TRS-80® Pocket Computer
Model PC-2
RS-232C Interface

Owner's Manual

TERMS AND CONDITIONS OF SALE AND LICENSE OF RADIO SHACK COMPUTER EQUIPMENT AND SOFTWARE
PURCHASED FROM A RADIO SHACK COMPANY-OWNED COMPUTER CENTER, RETAIL STORE OR FROM A
RADIO SHACK FRANCHISEE OR DEALER AT ITS AUTHORIZED LOCATION

LIMITED WARRANTY

I. CUSTOMER OBLIGATIONS

A   CUSTOMER assumes full responsibility that this Radio Shack computer hardware purchased (the "Equipment"), and any copies of Radio
    Shack software included with the Equipment or licensed separately (the "Software") meets the specifications, capacity, capabilities,
    versatility, and other requirements of CUSTOMER

B   CUSTOMER assumes full responsibility for the condition and effectiveness of the operating environment in which the Equipment and Software
    are to function, and for its installation

II. RADIO SHACK LIMITED WARRANTIES AND CONDITIONS OF SALE

A   For a period of ninety (90) calendar days from the date of the Radio Shack sales document received upon purchase of the Equipment, RADIO
    SHACK warrants to the original CUSTOMER that the Equipment and the medium upon which the Software is stored is free from manufacturing
    defects THIS WARRANTY IS ONLY APPLICABLE TO PURCHASES OF RADIO SHACK EQUIPMENT BY THE ORIGINAL CUSTOMER FROM
    RADIO SHACK COMPANY-OWNED COMPUTER CENTERS, RETAIL STORES AND FROM RADIO SHACK FRANCHISEES AND DEALERS AT ITS
    AUTHORIZED LOCATION The warranty is void if the Equipment's case or cabinet has been opened, or if the Equipment or Software has been
    subjected to improper or abnormal use If a manufacturing defect is discovered during the stated warranty period, the defective Equipment
    must be returned to a Radio Shack Computer Center, a Radio Shack retail store, participating Radio Shack franchisee or Radio Shack dealer
    for repair, along with a copy of the sales document or lease agreement The original CUSTOMER'S sole and exclusive remedy in the event of
    a defect is limited to the correction of the defect by repair, replacement, or refund of the purchase price, at RADIO SHACK'S election and sole
    expense RADIO SHACK has no obligation to replace or repair expendable items

B   RADIO SHACK makes no warranty as to the design, capability, capacity, or suitability for use of the Software, except as provided in this
    paragraph Software is licensed on an "AS IS" basis, without warranty The original CUSTOMER'S exclusive remedy, in the event of a
    Software manufacturing defect, is its repair or replacement within thirty (30) calendar days of the date of the Radio Shack sales document
    received upon license of the Software The defective Software shall be returned to a Radio Shack Computer Center, a Radio Shack retail store,
    participating Radio Shack franchisee or Radio Shack dealer along with the sales document

C   Except as provided herein no employee, agent, franchisee, dealer or other person is authorized to give any warranties of any nature on behalf
    of RADIO SHACK

D   Except as provided herein, **RADIO SHACK MAKES NO WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
    PARTICULAR PURPOSE.**

E   Some states do not allow limitations on how long an implied warranty lasts, so the above limitation(s) may not apply to CUSTOMER

III. LIMITATION OF LIABILITY

A   EXCEPT AS PROVIDED HEREIN, RADIO SHACK SHALL HAVE NO LIABILITY OR RESPONSIBILITY TO CUSTOMER OR ANY OTHER PERSON
    OR ENTITY WITH RESPECT TO ANY LIABILITY, LOSS OR DAMAGE CAUSED OR ALLEGED TO BE CAUSED DIRECTLY OR INDIRECTLY BY
    "EQUIPMENT" OR "SOFTWARE" SOLD, LEASED, LICENSED OR FURNISHED BY RADIO SHACK, INCLUDING, BUT NOT LIMITED TO, ANY
    INTERRUPTION OF SERVICE, LOSS OF BUSINESS OR ANTICIPATORY PROFITS OR CONSEQUENTIAL DAMAGES RESULTING FROM THE
    USE OR OPERATION OF THE "EQUIPMENT" OR "SOFTWARE" IN NO EVENT SHALL RADIO SHACK BE LIABLE FOR LOSS OF PROFITS, OR
    ANY INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY BREACH OF THIS WARRANTY OR IN ANY MANNER
    ARISING OUT OF OR CONNECTED WITH THE SALE, LEASE, LICENSE, USE OR ANTICIPATED USE OF THE "EQUIPMENT" OR "SOFTWARE"

NOTWITHSTANDING THE ABOVE LIMITATIONS AND WARRANTIES, RADIO SHACK'S LIABILITY HEREUNDER FOR DAMAGES INCURRED BY CUSTOMER OR OTHERS SHALL NOT EXCEED THE AMOUNT PAID BY CUSTOMER FOR THE PARTICULAR "EQUIPMENT" OR "SOFTWARE" INVOLVED.

B   RADIO SHACK shall not be liable for any damages caused by delay in delivering or furnishing Equipment and/or Software.

C   No action arising out of any claimed breach of this Warranty or transactions under this Warranty may be brought more than two (2) years after the cause of action has accrued or more than four (4) years after the date of the Radio Shack sales document for the Equipment or Software, whichever first occurs.

D   Some states do not allow the limitation or exclusion of incidental or consequential damages, so the above limitation(s) or exclusion(s) may not apply to CUSTOMER.

IV.   RADIO SHACK SOFTWARE LICENSE

RADIO SHACK grants to CUSTOMER a non-exclusive, paid-up license to use the RADIO SHACK Software on **one** computer, subject to the following provisions:

A   Except as otherwise provided in this Software License, applicable copyright laws shall apply to the Software.

B   Title to the medium on which the Software is recorded (cassette and/or diskette) or stored (ROM) is transferred to CUSTOMER, but not title to the Software.

C   CUSTOMER may use Software on one host computer and access that Software through one or more terminals if the Software permits this function.

D   CUSTOMER shall not use, make, manufacture, or reproduce copies of Software except for use on **one** computer and as is specifically provided in this Software License. Customer is expressly prohibited from disassembling the Software.

E   CUSTOMER is permitted to make additional copies of the Software **only** for backup or archival purposes or if additional copies are required in the operation of **one** computer with the Software, but only to the extent the Software allows a backup copy to be made. However, for TRSDOS Software, CUSTOMER is permitted to make a limited number of additional copies for CUSTOMER'S own use.

F   CUSTOMER may resell or distribute unmodified copies of the Software provided CUSTOMER has purchased one copy of the Software for each one sold or distributed. The provisions of this Software License shall also be applicable to third parties receiving copies of the Software from CUSTOMER.

G   All copyright notices shall be retained on all copies of the Software.

V.   APPLICABILITY OF WARRANTY

A   The terms and conditions of this Warranty are applicable as between RADIO SHACK and CUSTOMER to either a sale of the Equipment and/or Software License to CUSTOMER or to a transaction whereby RADIO SHACK sells or conveys such Equipment to a third party for lease to CUSTOMER.

B   The limitations of liability and Warranty provisions herein shall inure to the benefit of RADIO SHACK, the author, owner and/or licensor of the Software and any manufacturer of the Equipment sold by RADIO SHACK.

VI.   STATE LAW RIGHTS

The warranties granted herein give the **original** CUSTOMER specific legal rights, and the **original** CUSTOMER may have other rights which vary from state to state.

SERVICE POLICY

Radio Shack's nationwide network of service facilities provides quick, convenient, and reliable repair services for all of its computer products, in most instances. Warranty service will be performed in accordance with Radio Shack's Limited Warranty. Non-warranty service will be provided at reasonable parts and labor costs.

Because of the sensitivity of computer equipment, and the problems which can result from improper servicing, the following limitations also apply to the services offered by Radio Shack:

1. If any of the warranty seals on any Radio Shack computer products are broken, Radio Shack reserves the right to refuse to service the equipment or to void any remaining warranty on the equipment.

2. If any Radio Shack computer equipment has been modified so that it is not within manufacturer's specifications, including, but not limited to, the installation of any non-Radio Shack parts, components, or replacement boards, then Radio Shack reserves the right to refuse to service the equipment, void any remaining warranty, remove and replace any non-Radio Shack part found in the equipment, and perform whatever modifications are necessary to return the equipment to original factory manufacturer's specifications.

3. The cost for the labor and parts required to return the Radio Shack computer equipment to original manufacturer's specifications will be charged to the customer in addition to the normal repair charge.

Contents

Introduction

Congratulations on selecting the PC-2 RS-232C Interface. This unit allows your PC-2 (26-3601) to communicate with other computers and computer equipment via an RS-232C channel.

With this Interface, you can connect the PC-2 to and establish data communications with:

- Computers that have RS-232C capabilities
- Modems and Acoustic Couplers
- Printers or plotters equipped with Serial Interfaces

This manual will:

- Describe the RS-232C Interface.
- Show how to connect the RS-232C Interface to your PC-2.
- Provide you with BASIC and Terminal commands that let you do your own programming.

1/ Description of the PC-2 RS-232C Interface

Before doing anything, be sure the following accessories are included in the package:

① RS-232 Interface

② Two Metal Support Brackets

③ Two Long Screws to attach the Metal Support Bracket to the Printer/Plotter/Dual Cassette Interface

