680 GOTO 630
690 ON S GOSUB 1000 , 2000 , 3000 , 4000 , 5000 , 6000 , 7000 , 7000
700 GOTO 500
1000 '***** GET DATA FROM DISK *****
1010 GOSUB 8000 'CLEAR SCREEN
1020 PRINT TAB(10);"GET DATA FROM DISK":PRINT
1030 PRINT
1040 GOSUB 8100 'GET FILENAME
1050 IF FI$="" THEN RETURN
1060 ON ERROR GOTO 8300 'ERROR ROUTINE
1070 OPEN "R",#1,FI$
1080 FIELD#1,128 AS R$
1090 ON ERROR GOTO 0 'NO MORE ERROR HANDLING
```

# JE665 Program (continued)

```
1100 PRINT
1110 PRINT "GETTING DATA . . ."
1120 PRINT "[";TAB(CS*2);"]"
1130 FOR X = 1 TO CS*2
1140     GET#1,X
1150     PRINT "*";
1160     FOR Y%=1 TO 128
1170         POKE BA+(X-1)*128+Y%-1,ASC(MID$(R$,Y%,1))
1180     NEXT Y%
1190 NEXT X
1200 PRINT : PRINT
1210 PRINT "FUNCTION COMPLETE"
1220 CLOSE 1
1230 RETURN
2000 '***** PUT DATA ON DISK *****
2010 GOSUB 8000 'CLEAR SCREEN
2020 PRINT TAB(10);"PUT DATA ON DISK":PRINT
2030 PRINT
2040 GOSUB 8100 'GET FILENAME
2050 IF FI$="" THEN RETURN
2060 ON ERROR GOTO 8300 'ERROR ROUTINE
2070 OPEN "R",#1,FI$
2080 FIELD#1,128 AS R$
2090 ON ERROR GOTO 0
2100 PRINT
2110 PRINT "PUTTING DATA . . ."
2120 PRINT "[";TAB(CS*2);"]"
2130 FOR X=1 TO CS*2
2140     PRINT "*";:P$=""
2150     FOR Y%=1 TO 128
2160         P$=P$+CHR$(PEEK(BA+(X-1)*128+Y%-1))
2170     NEXT Y%
2180     LSET R$=P$
2190     PUT#1,X
2200 NEXT X
2210 PRINT : PRINT
2220 PRINT "FUNCTION COMPLETE"
2230 CLOSE 1
2240 RETURN
3000 '***** LOAD DATA FROM PROGRAMMER *****
3010 GOSUB 8000 'CLEAR SCREEN
3020 PRINT TAB(10);"LOAD DATA FROM PROGRAMMER":PRINT
3030 PRINT
3040 GOSUB 9000 'SET UP PORT
3050 GOSUB 9100 'CHECK DSR LINE
3060 IF F=1 THEN 3100
3070 PRINT "PRESS <RETURN> WHEN PROGRAMMER IS READY"
3080 A$=INPUT$(1):IF A$<>CHR$(13) THEN RETURN
3090 GOTO 3050
3100 GOSUB 9200 'TOGGLE DTR LINE
3110 D=161:GOSUB 9500 'SEND CONTROL WORD A1
3120 NB=CS-1 'NUMBER OF BLOCKS - 1
3130 N1=NB:IF NB=31 THEN N1=15
3140 PRINT : PRINT "LOADING DATA . . ."
3150 FOR X=0 TO N1
3160     D=168:GOSUB 9500 'SEND CONTROL WORD A8
3170     FOR Y=0 TO 255
3180         DS=255:GOSUB 9400:POKE BA+X*256+Y,DR 'GET A BYTE
3190     NEXT Y
3200     GOSUB 9300 'CHECK CTS LINE
3210     IF F=0 THEN PRINT "NO ACKNOWLEDGE FROM PROGRAMMER":A$=INPUT$(1):RETURN
