TANDY™
COMPUTER PRODUCTS

Cat. No. 26-3143

Orchestra-90™CC

Stereo Music Synthesizer

The FCC Wants You to Know...

This equipment generates and uses radio frequency energy. If not installed and used properly, that is, in strict accordance with the manufacturer's instructions, it 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, you should consult the dealer or an experienced radio/television technician for additional suggestions. You 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 U.S. government Printing Office, Washington, D.C. 20402, Stock No. 004-000-00345-4.

TERMS AND CONDITIONS OF SALE AND LICENSE OF TANDY COMPUTER EQUIPMENT AND SOFTWARE PURCHASED FROM RADIO SHACK COMPANY-OWNED COMPUTER CENTERS, RETAIL STORES AND RADIO SHACK FRANCHISEES OR DEALERS AT THEIR AUTHORIZED LOCATIONS

LIMITED WARRANTY

I. CUSTOMER OBLIGATIONS

A. CUSTOMER assumes full responsibility that this computer hardware purchased (the "Equipment"), and any copies of 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. 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 Tandy Equipment by the original customer from Radio Shack company-owned computer centers, retail stores, and Radio Shack franchisees and dealers at their authorized locations.** 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, a participating Radio Shack franchisee or a participating 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, a 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 EXPRESS WARRANTIES, AND ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE IS LIMITED IN ITS DURATION TO THE DURATION OF THE WRITTEN LIMITED WARRANTIES SET FORTH HEREIN.**

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. SOFTWARE LICENSE

RADIO SHACK grants to CUSTOMER a non-exclusive, paid-up license to use the TANDY 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.

8/84

Tandy Orchestra-90™ CC
Stereo Music Synthesizer
with Percussion
for Tandy Color Computer
Special Composer's Edition

Tandy Corporation
Fort Worth, Texas 76102

Orchestra-90™CC Owner's Manual

10 9 8 7 6 5 4 3 2 1

CONTENTS

INTRODUCTION

Tandy is proud to present Orchestra-90™CC, which lets you become maestro to your own electronic orchestra. The Orchestra-90 CC package is a unique combination of hardware and software that can transform any 16K Tandy Color Computer into a programmable music synthesizer. Orchestra-90 CC does not make your computer a sounding keyboard.

Orchestra-90 CC features:

- The cartridge that connects your Tandy's I/O Connector to your stereo to produce a stereo synthesizer.
- Six octave ranges and up to five simultaneous voices using two stereo channels.
- The easy-to-learn Music Language programming method that lets you enter all components of a piece of music line by line.
- Four or five voices, each of which can be assigned to any of five instrumental choices (tone color registers).

The Orchestra-90 CC software consists of one program in ROM that allows you to create, play, load, edit, and save music to tape or diskette. A Transfer Mode allows you to transmit files via a 300-baud modem.

The Orchestra-90 CC Music Language was designed to let you transcribe virtually any written sheet music into a form that a computer can read and "play." Musicians and non-musicians will find the language simple to learn and easy to use. The language contains the features and capabilities to let you produce highly sophisticated computerized music.

The Music Language program makes it possible to transcribe or compose music in any key or time signature. Diverse rhythm patterns are possible since the available time values for notes range from whole notes to sixty-fourth notes. Notes can be single-, double-, or triple-dotted and/or played as triplets. Single and double accidentals are supported, as are staccato and portato articulations. You can repeat sections of music, produce second endings (with or without retard), and change keys within a music file (modulation).

System Requirements

The minimum system required to run Orchestra-90 CC is:

- A Tandy 16K Color Computer
- A stereo (for stereo sound) or television (for monaural sound)

To save your compositions, you need either a Color Computer Disk Drive or a cassette recorder. To print a music file's contents, you need a Tandy serial printer and appropriate printer cable.

To use the file transfer mode, you need a 300 baud modem and the appropriate cable.

Using This Manual

Years of musical training will certainly give you an advantage when transcribing music using Orchestra-90 CC. However, this manual provides enough background information to help you get started in your composing career. If you plan to create original music, a piano, guitar, or other instrument will be helpful to have so that you can hear in advance the music you are composing.