④ AC Power Supply Adapter

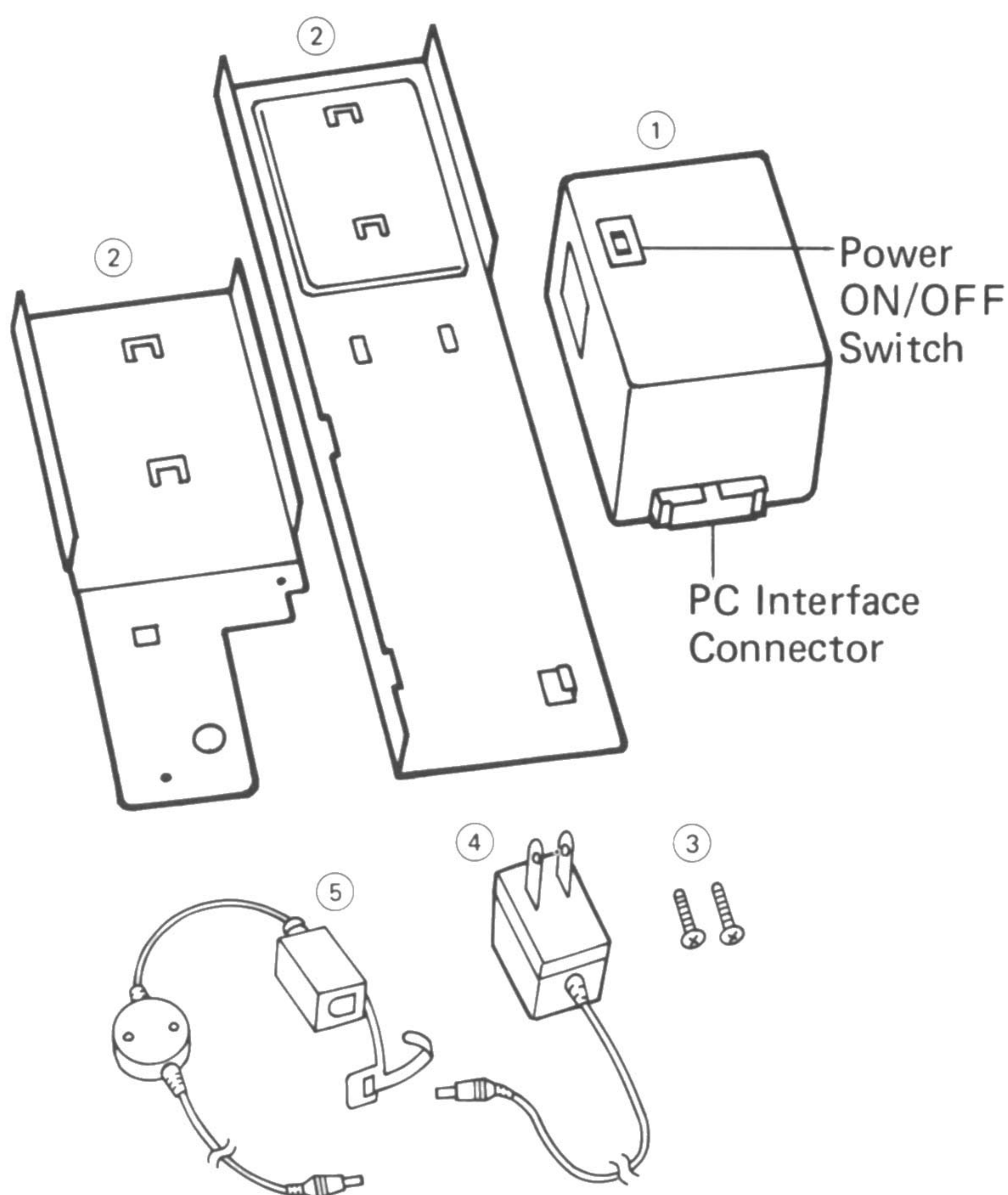⑤ Printer/Plotter/Dual Cassette Interface Choke Cord (For U.S.A. only)

Figure 1. PC-2 RS-232C Interface and Accessories.

(1) Power ON/OFF Switch   To turn the Interface's power on, set this switch to the ON position. Set to OFF to turn the power off.

(2) RS-232C Connector   Connect one end of an appro-
priate RS-232C cable to this connector. Connect the
other end of the cable to the device with which you
wish to communication.

(3) PC Interface Connector   Plug this connector into the
connector on the left side of the PC-2 or on the rear
panel of the PC-2 Printer/Plotter/Dual Cassette Inter-
face.

(4) AC Adapter   To use an external power supply or to
recharge, plug the AC Adapter into the Interface
Power Supply Jack.   (See recharging instructions
before attempting to use the PC-2 RS-232C Inter-
face.)

(5) Choke Cable. (U.S.A. only) Plug this cable into the
Printer/Plotter/Dual   Cassette   Interface's   External
Power Supply Jack.

(6) Metal Support Bracket A.   Protects the connection
between the PC-2 and the PC-2 RS-232C Interface.

(7) Printer/Plotter/Dual Cassette Interface Metal Support
Bracket B.   Protects the connection between the
Printer/Plotter/Dual Cassette Interface and the PC-2
RS-232C Interface.   Requires two screws for attach-
ment (included in the package).

2/ Setting-Up the PC-2 RS-232C Interface

The PC-2 RS-232C Interface can be used with or without a Printer/Plotter/Dual Cassette Interface. For this reason, there are two sets of diagrams which illustrate how to set-up the system.

Be sure to turn all equipment OFF before attempting to connect the PC-2 RS-232C Interface.

Connecting the PC-2 to the RS-232C Interface

If you are using the PC-2 without a Printer/Plotter/Dual Cassette Interface, follow these steps:

1. Slide Metal Support Bracket A into the slots located on the base of the PC-2 (Figure 2).
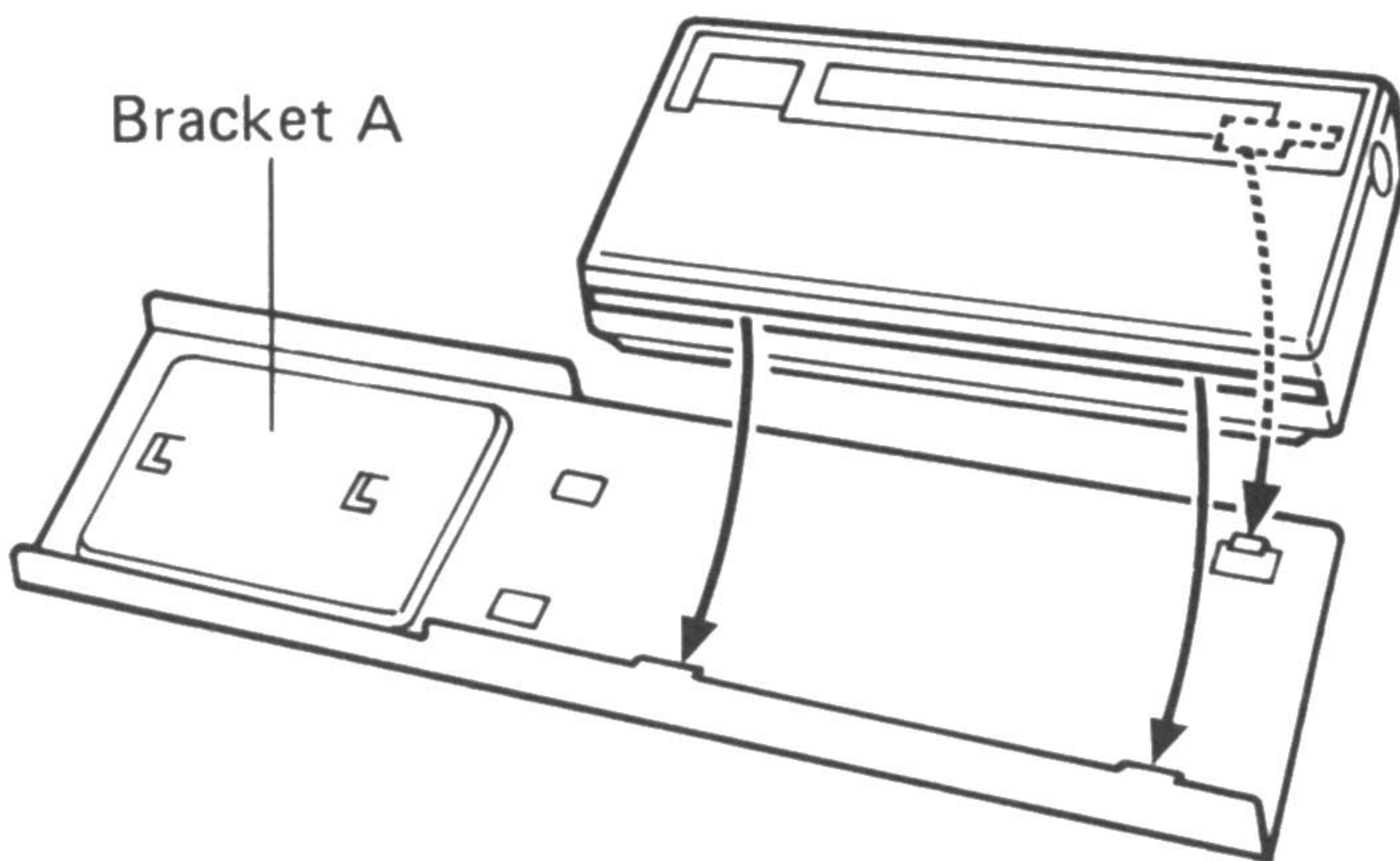

Figure 2. Connecting Metal Support Bracket A to the PC-2.

2. Remove the Interface Covers from both the PC-2 and the RS-232C Interface. Locate the notch on the underside of the PC-2 and slide it into the tab of Metal Support Bracket A.

3. Slide the RS-232C Interface onto Metal Support Bracket A so that the RS-232C 60-Pin connectors are connected. (See Figure 3.)
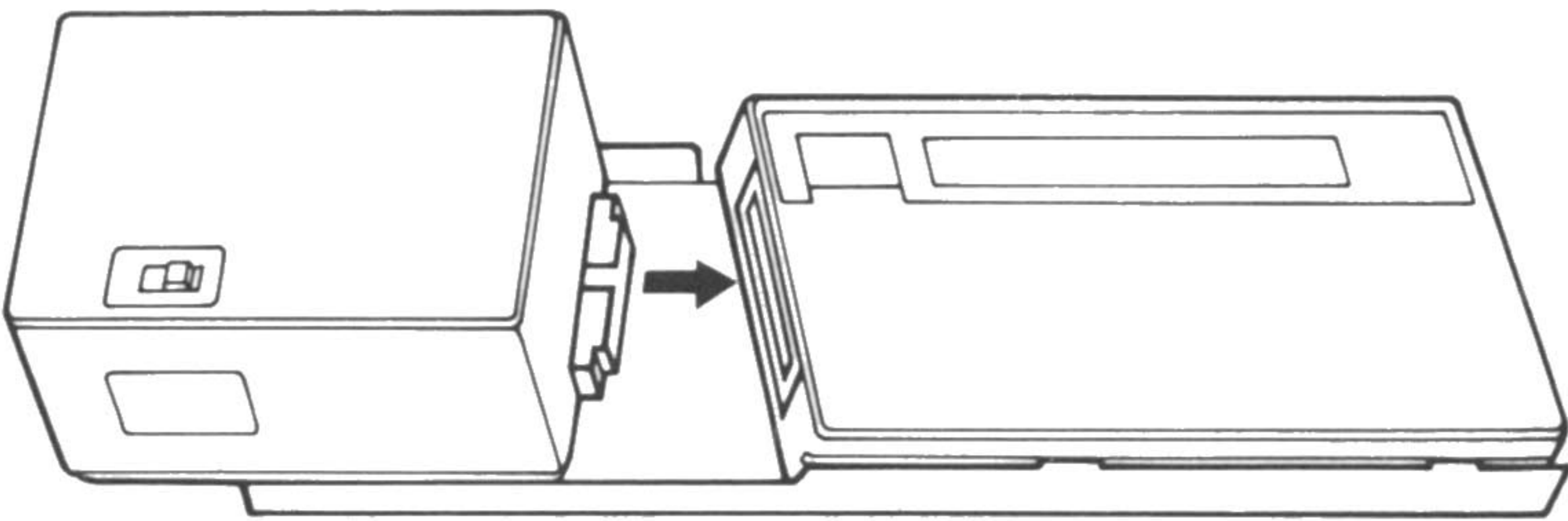

Figure 3. Connecting the PC-2 to the RS-232C Interface.