3220 NEXT X
3230 IF NB<>31 THEN RETURN
3240 D=160:GOSUB 9500 'SEND CONTROL WORD A0
3250 D=169:GOSUB 9500 'SEND CONTROL WORD A9
3260 FOR X=0 TO 15
3270     FOR Y=0 TO 255
```



## JE665 Program (continued)

```
5170 A1$="" : A2$=""
5180 FOR Y=0 TO 7
5190   A = PEEK(BA+SA+Y)
5200   IF A<32 OR A>127 THEN A1$=A1$+"." ELSE A1$=A1$+CHR$(A)
5210 NEXT Y
5220 FOR Y=8 TO 15
5230   A = PEEK(BA+SA+Y)
5240   IF A<32 OR A>127 THEN A2$=A2$+"." ELSE A2$=A2$+CHR$(A)
5250 NEXT Y
5260 PRINT USING US$;AD$,H1$,H2$,H3$,H4$,A1$,A2$
5270 SA=SA+16
5280 IF SA>=8192 THEN PRINT:PRINT "PRESS ANY KEY FOR MENU";:A$=INPUT$(1):RETURN
5290 NEXT X
5300 PRINT : PRINT : PRINT
5310 PRINT "PRESS <RETURN> TO CONTINUE - 'A' FOR NEW ADDRESS - ANY OTHER FOR MENU";
5320 A$=INPUT$(1):IF A$=CHR$(13) THEN PRINT : GOTO 5080
5330 IF A$="A" THEN 5000
5340 RETURN
6000 '***** ENTER DATA *****
6010 GOSUB 8000 'CLEAR SCREEN
6020 PRINT TAB(10);"ENTER DATA":PRINT
6030 SA$="0"
6040 INPUT "ENTER THE STARTING ADDRESS (0 TO 1FFF) ";SA$
6050 GOSUB 8400 'HEX TO DECIMAL
6060 IF SA>8191 THEN 6030
6070 AD=SA
6080 AD$=RIGHT$("0000"+HEX$(AD),4)
6090 PRINT AD$;";";
6100 I$=""
6110 INPUT I$
6120 IF I$="" THEN 6240
6130 FOR X=1 TO LEN(I$)
6140   IF MID$(I$,X,1)=" " THEN I$=LEFT$(I$,X-1)+RIGHT$(I$,LEN(I$)-X):GOTO 6140
6150 NEXT X
6160 L=LEN(I$):IF L/2<>INT(L/2) THEN 6090
6170 FOR Z=1 TO L-1 STEP 2
6180   SA$=MID$(I$,Z,2)
6190   GOSUB 8400 'HEX TO DECIMAL
6200   IF SA>255 THEN 6080
6210   POKE BASE+AD,SA : AD=AD+1 : IF AD>8191 THEN 6030
6220 NEXT Z
6230 GOTO 6080
6240 RETURN
7000 'EXIT THIS PROGRAM
7010 PRINT : PRINT : PRINT
7020 PRINT "ARE YOU READY TO EXIT (Y/N)?";
7030 IF INPUT$(1)<>"Y" THEN RETURN
7040 GOSUB 8000 'CLEAR SCREEN
7050 PRINT TAB(55);"THANK YOU"
7060 GOTO 9999
8000 'CLEAR THE SCREEN
8010 FOR ZZ=1 TO 25
8020   PRINT
8030 NEXT ZZ
8040 RETURN
8100 'GET A FILENAME
8110 ON ERROR GOTO 8500
8120 PRINT : PRINT "DIRECTORY:" : FILES "*.ROM" : PRINT : PRINT
8130 FI$=""
8140 INPUT "ENTER THE NAME OF THE FILE ";FI$
8150 LF = LEN(FI$)
8160 CL = INSTR(FI$,":")
8170 IF CL<>0 AND CL<>2 THEN 8140
8180 IF CL=0 AND LF>8 THEN 8140
8190 IF LF>10 THEN 8140
```

## JE665 Program (continued)