The Orchestra-90 CC cartridge supports both tape and diskette I/O. After inserting the cartridge and making all connections, you will be shown how to play the built-in demonstration music, the "William Tell Overture." Following the Orchestra-90 CC recital, a brief, yet memory-refreshing music theory review is given along with an explanation of how to transcribe sheet music using Orchestra-90 CC's Music Language.

Composing and transcribing music files is not an easy task at the beginning. The Music Language you use to program Orchestra-90 CC consists of many codes for traditional musical symbols. In this manual, all system commands and Music Language composition codes are explained with examples. The process of creating music files using the composition codes is discussed, followed by notes on saving, loading, and editing music files.

We recommend that you begin by transcribing simple sheet music. Later, after you become comfortable with the Music Language, proceed to more sophisticated music. As with any other music instrument, practice is essential to becoming a virtuoso performer.

Stepping Through Orchestra-90 CC

Composing and arranging music with Orchestra-90 CC can be simplified by dividing the process into steps.

1. Insert the cartridge into the cartridge slot or Multi-Pak Interface.
2. Connect the cartridge to your stereo.
3. Power up all equipment.
4. Play the demonstration piece, or...
5. Enter Orchestra-90 CC's Edit Mode and use the Music Language to compose or arrange music.
6. Return to the Command Mode, and score, then play the music.
7. When you have programmed and arranged the music to your satisfaction, you can write the music file to cassette or diskette for later listening pleasure.
8. Use the Transfer Mode to transmit or receive music files via modem.

When you have mastered the fundamentals of Orchestra-90 CC, you can experiment with its advanced features. Discover the potential of Orchestra-90 CC through the guides to special instrumentation, percussion, and stereo mapping. Stereo mapping lets you direct certain voices to play through the right or left channels of your stereo.

> **Note to diskette users:** Before attempting to save files with the Orchestra-90 CC cartridge, you must have prepared a formatted diskette (using the BASIC command, DSKINI0).

Setting up Orchestra-90 CC 1

Always keep the amplifier at minimum volume when music is not being played.

> **Warning:** The Orchestra-90 CC Interface contains sensitive devices that can be damaged by careless handling. Always turn the computer and amplifier **off** before connecting or removing the Interface. We also recommend that you avoid using non-U.L. approved amplifiers.

Follow these steps to connect the Orchestra-90 CC Interface:

1. Plug the cartridge into the cartridge slot on the right side of the Color Computer or into Slots 1, 2, or 3 of the Multi-Pak Interface. If you have a disk system, the disk pack must be plugged into Slot 4. Set the selector switch to the slot number in which the Orchestra-90 CC Interface cartridge is inserted.

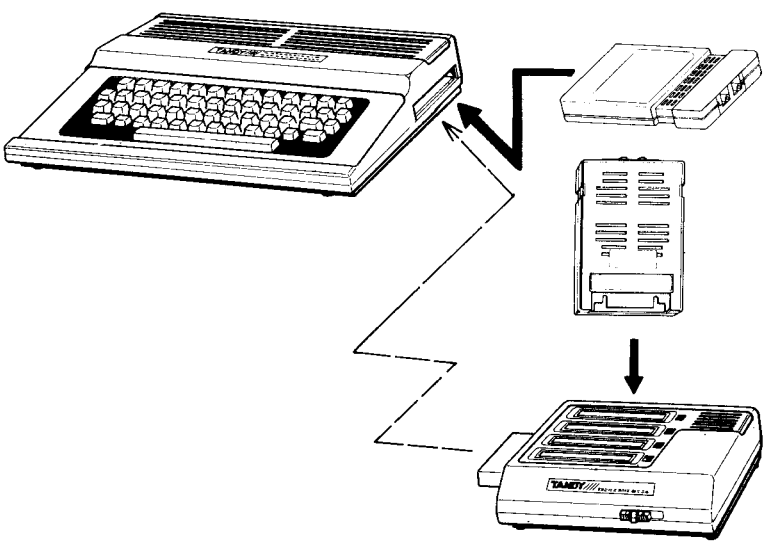


Figure 1. Installing the Orchestra-90 CC Cartridge.