4. If you are using the AC Adapter, plug it into the PC-2 RS-232C Power Supply Jack.

Connecting the PC-2 Printer/Plotter/Dual Cassette Interface to the PC-2 RS-232C Interface

1. At the bottom of the PC-2 Printer/Plotter/Dual Cassette Interface, remove screws A and B. Position Metal Support Bracket B so that the Interface clips to the slot of the bracket. To secure the metal bracket, use the long screws included with the package. (See Figure 4.)
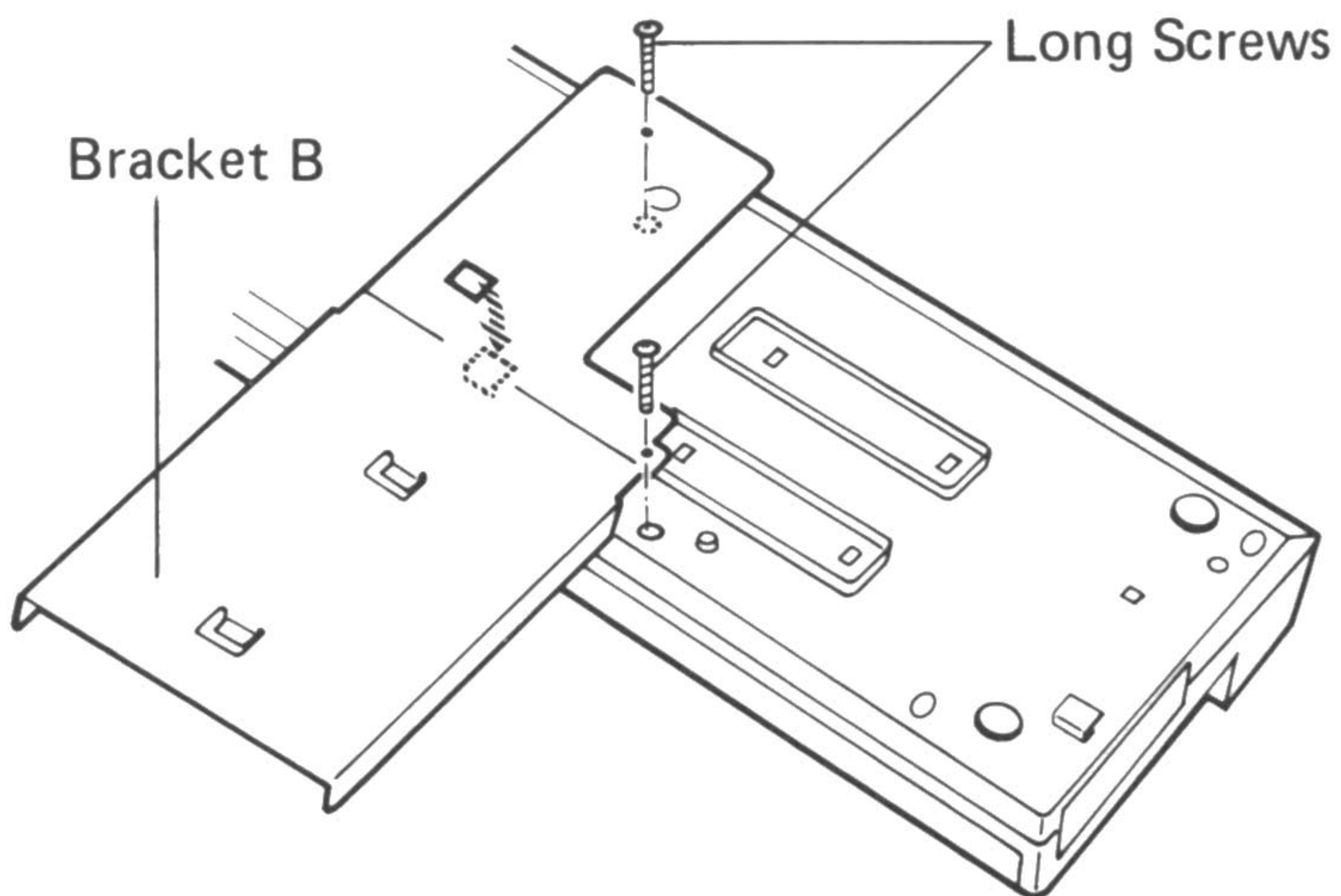
Figure 4. Connecting Metal Support Bracket B to the PC-2 Printer/Plotter/Dual Cassette Interface.

2. Slide the RS-232C Interface onto Metal Support Bracket B so that the Interface's male 60-Pin Plug connects with the Printer/Plotter/Dual Interface's female 60-Pin Plug. (See Figure 5.)
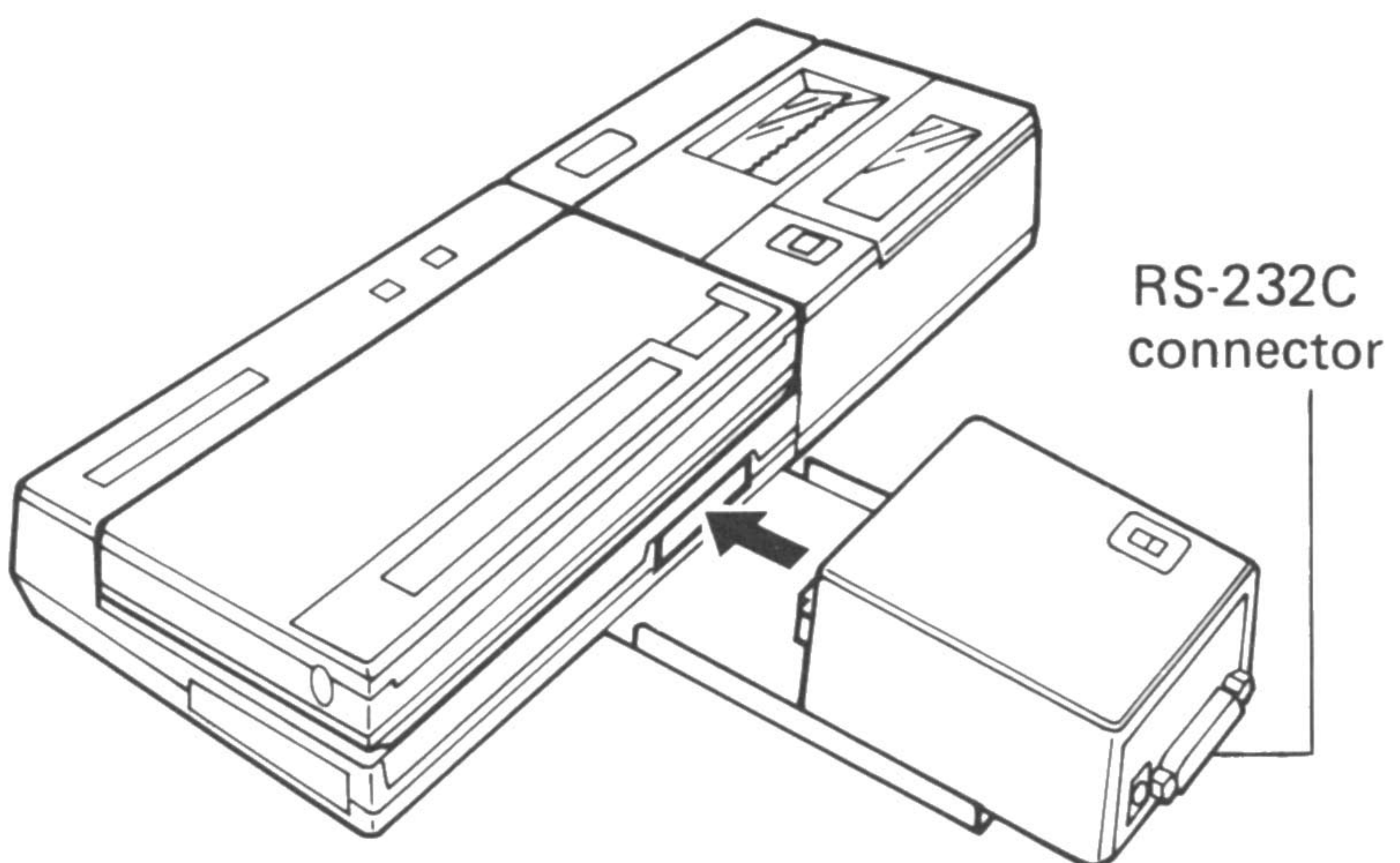

Figure 5. RS-232C Interface to Printer/Plotter/Dual Cassette Interface.

3. Connect one end of the RS-232C Cable to the PC-2's RS-232C Connector and the other end to the device with which you are communicating.

4. (For U.S.A. only) If you are using an external power supply for Printer/Plotter/Dual Cassette Interface, plug the Choke Cable into the external power supply connector of the Printer/Plotter/Dual Cassette Interface. (See Figure 6.)