```
8200 PD = INSTR(FI$, ".")
8210 IF PD>0 THEN 8140
8220 IF LF<1 THEN FI$="" : RETURN
8230 FI$ = FI$+".ROM"
8240 PRINT
8250 ON ERROR GOTO 0
8260 RETURN
8300 'ERROR HANDLING
8310 IF ERR=53 THEN PRINT "**** FILE NOT FOUND ****":CLOSE 1:RESUME 1030
8320 IF ERR=61 THEN PRINT "**** DISK FULL ****":CLOSE 1:RESUME 2030
8330 IF ERR=67 THEN PRINT "**** DIRECTORY FULL ****":CLOSE 1:RESUME 2030
8340 PRINT "ERROR NUMBER ";ERR;" AT LINE ";ERL :STOP
8350 RETURN
8400 'HEX TO DECIMAL (SA$ TO SA)
8410 SA=0
8420 FOR X=1 TO LEN(SA$)
8430     FOR Y=0 TO 15
8440         IF MID$(SA$,X,1)=HEX$(Y) THEN SA=SA*16+Y
8450     NEXT Y
8460 NEXT X
8470 RETURN
8500 'HANDLE "NO FILES"
8510 IF ERR=53 THEN PRINT "NO EPROM FILES"
8520 RESUME NEXT

9000 'SET UP THE RS 232 PORT
9010 P=5*16+8                'SET P TO PORT ADDRESS
9020 OUT P+3,128+8+4+2+1    'GET READY FOR BAUD DIVISOR
9030 OUT P,12:OUT P+1,0    'SEND BAUD DIVISOR
9040 OUT P+3,8+4+2+1      'SET 8 BITS,2 STOP BITS, ODD PARITY
9050 OUT P+5,0            'RESET FLAGS
9060 OUT P+1,0            'DISABLE ALL INTERRUPTS
9070 RETURN
9100 'CHECK DSR LINE (PROGRAMMER READY)
9110 F=0
9120 IF (INP(P+6)AND32)=32 THEN F=1    'DSR BIT IS ON
9130 RETURN
9200 'TOGGLE DTR LINE (SIGNAL "COMPUTER READY")
9210 OUT P+4,INP(P+4)AND254    'TURN DTR BIT OFF
9220 OUT P+4,INP(P+4)OR1      'TURN DTR BIT BACK ON
9230 RETURN
9300 'CHECK CTS LINE (PROGRAMMER ACKNOWLEDGE)
9310 F=0
9320 IF (INP(P+6)AND16)<>16 THEN F=1    'CTS BIT IS OFF
9330 RETURN
9400 'SEND DATA BYTE
9410 'DS - DATA SEND      DR - DATA RECEIVE
9420 IF (INP(P+5)AND32)<>32 THEN 9420    'WAIT FOR TX REG EMPTY
9430 OUT P,DS                    'SEND THE BYTE
9440 IF (INP(P+5)AND 1)<> 1 THEN 9440    'WAIT FOR DATA READY
9450 DR = INP(P)                'GET THE ECHO
9460 RETURN
9500 'SEND CONTROL WORD
9510 'D - CONTROL WORD
9520 OUT P,D                    'SEND IT OUT
9530 RETURN
9999 END
```

# ***JE665 Program Modification***

In order for the JE665 Program to work properly on your computer it will need to be modified. This is necessary due to the different hardware configurations available.

The first step is to locate an unused RS-232C port and determine which chip your computer uses as an interface for that port. You should be able to find this information in the computer's documentation. If not, you may ask your computer dealer or contact the customer support department of your computer manufacturer.

The chip will likely be one of the more popular ones such as:

8250	ACE	Asynchronous Communications Element
8251	USART	Universal Synchronous/Asynchronous Receiver/Transmitter
Z80 -	SIO	Serial Input/Output controller
Z80 -	DART	Dual Asynchronous Receiver/Transmitter
2651	PCI	Programmable Communications Interface

In any case you will need some information on how to program it. The JE665 Program uses the 8250 ACE as an example.