Note: Be sure that your stereo and computer are properly grounded.

The audio sockets on the box mate with standard (RCA jack) phono plugs. Connect one end of the shielded audio cables (not supplied) with phono plugs to the phono sockets on the box. Connect the other end to the AUX, TAPE IN, TUNER, or other high level inputs of a stereo amplifier or receiver. **Do not** use the PHONO, MIC or other low level input. Also, be sure to observe correct Left and Right connections.

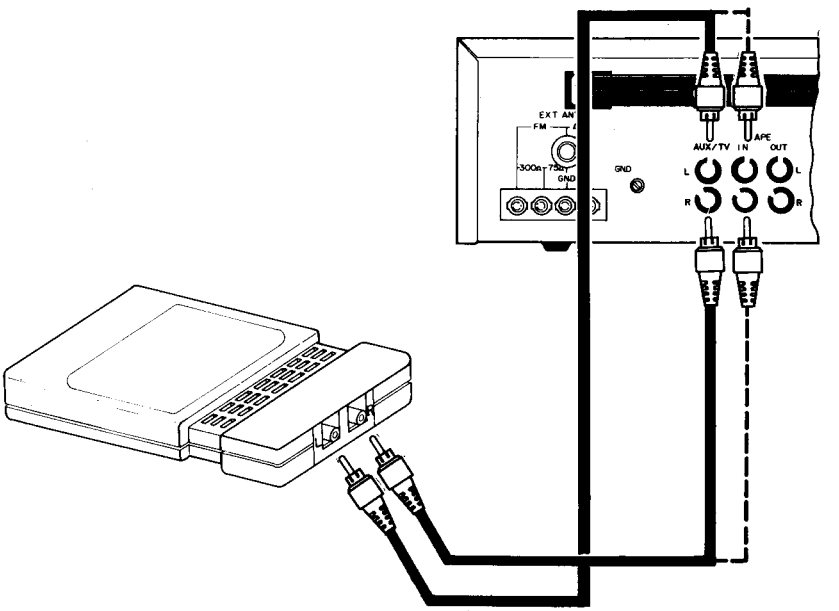


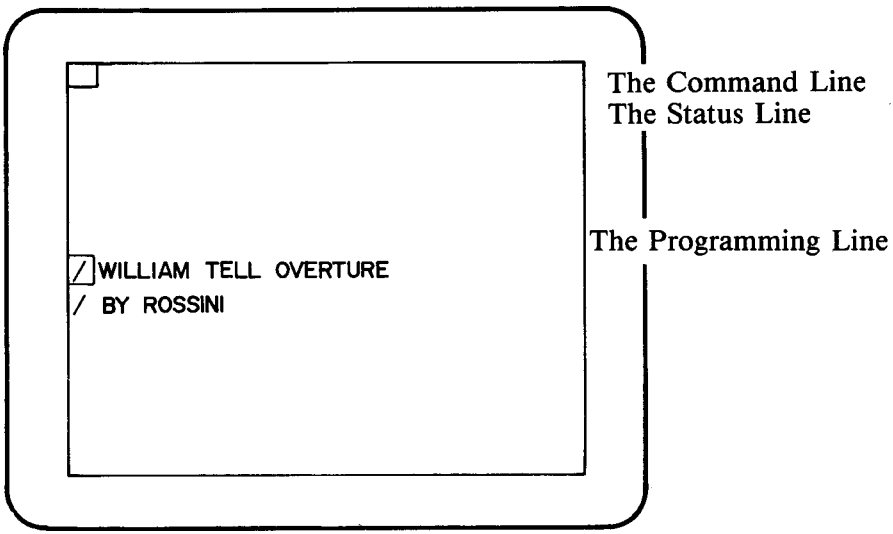
Figure 2. Orchestra-90 CC Cartridge-to-Stereo Connection.