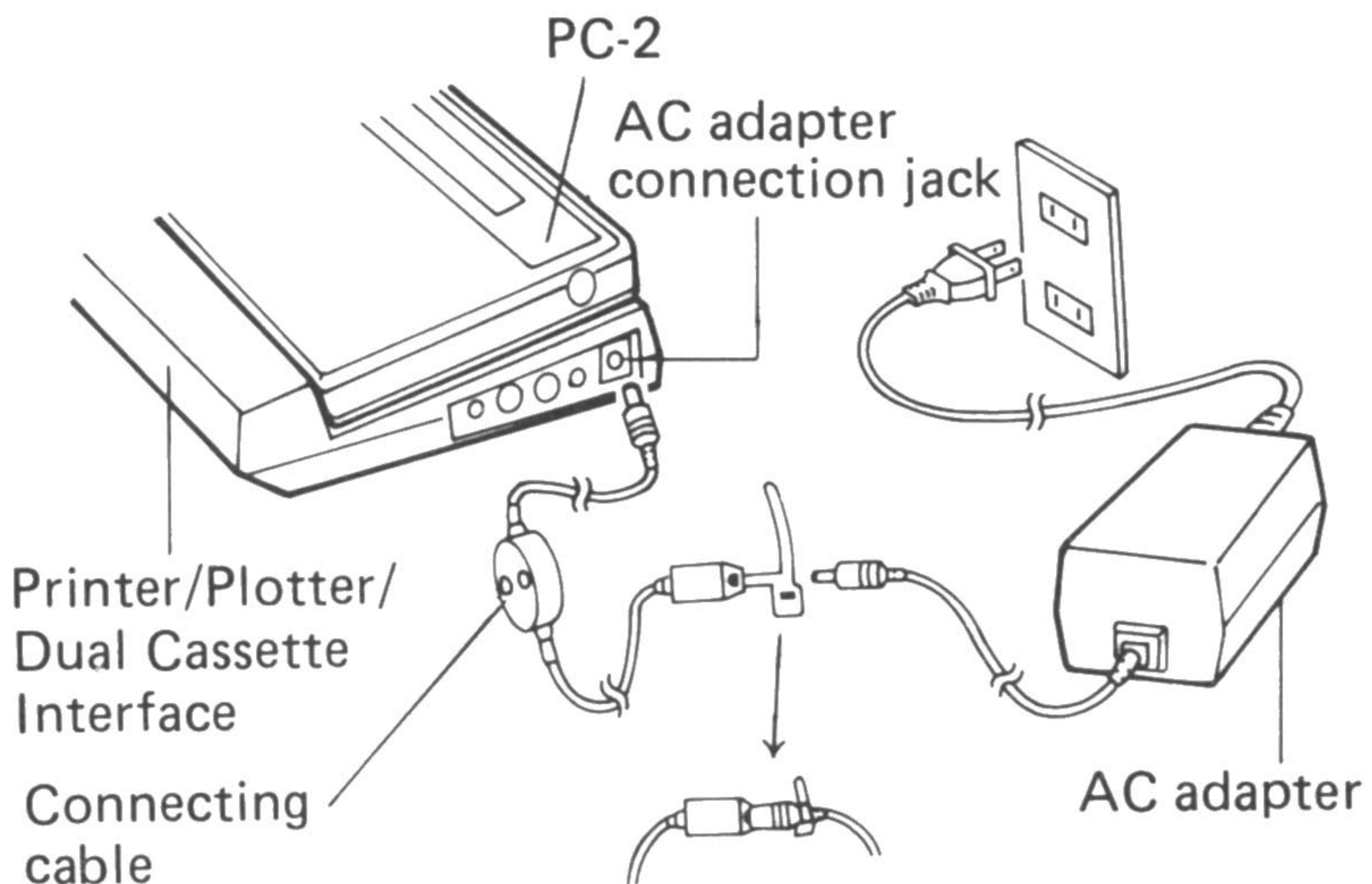How to connect the AC adapter and the connecting cable.

Figure 6. Using the AC Adapter with the Printer/Plotter/ Dual Cassette Interface.

Connecting the Assembled Interface to a Communication Device

To connect the RS-232C Interface to a communication device (i.e., a modem or another TRS-80), you will need an DB-25 cable such as the Model III RS-232C Cable (26-1408). (If you use the PC-2 to communicate with another TRS-80, you'll also need a Null Modem Adapter 26-1494).

For connection to a serial printer, use the PC-2 Serial Printer cable (26-1327).

8

If you are not able to determine which cable you will need to connect with a particular peripheral, contact your nearest Radio Shack Computer Center for advice.

Powering-Up the PC-2 RS-232C Interface

The PC-2 RS-232C Interface has a built-in rechargeable battery pack available as a power source. Also, the PC-2 RS-232C Interface can also be connected directly to an AC Outlet via the AC Adapter included with the package.

When you receive the RS-232C Interface, the battery pack is not "charged" up. Charge the batteries before using the unit.

1. Be sure the power switch of the Interface is OFF.

2. Connect the plug of the adapter (included with this package) to the Interface power supply jack.

3. Plug the Adapter into a wall outlet.

The charging time will be about 15 hours.

When the battery pack weakens and needs to be recharged, the Computer will tell you so by displaying one of these messages:

Message:	Meaning:
: CHECK 8	Check the RS-232C Interface for proper connection or possible recharging.
NEW0?	Check the system for proper connection or possible recharging.
: CHECK 68	RS-232C and Printer/Plotter/Dual Cassette Interface battery packs are low.
ERROR 50	Low power during BASIC programming.
LOW BATTERY	Low power during Terminal Mode operation.

Table 1

When one of the above messages is displayed, recharge the battery pack.

Using AC Power

To use the RS-232C Interface with AC power, connect the Adapter (See "Powering-up the PC-2 RS-232C Interface") and turn the power on.

Once connected to the PC-2 and powered up, the power to the RS-232C Interface can be controlled by the Computer.

- After turning the RS-232C's power ON, turn the PC-2 ON. (: CHECK 8 will be displayed if you don't follow this order.) Power is then supplied to the unit.

- When you turn the PC-2 off, the RS-232C power is also turned OFF.

- When not using the RS-232C Interface, set the unit's power switch to OFF.

10

3/ Using the PC-2 RS-232C Interface

The RS-232C Interface has its own set of commands and functions ("software") which control data communications with other devices. However, the execution of commands and functions will not be successful unless "hardware" is properly configured.

Specifically, this means that the input signals:

- CTS (Clear To Send)
- CD (Carrier Detect)
- DSR (Data Set Ready)

must all be "ready."

- Use the INSTAT command (described later) to check the status of these signals.

- Use the output signals commands RTS and DTR to tell the status of Interface to the other device, are OFF at initial power on. (See OUTSTAT.)

Important Note: The PC-2 must always be connected to the Interface when programming with the RS-232C commands and functions. Also, the Interface must be connected to the PC-2 when you list or execute such programs; otherwise, an error message or ~ (tilde) will be displayed.

The RS-232C Interface provides extensions to PC-2 BASIC as well as a smart terminal program in ROM.

These new commands fall into four general categories:

- Control statements for the RS-232C hardware. This group allows you to configure the RS-232C hardware and built-in terminal software for communication with various devices.

11

- Enhancements to BASIC not related to communications. These commands are not directly related to communications functions but they do add power and flexibility to the existing PC-2 BASIC language.

- Enhancements to BASIC related to communications. This group of commands allows BASIC to interact with external devices, utilizing the RS-232 communications hardware.

- Terminal Mode Commands. These commands allow you to configure and control the machine-language smart Terminal Program.

In this section, we'll describe all of the BASIC Mode commands first. These will be listed in alphabetical order for your convenience.

Control and Status Commands

Control and status commands provide control over various RS-232C protocols and configurations. All are implemented as new BASIC commands.

Since the RS-232C Interface lets you control many types of input and output devices, we'll use the following abbreviations when referring to them. See Table 2.

KI	——	Keyboard Input
DO	——	Display Output (LCD)
PO	——	Printer Output
CI	——	Cassette Input
CO	——	Cassette Output

Table 2

CLOADa

Load an ASCII Format Program from RS-232C

CLOADa

CLOADa reads an ASCII format program from the RS-232C. This means:

- Each line must start with a line number.
- Each line is ended with a carriage return (0DH).
- Maximum length of line is 160 characters, including line number and the carriage return.
- Leading blanks (after a carriage return and before a line number) are ignored.
- File is ended with a null line (carriage return only).
- After compression, a maximum of 80 bytes are allowed.

CLOADa functions only when the cassette input is routed from the RS-232C. Except for its characteristic of reading the program in ASCII format, it functions identically to CLOAD.

It should be noted that the sending device must pause two seconds between lines (after each carriage return) to allow BASIC time to insert the new line into memory.

To load a program in ASCII format through the RS-232C, use the following sequence:

        SETDEV CI
        CLOADa

Note: A program loaded via CLOADa must not contain the Pi symbol or the square root symbol within quoted text.

CLOADr

Load Reserve Program from RS-232C

> **CLOADr "filename"**
>> filename is a standard, previously assigned file name.

CLOADr loads the set of function key definitions and menus (a "reserve program") from the RS-232C. CLOADr functions only when cassette input is routed from the RS-232C (as with SETDEV CI).

The syntax for CLOADr is identical to that for CLOAD.

For example, to load a set of function key definitions and menus via the RS-232C, use the following command sequence:

> **SETDEV CI**
> **CLOADr**

COM$

Fetch Communications Parameters

COM$ provides a way for you to examine the current RS-232C settings. COM$ may be used in either the RUN or PRO Mode as a direct command or part of a program.

This function returns a string of characters defining the current RS-232C setting, in format "baud, word, parity, stop." (This is the same format SETCOM uses to set the parameters.)

As with any function, COM$ must be used in conjunction with a command (such as PRINT COM$) or variable (as in Z$ = COM$).

It should be noted that COM$ always returns a string no longer than 16 bytes; therefore, it can be used safely with undimensioned string variables.

For example, if the RS-232C is currently set at 2400 Baud, 7 Bit Words, Odd Parity, and 2 Stop Bits, type:

PRINT COM$ (ENTER)

and the Display will show:

2400, 7, O, 2

During execution of a BASIC program, reset communication parameters to 300 baud, 8 bit words, even parity, and 2 stop bits, execute subroutine 1000, then restore to original settings.

```
100  A$ = COM$
110  SETCOM  300, 8, E, 2
120  GOSUB  1000
130  SETCOM  A$
```

CONSOLE

Set End-of-Line Protocol

CONSOLE x, y, z

x sets the number of characters sent and is either 0 or a number between 16 ~ 255.

y, z specifies to send a carriage return, line feed, or both. y, z is a combination of 0 and 1. See Table 3 for details.

CONSOLE is used to specify the combination of line feeds and carriage returns transmitted as an end-of-line symbol.

The general syntax for this statement is CONSOLE x, y, z. Both y and z are optional, but if y or z is specified the preceding parameter must also have been specified.

x is either 0 or 16~255, and controls the number of characters allowed in a single block of data sent from the RS-232C. A carriage return and/or line feed sequence is automatically transmitted each x characters. If x = 0, this automatic blocking of records is disabled.

y and z are each either 0 or 1, and each specifies either a carriage return (0) or line feed (1) to be sent at the end of a data record.

If only y is specified, only one carriage return (y = 0) or line feed (y = 1) is sent at the end of a record. If y and z are both specified, then two characters are sent as defined by y and z, respectively.

If y =	and z = ,	Characters Sent
0	0	CR CR
0	1	CR LF
1	0	LF CR
1	1	LF LF

Table 3

For example, to set maximum data line length to 80 bytes and end-of-line protocol to carriage return only, type:

CONSOLE 80, 0 (ENTER)

To set maximum data line length to no limit and end-of-line protocol to carriage return only, type:

CONSOLE 0, 0 (ENTER)

CSAVEa

Save Program to RS-232C in ASCII Format

> **CSAVEa startline, endline**
>> **startline** is the first line of the program you want to save. startline is optional; if omitted, the first line of the program is used.
>> **endline** is the last line of the program you want to save. endline is optional; if omitted, the last line of the program is used.

CSAVEa saves the current program to the RS-232C in ASCII format. This means:

- Each line must start with a line number.
- Each line is ended with a carriage return (0DH).
- Maximum length of line is 160 characters, including line number and the carriage return.
- Leading blanks (after a carriage return and before a line number) are ignored.
- File is ended with a null line (carriage return only).
- After compression, a maximum of 80 characters are allowed.

CSAVEa functions only when cassette output is routed to the RS-232C.