Now you need to know where (what port address) your interface chip is located. This will be important for programming. The base address should be in the range from 0 to 254. If you have a 16 bit computer this number may be higher. There will be two or more addresses for the chip. The 8250 uses six addresses. At least one of these will be a transmit and/or receive register and at least one other will be a status and/or control register.

In the chip specifications you should locate the registers and particular bits that indicate and control each of the following:

- DTR — Data Terminal Ready (computer ready)
- DSR — Data Set Ready (programmer ready)
- CTS — Clear to Send (programmer acknowledge)

These lines are individually monitored and controlled by the JE665 program.

The program lines that you will need to modify will be in the range from 9000 to 9999. No other changes should be needed.

- Lines 9000 through 9070 set up the RS-232C port for the correct characteristics. They set the baud rate to 9600, the number of data bits to 8, the number of stop bits to 2, and set odd parity. They also reset all of the flags and turn off any interrupt modes the port may have. Line 9010 tells the program that the base address of the port chip is 58H (88 decimal). You should change all of these lines to set up your chip. If you have an 8250 then the only change necessary should be setting P to the correct base address in line 9010. Remember that these modifications need not agree line for line with the sample program but only that each complete routine does its particular task in any way possible.

Some computers may require that you set up the port from hardware rather than software. In this case it may be necessary to move some jumpers or change a DIP switch or two. Instructions for this will be found in your hardware manuals.

- The DSR (Data Set Ready) line is checked in lines 9100 to 9130. If the DSR bit is in the proper state (1) then F is set to 1, otherwise it is set to 0. Line 9120 does all the work here. It checks the DSR bit (bit 5) of the status register on the 8250 chip. To check any bit (0-7) of a register you can use the general statement

```
IF (INP(REGADDR) AND (2 ^ BIT)) = (2 ^ BIT) THEN . . .
```

This will be true if BIT is a 1. If you want to check that a bit is 0, replace the “=” with a “< >.”

- Lines 9200 to 9230 toggle the DTR (Data Terminal Ready) line to 0 and then back to 1. They do this by turning the DTR bit of the control register off and then back on. To turn any particular bit off use the statement

```
OUT REGADDR,INP(REGADDR) AND (256-2 ^ BIT)
```

and to turn any bit on use

```
OUT REGADDR,INP(REGADDR) OR (2 ^ BIT)
```

Change lines 9210 and 9220 to reflect your register locations and bit numbers.

- The CTS (Clear to Send) bit is checked in line 9320 of the CHECK CTS LINE routine. If it is correct (off) then F is set to 1, otherwise it is set to 0. This routine starts at line 9300 and ends at line 9330. Change line 9320 to reflect your chip address and bit location.

- The routine that actually sends the data over the RS-232C link is from 9400 to 9460. DS is the data to be sent. Line 9420 waits for the transmitter (TX) buffer to be empty and then line 9430 sends DS to the transmitter buffer which takes care of the rest of the transmission. Line 9340 similarly waits for the receiver (RX) buffer to be full and then line 9350 reads the echoed data into DR. Once again, these lines should be changed to reflect the proper addresses and bits for your chip.
- The final routine that needs to be modified is very similar to the previous routine. D is the control word to be sent. It is output to the transmitter register in line 9350.

Congratulations! You have finished the modifications to the JE665 Program. You can now save it as "JE665." You can then proceed to ***USING THE JE665 PROGRAM.***

# Using the JE665 Program

Now you are ready to begin using the JE665 and the JE665 Program. Your first step is to be sure that the JE664 Programmer is set up correctly. You should connect your RS-232C cable between the programmer and the port on your computer.

There are two distinct members in an RS-232C link: the DTE (Data Terminal Equipment) and the DCE (Data Communication Equipment). They are distinguished by the locations of the signals on the connector. Some computers act as DTE and some act as DCE. Others are hardware selectable. The JE665 is DTE.