Its syntax is slightly different from CSAVE's, and is used in the form CSAVEa x, y where x and y are both optional and specify the line numbers of the first and last lines to be saved — just like the LLIST x, y. This lets you save a portion of the program if desired.

Note: CSAVEa pauses two seconds between lines.

For example, to save the current program through the RS-232C in ASCII format, use the following command sequence:

SETDEV CO
CSAVEa "PROGRAM"

CSAVEr

Send Reserve Program to RS-232C

CSAVEr "filename"

CSAVEr saves the current function key definitions and menus (a "reserve program") from the RS-232C. CSAVEr functions only when the cassette output is routed to the RS-232C (as with SETDEV CO).

For example, to save the function key definitions and menus (the "reserve program") through the RS-232C, use the following command sequence:

SETDEV CO
CSAVEr "RESRVPRG"

DEV$

Fetch Current Device Routings

DEV$

DEV$ does for the PC-2's current device setting what COM$ does for RS-232C settings. DEV$ may be used in either the RUN or PRO mode.

DEV$ returns a string of up to 14 characters listing the names of the input devices which are currently disabled and whose functions are being handled by the RS-232C.

It is used with the same syntax as any other function which returns a string. For example, PRINT DEV$ and Z$ = DEV are both valid uses.

For example, to display route settings from within a BASIC program, use the following string:

        100 PRINT DEV$

To route cassette input from the RS-232C without altering any other routings, use a similar command sequence to the following:

        100 A$ = DEV$ + ", CI"
        110 SETDEV A$

To route the printer to the RS-232C, execute a subroutine 1000, then return to the original settings, use a command sequence similar to the following:

        100 A$ = DEV$
        110 SETDEV PO
        120 GOSUB 1000
        130 SETDEV A$

ERL

Return Line Number of Error

        ERL

ERL is used with ON ERROR GOTO in error trapping. ERL returns the line number of the statement which caused the current error. If an error is not present, ERL returns 0.

ERL has same syntax as any other function which returns a number in that a statement or variable must be used. For example, PRINT ERL and I = ERL are both valid uses.

To display error and end error line numbers if errors occur in a BASIC program, use the following sequence:

```
100  ON  ERROR  GOTO  9000
   ⋮
9000  PRINT  "ERROR";  ERN;"  IN  LINE";  ERL
```

ERN

Return Error Code

ERN

ERN is for use in error trapping in conjunction with ON ERROR GOTO. ERN returns the number (i.e., code) of the current error, or 0 if an error has not occurred.

ERN has the same syntax as any other function which returns a numeric quantity. For example, PRINT ERN and B = ERN are both valid uses. (See example for ERL.)

FEED

Transmit Carriage Return/Line Feed Sequences

FEED n
> **n** is number from 1 to 65535 n is optional;
> if omitted, 0 is used.

FEED transmits a specified number of end-of-line sequences (as defined by CONSOLE). FEED interacts with the RS-232C just as LF interacts with the printer.

For example, to transmit six end of line groups, enter:

FEED 6 (ENTER)

To list the current contents of an array A$ (10) with three carriage return/linefeed groups at the end of each string, use a command sequence similar to the following:

```
100 CONSOLE 0, 0, 1
110 FOR I = 0 to 10
120 PRINT #-8, A$(I)
130 FEED 2
140 NEXT I
```

Note that FEED 2 will generate three end-of-line sequences because the first is automatically sent at the end of the PRINT statement.

INPUT$

Input a String of Characters from the RS-232C

INPUT$ "prompt"; variable
> **prompt** is the message you want displayed. prompt is optional; if omitted, a question mark (?) will be displayed.
> **variable** represents the data to be typed in.

INPUT$ is a variant of BASIC's INPUT function, differing from that function in two ways.

• INPUT$ requests its data from the RS-232C, instead of the keyboard.

- INPUT$ does not attempt to evaluate input as an arithmetic expression but treats it strictly as a character string.

INPUT$ is effective if, and only if, KI has been routed to the RS-232C via a SETDEV command (that is, SETDEV KI has been specified so that data requests normally filled from the keyboard are filled from the RS-232C).

It is used with the same general syntax as INPUT, the only restriction being that the INPUT$ variable must be a string variable.

If INPUT$ has a quoted string prompt, as in INPUT$ "LOAD DATA"; A$, the prompt is handled in accordance with any current display output (DO) routing.

For example, enter a command sequence similar to the following:

```
100 SETDEV KI
110 INPUT$ A$
```

INPUT%

Fill a String Array with RS-232C Data

> **INPUT% "prompt"; variable**
> > **prompt** is the message you want displayed. prompt is optional; if omitted, a question mark (?) will be displayed.
> > **variable** represents the data to the typed in and must be a dimensioned string variable.

INPUT% is another variant of BASIC's INPUT function. Its sole function is to load an entire array of non-numeric data from the RS-232C.

INPUT% is effective if, and only if, KI has been routed to the RS-232C via a SETDEV command (that is, SETDEV KI has been specified so that data requests normally filled from the keyboard are filled from the RS-232C).

The syntax of an INPUT% statement takes the same form as for BASIC's normal INPUT, with the single exception that the variable must be a dimensioned string variable in the form A$ (*).

Any prompt supplied with INPUT% is not processed according to the current DO routing, but is always displayed.

When INPUT% is executed, consecutive characters received from the RS-232C port are placed in consecutive array elements in row-major order, with each array element being filled to its maximum dimensioned length from left to right.

INPUT% execution stops when:

• a carriage return is received

• the array is full.

Array elements which are not filled during INPUT% are nulled (set to '' '').

For example, to declare a string array Z$ as having three rows and four columns of two-character elements, then fill that array from RS-232C input, enter a command sequence similar to the following:

```
100  DIM  Z$ (4, 3) * 2
110  SETDEV  KI
120  INPUT%  ''RECEIVING ARRAY''; Z$
```

As an another example, suppose that array P$ (1, 2) ∗ 3 has already been properly dimensioned, that SETDEV KI is in effect, and that the RS-232C has been properly configured via SETCOM; then, INPUT% P$ (∗) is executed. The data string received at the RS-232C port is "ABCDEFGHIJLKMN".

First, all elements in array P$ are nulled, or set to " ". NEXT, P$ (0, 0) is filled from left to right with RS-232C input characters to its maximum length (determined by the DIM statement which is in this case three characters). Thus, P$ (0, 0) is set to "ABC".

Next, P$ (0, 1) is filled in the same manner. P$ (0, 1) winds up as "DEF". Then, P$ (0, 2) is filled with "GHI". Next, progressing through the array in row-major order, P$ (1, 0) is filled with "JKL". Then, P$ (1, 1) is filled, but since input stops at "MN", P$ (1, 1) is set equal to "MN" and INPUT% stops.

INPUT # −8,

Input Data from the RS-232C.

> **INPUT#−8, "prompt"; variable**
>> **prompt** is the message you want displayed. prompt is optional; if omitted, a question mark (?) will be displayed.
>> **variable** represents the data to the typed in.

INPUT#−8, differs from INPUT only in that it requests data from the RS-232C instead of the keyboard. Whatever character string is received is evaluated in precisely the same manner as for INPUT (specifically including evaluation of arithmetic expressions). Use INPUT$ if this expression evaluation is not desired.

The syntax of an INPUT#−8, statement takes exactly the same form as BASIC's normal INPUT. If a quoted string prompt is provided, as in INPUT#−8, "SEND DATA "; X, the prompt is displayed in all cases.

For example, to input a string value from the RS-232C port, use a command similar to the following:

        100 INPUT#−8, X$

To input Y from the keyboard (where the input is equal to 0), then input Y again from the RS-232C, use a command sequence similar to the following:

        100 INPUT "ENTER Y "; Y
        110 IF Y = 0 INPUT#−8, Y

MERGEa

Merge an ASCII Format Program

> ### MERGEa
>
>> The files to be merged must be saved with the A ("ASCII") option.

MERGEa is to MERGE as CLOADa is to CLOAD. MERGEa functions only when cassette input is routed from the RS-232C. MERGEa functions precisely the same as does MERGE except that the PC-2 will expect ASCII format input from the RS-232C. This means:

- Each line must start with a line number.
- Each line is ended with a carriage return (0DH).
- Maximum length of line is 160 characters, including line number and the carriage return.
- Leading blanks (after a carriage return and before a line number) are ignored.

- File is ended with a null line (carriage return only).
- After compression, a maximum of 80 characters are allowed.

To merge ASCII format program with the resident program, through the RS-232C, use a command sequence similar to the following:

        SETDEV CI
        MERGEa "PROGRAM"

INSTAT

Input Statuts

INSTAT

INSTAT reads the current RS-232C status and returns it as an integer from 0 to 31. To interpret the status, convert this integer to its binary equivalent (0's and 1's). A 1 will mean the line is "ready;" a 0 will mean the line is "not ready."

This number is a binary bit image of the RS-232C's handshake signal whose bits are defined as follows:

Bit	Signal
0 (write)	DTR (Data Terminal Ready)
1 (write)	RTS (Request To Send)
2 (read)	CTS (Clear To Send)
3 (read)	CD (Carrier Detect)
4 (read)	DTR (Data Terminal Ready)
5 and up	not used

For instance, if you type INSTAT (ENTER) and

10

appears on the Display, you should convert this number to its binary equivalent:

00001010

Since bit counting starts at the right, Bit 0 (DTR) is "not ready," Bit 1 (RTS) is "ready," Bit 2 (CTS) is "not ready," Bit 3 (CD) is "ready," and Bit 4 (DTR) is "not ready."

Note that this means the device the PC-2 is communicating with should have the opposite status.

INSTAT can be used in either RUN or PRO mode and should be used with the same syntax as any function which returns numeric information. For example, A = INSTAT and PRINT INSTAT are both valid.

The following program would display whether or not a "clear to send" status is present (bit 2 is set) during execution of a BASIC program:

```
100 WAIT 0
110 CLS
120 A = INSTAT
130 IF (A AND 4) = 0 PRINT "NOT ";
140 WAIT
150 PRINT "CLEAR TO SEND."
```

OUTSTAT

Output Status

OUTSTAT n

n is a number from 0 to 31 whose binary bit configuration represents the desired RS-232C output status.

The OUTSTAT command forces a certain status to be output at the RS-232C port. OUTSTAT may be used in either RUN or PRO mode.

Bit	Signal
0 (write)	DTR (Data Terminal Ready)
1 (write)	RTS (Request To Send)
2 (read)	CTS (Clear To Send)
3 (read)	CD (Carrier Detect)
4 (read)	DTR (Data Terminal Ready)
5 and up	not used

For example, to force output status to data terminal ready and clear to send (set bits 0 and 4), type:

```
100 A = INSTAT
110 B = A OR 5
120 OUTSTAT B
```

PRINT#−8,

Send Data to the RS-232C

PRINT#−8,

PRINT#—8, differs from PRINT only in that it sends data to the RS-232C instead of the display. Whatever list of data is sent is formatted exactly as it would be for PRINT.

The syntax of a PRINT#—8, statement takes exactly the same form as BASIC's normal PRINT.

For example, to list the contents of an array U(3, 2) to a remote terminal via the RS-232C, use a command sequence similar to the following:

```
100  FOR  I = 0  TO  3
110  FOR  J = 0  TO  2
120  PRINT  #—8,  U (I, J)
130  NEXT  J
140  NEXT  I
```

PRINT and LPRINT Enhancements

The previously existing BASIC statements PRINT and LPRINT are both given extra capabilities with the addition of the RS-232C accessory when their output is routed to the RS-232C channel.

First, multiple commas may be used in one print list as delimiters. For example, PRINT A, B, C, D is an acceptable statement.

Second, with the same results as normal comma and semicolon statements, delimiters may be mixed randomly within a print list. For example, LPRINT A$; B, C$; D is an acceptable statement.

RINKEY$

Fetch Current Character from the RS-232C

RINKEY$

RINKEY$ is used to fetch a single character from the RS-232C. RINKEY$ interacts with the RS-232C in exactly the same way as INKEY$ interacts with the keyboard: if a character is available, it is fetched as a single character string; otherwise null (" ") is returned.

It is used with the same syntax as any other function which returns a string. For example, Z$ = RINKEY$ is a valid use.

To pause run of a BASIC program until a carriage return (CHR$ 13) is received from the RS-232C port, use a command sequence similar to the following:

```
100 A$ = RINKEY$
110 IF A$ <> CHR$ 13 THEN 100
```

The following short BASIC program will further demonstrate RINKEY$. It will check the RS-232C port twice a second and if any character is present at the RS-232C port, it will be displayed:

```
100 WAIT 32
110 A$ = RINKEY$
120 IF A$ = " " PRINT "NO CHARACTER":
    GOTO 110
130 PRINT A$: GOTO 110
```

SETCOM

Set Communications Parameters

> **SETCOM baud, word length, parity, stop bits**
>> **baud** is the baud rate and is 50, 100, 110, 200, 300, 600, 1200, or 2400. On power-up, 300 baud is used.
>>
>> **word length** is the word length or number of bits used to transmit a word 5, 6, 7, or 8. On power-up, 8 is used.
>>
>> **parity** is odd, even, or none. On power-up, none is used.
>>
>> **stop bits** is the number of stop bits per byte and is either 1 or 2. On power-up, 1 is used.

The SETCOM command is used to set the RS-232C's communications parameters — baud rate, parity, word length, and number of stop bits. SETCOM may be used in either the RUN or the PRO mode.

Typing SETCOM (ENTER) resets the RS-232C to their power-up status:

SETCOM Power-Up Status
300 Baud
8 Bit Words
No Parity
1 Stop Bit

This is the same as typing:

> SETCOM 300, 8, N, 1

SETCOM may be used to reset only some of the communications paramters by specifying null parameters for the settings not to be changed. For example, to change only the parity to ODD, you could enter SETCOM, ,O (ENTER)

To reset only the baud rate, use the command:

    SETCOM 1200

To reset the baud rate to 600 and the parity to even, use the command:

    SETCOM 600, ,E

To change the communications parameters from within a BASIC program, enter:

    100 INPUT "Enter new commu params "; A$
    110 SETCOM A$

SETDEV

Route I/O Devices To/From the RS-232C

> **SETDEV device, device, . . .**
>> **device** is two-letter name which specifies a PC-2 I/O device. devices are: KI —— Keyboard Input; DO —— Display Output (LCD); PO —— Printer Output; CI —— Cassette Input; CO —— Cassette Output

The SETDEV command is used to route the PC-2's I/O devices (keyboard, LCD, printer, tape) to or from the RS-232C communication channel. SETDEV is a new BASIC word, and may be used in either the RUN or PRO modes as a direct command or as a part of a program.

Each device specified in the SETDEV command is disabled and its input (or output) is filled from (or sent to) the RS-232C channel.

If the command SETDEV is used alone, all device routings are cleared and all devices are enabled.

As an example, output from the LPRINT statement normally is handled by the printer/plotter. If SETDEV PO is specified, anything that would have been handled by the printer (LPRINT, LLIST, etc.) will now be sent to the RS-232C channel.

For example, to route all printer output to the RS-232C, enter:

SETDEV PO (ENTER)

To route all tape input and output to (or from) the RS-232C, enter:

SETDEV CI, CO (ENTER)

To clear all device routings, enter:

SETDEV (ENTER)

To route cassette input from the RS-232C without altering any other routings during execution of a BASIC program, use a command sequence similar to the following:

```
100  A$ = DEV$ +'', CI''
110  SETDEV A$
```

To route the printer to the RS-232C, execute subroutine 1000, then restore to original settings while executing a BASIC program, use a command sequence similar to the following:

```
100  A$ = DEV$
110  SETDEV  PO
120  GOSUB  1000
130  SETDEV  A$
```

Note: When SETDEV DO and/or SETDEV KI are in effect, display output and keyboard input are routes only when the PC-2 is running a program. They have no effect in Immediate Mode, in which requests for keyboard input and display are always handled normally.

SPACE$

Generate a String of ASCII Spaces

SPACE$ n

> **n** is a number from 0 to 31 which specifies the number of spaces you wish to generate.

SPACE$ n returns a string of n spaces. For example, the string DF$ = "#" + SPACE$ 3 + " * " would set DF$ = "# ⌣ ⌣ ⌣ *"

To print fifteen spaces, type:

PRINT SPACE$ 15 (ENTER)

To print S$, then a number of spaces equal to the length of S$, enter:

```
100  S = LEN  S$
110  PRINT  S$;  SPACE$  S
```

TRANSMIT BREAK, n

Send Long Space

TRANSMIT BREAK n

n is a value from 0 to 255 and represents n/64 of a second. n must be specified.

TRANSMIT BREAK, n is used to send a "long space" over the RS-232C.

For example, to transmit a two-second long space, use a command string similar to the following:

100 TRANSMIT BREAK, 128

ZONE

Adjust Print Zone Formatting

ZONE n

n sets the data zones and is a number from 1 to 31. If omitted, 1 is used.

ZONE is used to modify LPRINT output when that output is routed to the RS-232C (via SETDEV PO). Commas must be used as delimiters in the list of data to be printed.

When a comma separates two items to be printed on the PC-2 Display, the Display is split in half with one item displayed on each side. This allotts to a "zone" of 13 characters.

For example, ZONE 10 followed by LPRINT 2, 3.45 would result in the following character sequence being sent to the RS-232C:

.2 . . . . 3.45 . . . . . (" . " represents a space)

To set the print zone width to 10 characters, use the command:

    ZONE 10

To reset the print zone width during a BASIC program, enter:

    100 INPUT "DESIRED ZONE WIDTH? "; Z
    110 ZONE Z

Terminal Mode

The PC-2 RS-232C Interface includes machine-language smart terminal software in ROM. Operation of the PC-2 under this terminal software will be referred to as operating it in "terminal mode."

The PC-2 cannot operate in Terminal Mode unless there are at least 570 bytes of free memory. If less than this amount is available, ERROR 51 will be generated. Try using CLEAR or NEW to free additional memory.

When operating in Terminal Mode, the (BREAK) (ON) key is used to interrupt terminal operations and return to the Terminal Menu display.

Entry to Terminal Mode Via BASIC

Terminal Mode is entered from BASIC by execution of either the command TERMINAL or the command DTE. Both place the PC-2 in Terminal Mode, but with slightly different actions. Essentially the only difference in the two Terminal Mode access commands are that both initiate different default RS-232C communications parameters.

The TERMINAL Command

To enter Terminal Mode via the TERMINAL command, type TERMINAL (ENTER) .

When Terminal Mode is entered via the BASIC TERMINAL command, the following defaults are set for the terminal software and RS-232C Interface:

- Communications parameters are retained as last set using SETCOM.
- XON/XOFF protocol is ON.
- Echo (display of keyboard input) is OFF (disabled).
- Ext/long space keys are OFF (disabled).

The DTE Command

When Terminal Mode is entered using the BASIC DTE command, the following default parameters are set for the terminal software and RS-232C Interface:

- Communications parameters are set to 300 Baud, 7 Bit Words, Even Parity, and 1 Stop Bit.
- XON/XOFF protocol is OFF.
- Echo is ON (enabled).
- Ext/long space keys are ON (enabled).

Terminal Mode Menus

When Terminal Mode is entered using either TERMINAL or DTE, "--- ENTER MENU SELECTION .." will be displayed briefly, followed by the Terminal Menu. The Terminal Menu is one of four setup and control menus available during Terminal operation. The other three menus are the SETUP, OPERATE, and PROTOCOL.

37

Each of these four menus offers two or three selections, which are made by pressing F4, F5, or F6. To display the menus offered, press and release the up or down arrow keys. Press (↓) to display the next menu.

The Terminal Menus and selections are:

Menus	Selections		
	F4	F5	F6
TERMINAL menu	Ent	Aut	Quit
SETUP menu	Aut	Fnc	Com
OPERATE menu	Nrm	A/P	A/L
PROTOCOL menu	XO/O	Echo	

Terminal Menu

The Terminal Menu is the menu displayed upon entry to Terminal Mode, and offers three functions: Ent, Aut, and Quit. The Ent function lets you enter the Terminal Mode directly without initiating auto log-on protocol. The Aut lets you use a previously defined auto log-on. (See SETUP Menu for details.)

Entering Terminal Mode Directly (Ent)

Press (F4) which corresponds on the display to Ent to enter Terminal Mode directly, without performing an auto log-on dialogue.

Upon entry into Terminal Mode via (F4), the buffer is allocated all available free memory. The printer (if attached) is set to CSIZE 2 and ROTATE 0 and is initialized by the transmission of one carriage return.

Entering Terminal Mode with an Auto Log-On Dialogue (AUT)

Press (F5) to enter Terminal Mode and if an auto log-on dialogue is defined, it will be transmitted at that time. (See Defining an Auto Log-On Sequence in SETUP instructions.)

Upon entry into Terminal Mode via the auto log-on option (F5) , the buffer is allocated all available free memory. The printer (if attached) is set to CSIZE 2 and ROTATE 0 and is initialized by the transmission of one carriage return.

Exiting Terminal Menu to BASIC (Quit)

Press (F6) to exit Terminal Mode and return the PC-2 to the normal BASIC prompt.

SETUP Menu

At the Terminal Mode menu, press (↓) to enter the Setup Menu. The Setup Menu can be used to define an Auto Log-On sequence (Aut), alter the defined functions of keys (F1) through (F5) (Fnc), or change the RS-232C communications settings via (F6) (Com). Each of these options are described in the following discussion.