You must be sure that all of the signals from the computer go to the proper lines on the JE665. Check the hardware manual for your port and be sure that the DTR signal goes to pin 20 of the JE665, CTS goes to pin 5, etc. You can change signals by pushing the pins out and swapping them on one end of the RS-232C cable. Some common signal swaps are the transmit and receive lines (pins 2 and 3) and the DTR and DSR lines (pins 20 and 6).

After the cable is connected with the proper lines, be sure that the programmer is plugged in and turned on. Set the mode switch to **KEYBD** and the **PRGM PULSES/RS 232** switch to **RS-232**. Also be sure the **WRITE ENABLE** switch is off. Now invoke the program by typing

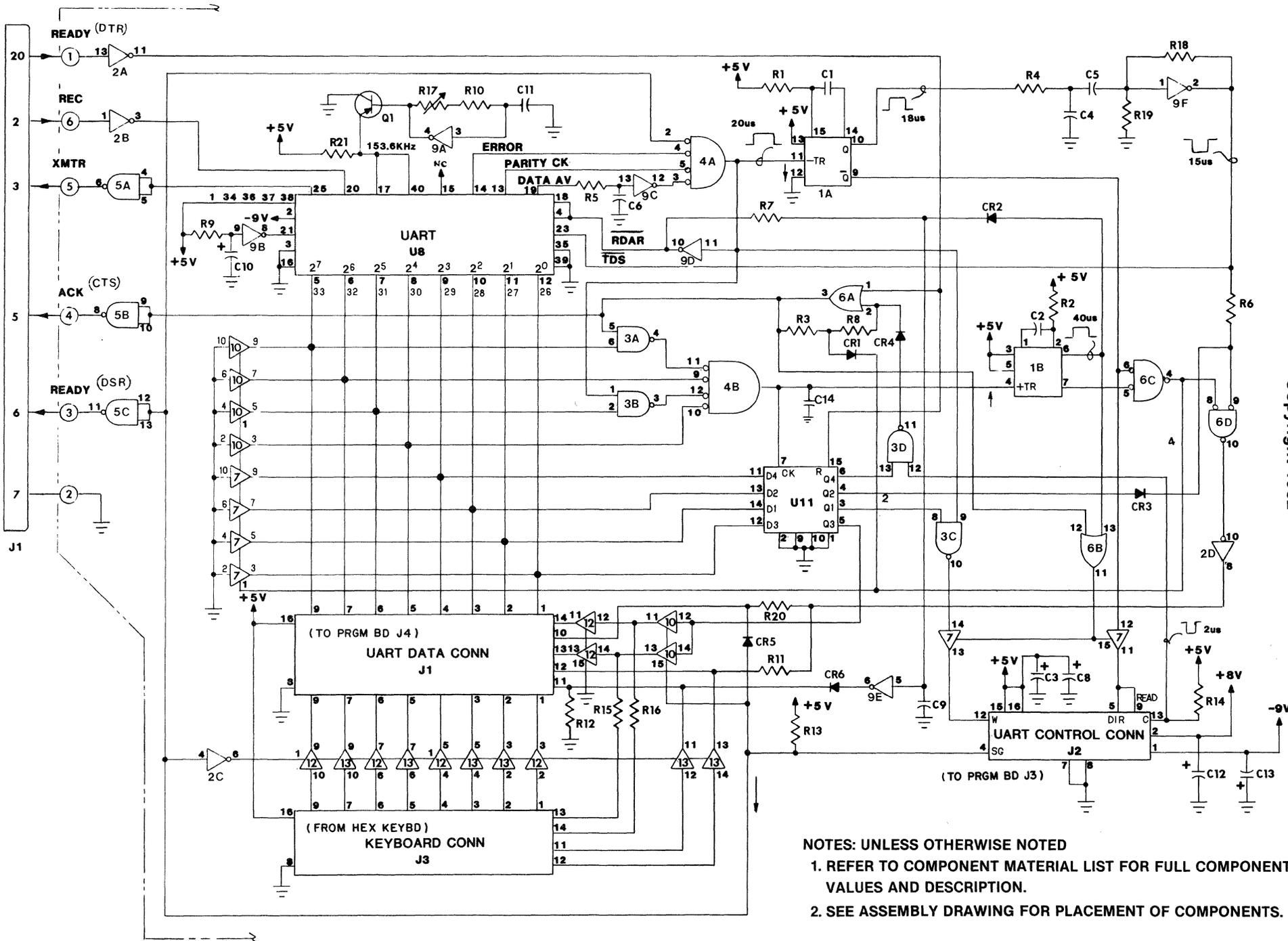
MBASIC JE665

A log-on message will indicate that the program is initializing. Next a menu will appear asking for the size of EPROM you will be working with. This may easily be changed later. Press the number of your choice. Now the main menu will appear. It will allow you to load data from the programmer or from disk and to save data to the programmer or to disk. It will also allow you to view the data presently in computer RAM and to enter new data. These options give you the flexibility to store, retrieve and modify EPROM data.