Defining an Auto Log-On Sequence (Aut)

The PC-2 RS-232C Interface supports an auto log-on procedure and lets you define an auto log-on and/or an auto dialing dialog, depending on the type of modem you have. If you have an automatic dialing modem such as the TRS-80 Modem II, you can enter the telephone number using the syntax required to communicate with the modem. Consult your modem owner's manual for details. If you have a modem or acoustic coupler without an automatic dialing feature, you can enter the log-on dialog (user ID, password, etc.) and dial the telephone manually. Follow the instructions in your modem owner's manual for modem communications settings.

At the SETUP Menu, press (F4) (Aut) to change an existing auto log-on sequence or enter a new one. When you press (F4), you'll be asked, CHANGE AUTO SIGN ? (Y/N).

- Type (N) (ENTER) if you don't want to change the auto log-on sequence.

- If you do want to change the current setting, type (Y) (ENTER) . Then type the new log-on sequence and press (ENTER) . Press (SHIFT) . Next press (=) or (↓) to generate a carriage return in the log-on sequence (which will be displayed as @).

The auto log-on is stored in the reserve area (where the Function Key definitions are stored). If there is not enough room in the reserve area to store the auto log-on sequence, the message, SETTING ERROR, will be displayed.

Changing F Key Functions

At the SETUP Menu, press (F5) to change the defined keys. Each of the five F keys used in Terminal Mode will be displayed in turn, in the format "CHANGE Fn = . . . ? (Y/N)" (where n is a value from 1 to 5 and " . . . . " is the present Fn key definition).

- To change the key's definition, type (Y) (ENTER) . Then type the new definition and press (ENTER) .

- Type (N) (ENTER) to return to the Setup Menu.

- Press (↓) to display to the next function key definition, or if the display is at the last one (F6), press (↓) to return to the Setup Menu.

Only keys (F1) through (F6) can be used as definable keys in Terminal Mode.

Changing Communications Settings

At the Setup Menu, press (F6) to change communications parameters. The display will show the current buffer length, Baud Rate, Word Length, Parity, and Stop Bit settings (in that order). With each current setting, the current communications parameters will display with the prompt ''? (Y/N).

- To retain the current setting, type (Y) (ENTER) .

- To change it, type (N) (ENTER) . Then, type the new setting and press (ENTER) .

If a change was made, the new setting will be displayed. If no change is made, the next parameter will be displayed. When all five parameters have been displayed, the display will return to the SETUP Menu.

Operate Menu

The Operate Menu lets you choose from normal, auto-line, and auto-page operation. Upon entry to this menu, the options' abbreviations, Nrm, A/P, and A/L will be displayed under F4, F5, and F6, respectively.

Selecting Normal Operation (Nrm)

At the Operate Menu, press (F4) to select normal terminal operation. This will disable auto-page or auto-line operation if either is in effect.

Selecting Auto-Page Operation

At the SETUP Menu, press (F5) to select auto-page operation. In auto-page operation, the PC-2 will stop (and send an XOFF if that protocol is enabled) after each 512th character is received.

41

Selecting Auto-Line Operation

At the SETUP Menu, press (F6) to select auto-line opera-
tion. In auto-line operation, the PC-2 will stop (and send
an XOFF if that protocol is enabled) after each carriage
return is received.

Protocol Menu

The Protocol Menu lets you change the XON/XOFF and
Echo protocols. The options (XO/O and Echo) will dis-
play beneath (F4) and (F5) . Press (F4) to alter the
XON/XOFF setting or (F5) to alter the Echo setting.

Alter XON/XOFF Protocol (XO/O)

At the Protocol Menu, press (F4) to display the current
XON/XOFF protocol (On or Off). You will be asked if
you want retain the displayed setting.

- Type (Y) (ENTER) to keep the present setting.
- Type (N) (ENTER) to change the setting.

Altering Echo Protocol (ECHO)

Press (F5) to display the current echo protocol (On or
Off). You will be asked if you wish to retain the dis-
played setting.

- Type (Y) (ENTER) to retain the displayed setting.
- Type (N) (ENTER) to change it.

Special Keys and Functions in Terminal Mode

The PC-2's Terminal Mode is a powerful smart terminal package. Due to the vast capacity of the Terminal Mode, many commands are needed to control its various features and many different indicators are needed to prompt you during operation. This section will describe the additional key functions available when using Terminal Mode.

Definable Function Keys

In Terminal Mode, the Function Keys are used to serve several groups of special functions. These can be divided into three categories —— Mode I, Mode II, and Mode III.

Function Keys in Mode I

In Mode I, keys (F1) through (F5) serve as user-defined keys, just as in BASIC.

Function Keys in Mode II

In Mode II, keys (F1) through (F3) are not used. Keys (F4) through (F6) control line/pagination, exactly as at the operation menu:

(F4) —— normal operation —— indicated by "RUN"
(F5) —— auto-page operation —— indicated by "PRO"
(F6) —— auto-line operation —— indicated by "RESERVE"

Auto-page operation stops (sends an XOFF if that protocol enabled) after each 512th character. Auto-line operation stops after each carriage return is received. In normal operation, neither of the above options is used.

Function Keys in Mode III

In mode III, key (F1) toggles XON/XOFF

(F2) is a backspace key (generates 08H)

(F3) toggles Printer ON/OFF (spooled output)

(F4) generates alternate characters

(F5) is an escape key (generates 1BH)

(F6) is a control key

Toggle XON/XOFF

Press (F1) to toggle XON/XOFF.

(F2)

Backspace Key

(F2) is the backspace key. Note that it will delete the last character received from the buffer and the last character displayed. If the last character received was not a displayable character, the last character displayed may not necessarily have been the last character received.

(F3)

Toggle Print Spooler

Press (F3) to toggle the print spooler On and Off. The print spooler prints the contents of the RAM buffer, which contains the most recent data received (up to buffer capacity) and keyboard input (if the echo protocol is on). The printer cannot be started during receipt of data. You must first press (F1) (XON/XOFF) so that the BUSY indicator is on.

Generating Special ASCII Characters

The following chart illustrates the various key combinations that can be used to produce special characters, rather than the ones normally associated with those keys. The "DEG" segment will be turned on to indicate this.

Key Combination	ASCII Character Generated
(F4) (Q)	1CH (FS)
(F4) (W)	1DH (GS)
(F4) (E)	1EH (RS)
(F4) (R)	1FH (VS)
(F4) (T)	27H '
(F4) (U)	5BH [
(F4) (L)	5CH \
(F4) (I)	5DH]
(F4) (O)	5EH ↑
(F4) (P)	5FH ←
(F4) (K)	60H `
(F4) (H)	7BH {
(F4) (F)	7CH ¦
(F4) (J)	7DH }
(F4) (G)	7EH ~
(F4) (D)	7FH ■

(F5)

Escape Key

(F5) serves as an ESCAPE key, and when pressed generates an ASCII 1BH.

45

(F6)

Control Key

(F6) serves a control key, and can be used to generate control codes according to the following table. Press (F6) ; a RAD will appear to indicate that a control key has been pressed.

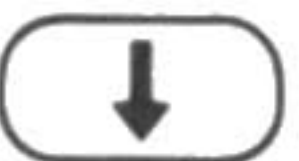| 0 00H	I 09H	R 12H
A 01H	J 0AH	S 13H
B 02H	K 0BH	T 14H
C 03H	L 0CH	U 15H
D 04H	M 0DH	V 16H
E 05H	N 0EH	W 17H
F 06H	O 10H	X 18H
G 07H	P 11H	Y 19H
H 08H	Q 12H	Z 1AH

Graphic Codes Handling

A graphic code is an 8 bit byte with bit 7 (the high bit) set. Of course, they can only be received when SETCOM has specified 8 bit words. When the PC-2 terminal receives a graphic code, it is displayed as a graphic character by modulating the byte to ASCII (0—127) and setting LCD pixels in a bit image of the remaining byte, with bit 0 setting to the top pixel in a column and bit 7 setting the bottom. See GPRINT for more details.

Window Keys and Window Mode

Six key combinations serve as "window keys".

The six Window Functions are:

▶	Moves the display right one word
◀	Moves the display left one word
↑	Displays the start of the previous line
↓	Displays the start of the next line
SHIFT ↑	Displays the start of the buffer
SHIFT ↓	Displays the end of the buffer

Press any of them to:

- Send an XOFF to the remote terminal if that protocol is enabled.
- Activate the Window Mode in which the contents of the text buffer are displayed.

Words are considered to be terminated by spaces or carriage returns. Lines are considered terminated by carriage returns. Carriage returns are displayed as blanks.

CL Key

Provided the Terminal Mode was entered via DTE (hence ETX/long space protocol is enabled), the CLear key sends an ETX character (03H) and SHIFT CL sends a long space of about 1/4 second.

BREAK (ON) Key

Press BREAK to interrupt whatever the terminal program is doing and go to the Terminal Mode menus.

Display Indicators

Several of the PC-2's display indicators are used in Terminal Mode to signify when a certain condition exists. They are used as follows:

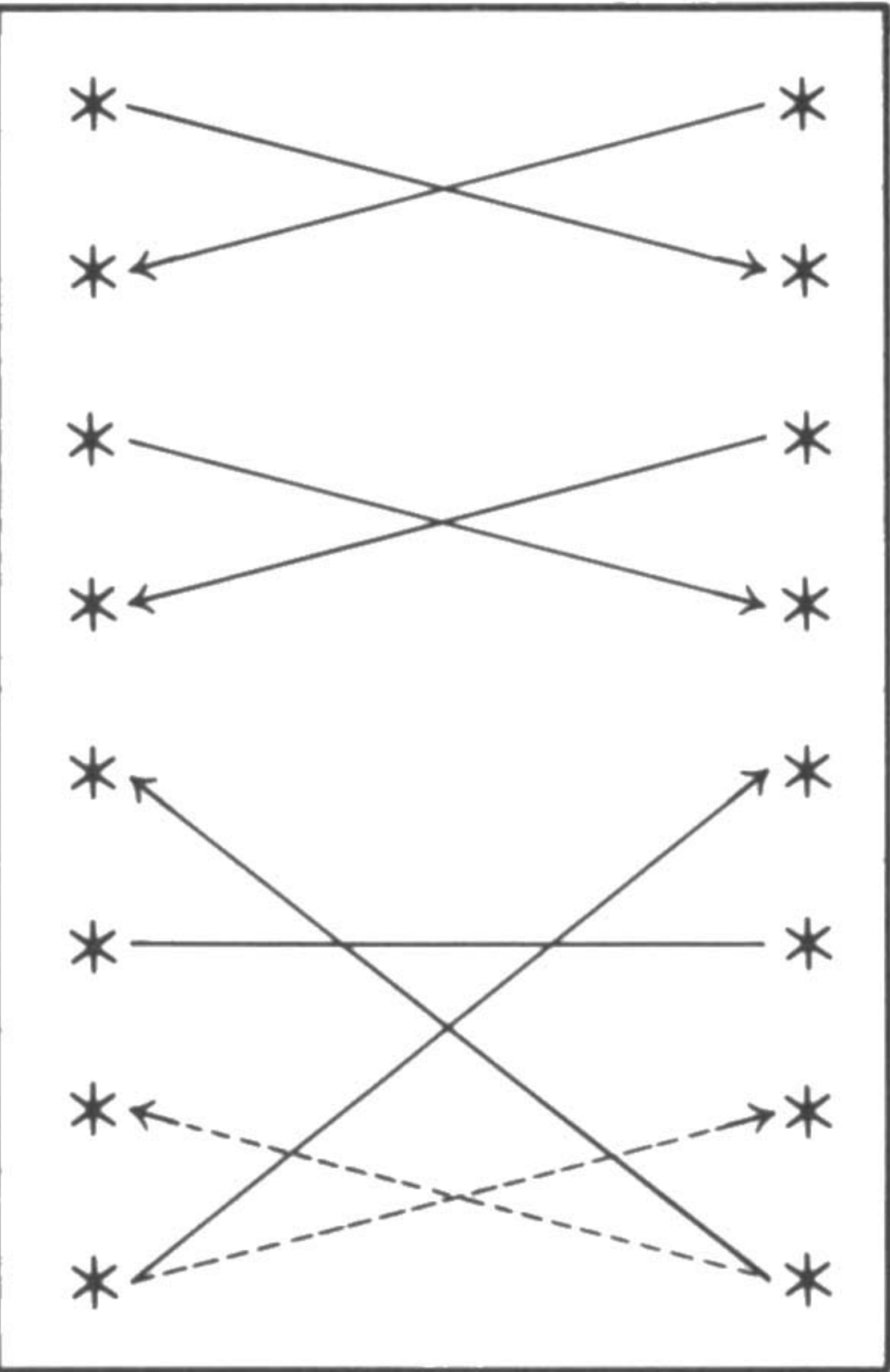| Indicator	Signifies
BUSY	The terminal is not ready to receive data, as when XOFF has been sent. (if that protocol is enabled)
SML	Identical to BASIC.
SHIFT	Identical to BASIC.
DEG	Alternate-character key ((F4) , Mode III) has been pressed. The next keystroke will produce an alternate character.
RAD	Control Key ((F6) , Mode III) has been pressed. The next keystroke will produce a control code.
RUN	Normal (no auto-line or auto-page) operation.
PRO	Auto-page operation.
RESERVE	Auto-line operation.
I	F keys serve as definable keys.
II	(F4) through (F6) select normal, auto-page, and auto-line respectively.
III	(F1) toggles XON/XOFF, (F2) generates 08H (backspace), (F3) toggles printer PRINT ON/OFF (spooled output), (F4) generates alternate characters, (F5) generates 1BH (escape), (F6) is a control key.

4/ Care and Maintenance

- Always make sure that the PC-2, RS-232C Interface, and any peripheral devices are Off before connecting.

- When not using the unit, turn the power switch Off to reserve power.

5/ Technical Information

DB-25 Pin Connections Used

Pin	Name	Symbol	I/O	Function
2	Transmit Data	TD	o	DC output signal
3	Receive Data	RD	i	DC input signal
4	Request Data	RTS	o	ON: sends carrier
5	Clear To Send	CTS	i	ON: xmission enabled
6	Data Set Ready	DSR	i	ON: remote device ready to receive data
7	Signal Ground	SG	—	Reference 0 voltage for all signals
8	Data Carrier Detect	DCD	i	ON: carrier present
20	Data Terminal Ready	DTR	o	ON: local terminal ready

DB-25 Pin Connections

Local Unit (PC-2)   Remote Unit (peripheral)

Pin 2	Transmit Data
Pin 3	Receive Data
Pin 4	Request to Send
Pin 5	Clear to Send
Pin 6	Data Set Ready
Pin 7	Signal Ground
Pin 8	Carrier Detect
Pin 20	Data Terminal Ready

Notes: Pins not depicted are not used.

All pins except pin 6 (DSR) and pin 8 (DCD) are considered OFF if open. Pin 6 (DSR) and pin 8 (DCD) is considered ON if open. If the output signal from the DTR is connected to both DCD and DSR, the output high/low voltages may not be within specifications.

Dotted lines represent optional or alternate connections.

Signal Bit Structure

A.   One start bit
B.   From 5 to 8 data bits
C.   If parity is in use, 1 parity bit
D.   1 or 2 stop bits

A logical "high" signal (+ voltage) is used to represent a binary 1 bit. A logical "low" (− voltage) is used to represent a binary 0 bit.

Character Table in Terminal Mode

LOWER NIBBLE \ UPPER NIBBLE	0	1	2	3	4	5	6	7
0			spc	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7	BEL		'	7	G	W	g	w
8	BKS		(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	|
D	CR		−	=	M	]	m	}
E			.	>	N	↑	n	~
F			/	?	O	←	o	▮

ASCII Program Format

- Each line must start with a line number.
- Each line is ended with a carriage return (0DH).
- Maximum length of line is 160 characters, including line number and the carriage return.
- Leading blanks (after a carriage return and before a line number) are ignored.
- File is ended with a null line (carriage return only).

6/ Specifications

Transmission method: Asynchronous

Applicable Standards: EIA RS-232C compliance

Baud Rates: 50, 100, 110, 200, 300, 600, 1200, 2400 Baud; programmable

Data Bits: 5, 6, 7, 8 Bits; programmable

Parity Bits: Even, odd, or no-parity; programmable

Stop Bit: 1 or 2 (1.5 for 5 Bit words)

Connectors Used: 60-pin male connector for connection with the PC-2 or Printer/Plotter/Dual Cassette Interface. 25-pin connector, DB-25 (W) for connection with an external device. Power Adapter Jack.

Power Supply Source: 4.8V (DC): Ni-Cd rechargeable battery

Power Consumption: 4.8V (DC), 0.65W

Internal Power Supply: Four type AA Ni-Cd cells good for about 3.5 hours of operation

External Power Supply: AC Adapter 26—3612

Output Signal Level: High level: +5V to +10V (3 to 7 K ohms load)
Low level: −5V to −10V (3 to 7 K ohms load)

Interfacing Signals:	Inputs:	RD, DSR, CD, CTS
	Outputs:	TD, RTS, DTR
	Others:	SG

Load Tolerance: 3,000 to 7,000 ohms
Max. capacitance = 2500 pF (including cable capacitance)

Dimensions: 86 (W) x 115 (D) x 50 (H) mm
3-3/8 (W) x 4-17/32 (D) x 1-31/32 (H) iches

Weight: 405 grams or 0.89 pounds

Appendix A/ Interface Language Reference Summary

Control and Status Commands

COM$ Displays current Commination   Page
                  Parameters.   14

    PRINT COM$

CONSOLE x, y, z Specifies combination of line- Page
                  feeds and carriage returns trans- 15
                  mitted as end-of-line symbols.

    CONSOLE 0, 1, 1

INSTAT   Reads current RS-232C status and Page
                  returns it as an integer from 0 to 26
                  31.

    PRINT INSTAT

OUTSTAT n   Forces a certain status to be output Page
                  at the RS-232C handshake port.   28

    PRINT OUTSTAT 31

SETCOM   Sets RS-232C's communications   Page
                  parameters.   31

    SETCOM 300, 8, N, 1

ZONE n   Modifies LPRINT output when that Page
                  output is routed to the RS-232C   35
                  and commas are used as delimiters
                  in the list of data to be printer.

    ZONE 10

Non-Communications BASIC Commands

Comminications BASIC Enhancements

DTE Enters Terminal Communications Mode. Page 37

    DTE

FEED n Transmits a carriage return/linefeed sequence. Page 20

    FEED 65535

INPUT$ Inputs a string of characters from the RS-232C. Page 21

    INPUT $ ''LOAD DATA''; A

INPUT% Fills a string array with RS-232C data. Page 22

    INPUT% ''SEND DATA ''; A$ (✳)

INPUT #−8, Requests data from the RS-232C. Page 24

    INPUT #−8, ''SEND DATA ''; X

MERGEa Merges an ASCII format program. Page 25

    MERGEa

PRINT #−8, Sends data to the RS-232C. Page 28

    PRINT #−8, A, B, C, D

PRINT and LPRINT Expanded capability when output is routed to the RS-232C allows multiple commas in one print list as delimiters and comma and semicolon combined delimiters. Page 29

    PRINT A, B, C, D
    LPRINT A$; B, C$; D

RINKEY$ Fetches single character from the   Page
RS-232C.   30

    Z$ = RINKEY$

SETDEV   Routes the PC-2's I/O's devices to   Page
or from the RS-232C communica-   32
tion channel.

    SETDEV KI, DO, PO

TERMINAL   Enters the Terminal Communica-   Page
tions Mode.   37

    TERMINAL

TRANSMIT BREAK, n   Sends a "long space"   Page
over the RS-232C.   35
"n" can be from 1 to
255.

    TRANSMIT BREAK, 100

Appendix B/ Error Codes

The following Error Codes are applicable to the PC-2 when using the RS-232C Interface:

Error Code	Meaning
50	RS-232C Power Problems —— low battery or off.
51	a) Improper SETCOM parameter, or b) Improper SETDEV parameter, or c) Insufficient buffer space for Terminal mode.
52	RS-232C receive data error (parity, overrun)
53	LPRINT TAB value exceeded record length set by CONSOLE.
58	Received data does not conform to SETCOM spec.
61	Bad type of that header during CLOAD, CLOADr, MERGE and INPUT#.
67	Improper line format during CLOADa or MERGEa. a) line did not start with a number b) line longer than 160 characters as received c) line longer than 80 bytes after tokenization.

"NOTES"

''NOTES''

"NOTES"

IMPORTANT INFORMATION

"This equipment generates and uses radio frequency energy and if not installed and used properly, that is, in strict accordance with the manufacturer's instructions, may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

● Reorient the receiving antenna

● relocate the computer with respect to the receiver

● move the computer away from the receiver

● plug the computer into a different outlet so that computer and receiver are on different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet prepared by the Federal Communications Commission helpful: "How to Identify and Resolve Radio-TV interference Problems". This booklet is available from the US Government Printing Office, Washington, D.C., 20402, Stock No. 004-000-00345-4"

CUSTOM MANUFACTURED FOR RADIO SHACK
A DIVISION OF TANDY CORPORATION

RADIO SHACK, A DIVISION OF TANDY CORPORATION

U.S.A.: FORT WORTH, TEXAS 76102
CANADA: BARRIE, ONTARIO L4M 4W5

TANDY CORPORATION

AUSTRALIA	BELGIUM	U.K.
91 KURRAJONG ROAD	PARC INDUSTRIEL DE NANINNE	BILSTON ROAD WEDNESBURY
MOUNT DRUITT, N. S. W. 2770	5140 NANINNE	WEST MIDLANDS WS10 7JN