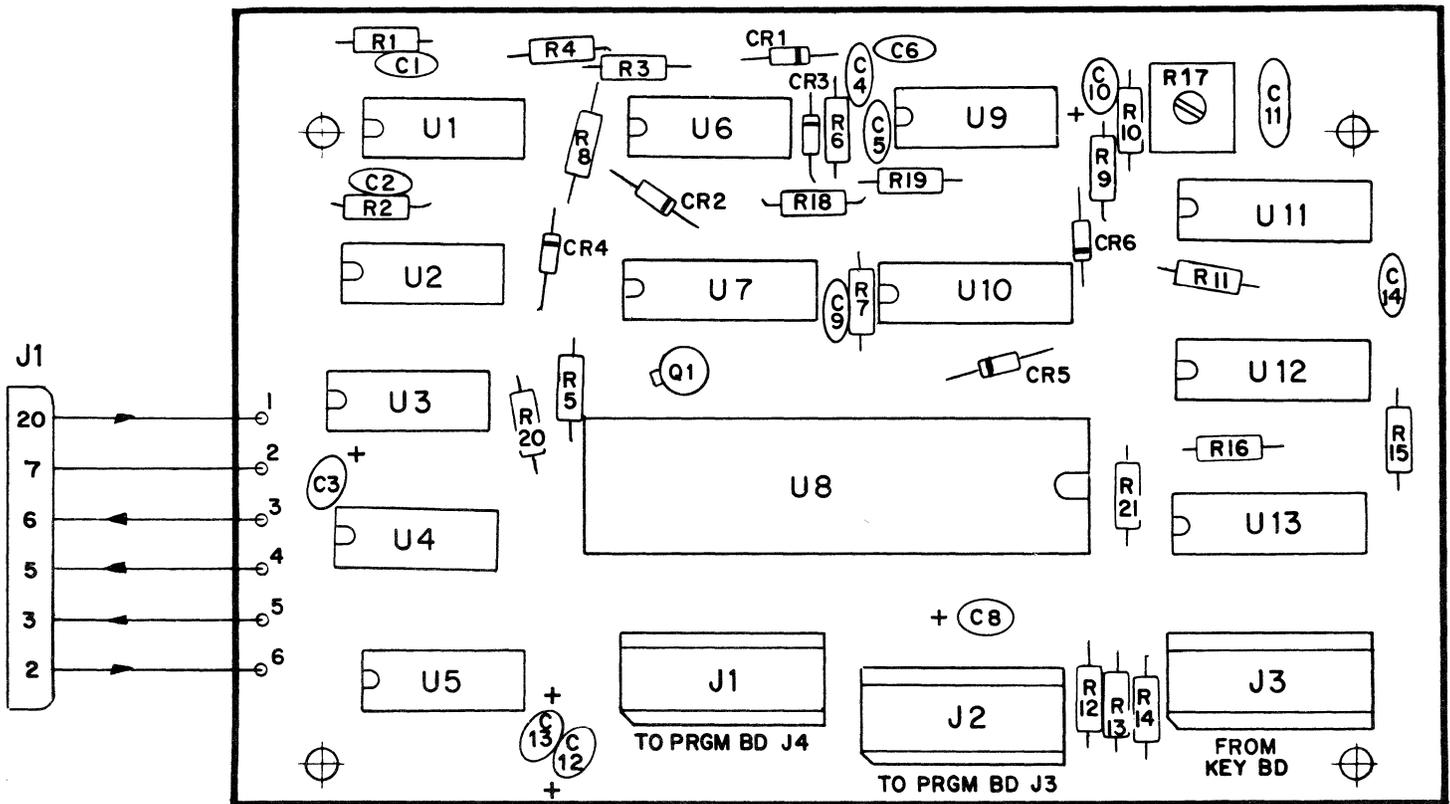
For the Enter Data option, when the address appears enter the data in groups of two hex digits separated by spaces. Enter a blank line to return to the main menu. This same technique will return you from a disk command.

The rest of the program is self explanatory. Simply press the letter of your selection and answer any questions that may be asked. If an error message appears just press any key to return to the main menu. When you are through with the program, select option X and answer Y when asked if you are ready to exit.

You have added a new dimension to using the JE664 EPROM Programmer. We hope you enjoy this new flexibility.



# Appendix B — JE665 Assembly Drawing and Component Material List



SCHEM REF	DESCRIPTION	QTY	SCHEM REF	DESCRIPTION	QTY
U1	IC-CD4098, DUAL MULTIVIBRATOR	1	J1	CONNECTOR, RS-232, 25-PIN, DB25S	1
U2	IC-LM1489, QUAD LINE RECVR	1	R1	RESISTOR, 1/4W, 5%, 68K	1
U3	IC-CD4011, QUAD NAND GATE	1	R2	RESISTOR, 1/4W, 5%, 47K	1
U4	IC-CD4002, DUAL NOR GATE	1	R3,8,14	RESISTOR, 1/4W, 5%, 27K	3
U5	IC-LM1488, QUAD LINE DRIVER	1	R4,7	RESISTOR, 1/4W, %, 4.7K	2
U6	IC-CD4071, DECADE CNT/DIV	1	R5	RESISTOR, 1/4W, 5%, 18K	1
U7,10,12,13	IC-CD4503, TRI HEX BUFFER	4	R6,11,15,16,20,21	RESISTOR, 1/4W, 5%, 10K	6
U8	IC-AY-5-1013A, UART IC	1	R9,12,19	RESISTOR, 1/4W, 5%, 100K	3
U9	IC-74C14, HEX SCH TRIGGER	1	R10	RESISTOR, METAL FILM 10K, 1%	1
U11	IC-CD4076, QUAD REGIST	1	R13	RESISTOR, 1/4W, 5%, 1K	1
C1,4,9	CAPACITOR, DISC, 220PF, 50V	3	R17	RESISTOR, TRIM POT, 1K (840P-1K)	1
C2,C6	CAPACITOR, DISC, .001MF, 50V	2	R18	RESISTOR, 1/4W, %, 200K	1
C3,8,10,12,13	CAPACITOR, TANT, 2.2MF, 35V	5	Q1	TRANSISTOR, 2N2907A	1
C5,C14	CAPACITOR, DISC, 100PF, 50V	2		RS232 PC BOARD JE665-1	1
C11	CAPACITOR, MICA, 240PF, 5%, 500V	1			
CR1-CR6	DIODE, 1N4148	6			

---

*NOTICE — Changes are periodically made to the information herein. These changes will be incorporated in new editions of this manual.*

---

**FOR SERVICE, SEND EQUIPMENT TO:**

**JAMECO ELECTRONICS REPAIR CENTER, 1355 SHOREWAY ROAD, BELMONT, CALIFORNIA 94002  
PLEASE PACK SECURELY. WE SUGGEST YOU INSURE THE SHIPMENT.**