2. After all connections are secure, turn on the computer and amplifier.

When the computer is turned on, the screen displays the Orchestra-90 CC copyright and licensing information. Press any key to display the internal demonstration music file, Rossini's *William Tell Overture.* The blinking cursor in the upper left corner indicates that Orchestra-90 CC is in Command Mode. The background color is light green.

The Orchestra-90 CC Music Composer

The screen shows:



- The Command Line is the location at which you initiate all system commands. For example, you can play a song or write a song to tape or diskette by entering the appropriate command at this line.
- The Status Line is the location at which you receive any messages from Orchestra-90 CC, such as a message that tells you of a programming error in a music file.
- The Programming Line is the location at which you program and edit music using the Music Language.

A Mini-Concert

Now that the cartridge is loaded, you can hear a sample of the way previously programmed music sounds through your stereo. To actually hear what Orchestra-90 CC can do, be sure that your stereo is turned to a low volume and that all equipment is connected correctly.

To play the demonstration file, you must first *score*, or compile, it. Type S (ENTER) , and wait a few seconds until the blinking cursor reappears.

Type P (ENTER) to play the demonstration file. Just sit back and enjoy it. The piece lasts approximately one minute. If you do not want to hear the entire piece, press (BREAK) at any time while the music is playing. The file remains in memory and can be played again by typing P (ENTER) . After completion, or if you press (BREAK) , the blinking cursor at the Command Line reappears.

There are two ways to move from the Command Line to the Music Programming Line. For now, type:

E (ENTER) (E stands for "edit.")

The cursor is blinking at the center of the screen, at the first line of WILLIAM TELL, and the programming lines turn red. Use the (↓) and (↑) keys to scroll through the lines of the WILLIAM TELL file. Holding down an arrow key repeats the up or down scrolling action until you release the key.

As you examine the demonstration piece, keep in mind that the authors took advantage of all possible compiler syntax and reduced the file to its absolute minimum size to fit it into Orchestra-90 CC's single 8K ROM.

Creating Sample Files 2

As you can probably tell, programming a one-minute song requires a great deal of patience, especially in the beginning, when you are learning the Music Language codes. The following sample session was developed to make you, a novice Music Language programmer, feel more comfortable with Orchestra-90 CC programming methods. First, you will learn how to create a music file. Then, you will be told how to play and save that file.

Press (BREAK) at any line in the file to return the blinking cursor to the Command Line.

If you have followed along so far, the blinking cursor should be displayed in the upper left corner of the screen, and if you have just played WILLIAM TELL, the code should also be displayed.
To clear the screen of the WILLIAM TELL file, type N (ENTER) . (N stands for "new.") The file is not "lost"—it resides on the Orchestra-90 CC Interface and can be played again and again. To move to the Programming Line and enter music, type:

E (ENTER)

or

(SHIFT) (BREAK)

The blinking cursor is now at the center left of the screen.

Programming a Short Composition

You can easily program that familiar standby, *Twinkle Twinkle, Little Star,* using the Music Language. The particular version you will program contains no "extras." Only one voice is used—no chords, just the bare essentials. Later, you will encounter the musical condiments that embellish your compositions or arrangements. But before you attempt to transcribe a Bach *Brandenburg Concerto,* carefully type the following:

```
N@=6@ [ENTER]
M1 @3,3,7,7, [ENTER]
M2 @8,8,7,$ [ENTER]
M3 @6&,6&,5,5, [ENTER]
M4 @4,4,3,$ [ENTER]
M5 @7,7,6&,6&, [ENTER]
M6 @5,5,4,$ [ENTER]
M7 @7,7,6&,6&, [ENTER]
M8 @5,5,4,$ [ENTER]
M9 @3,3,7,7, [ENTER]
M1@ @8,8,7,$ [ENTER]
M11 @6&,6&,5,5, [ENTER]
M12 @4,4,3,$ [ENTER]
```

If you make a mistake while entering this information, correct the error using the arrow keys to move the cursor to the error. Type over the mistake. If the corrected line is shorter than the original line, press [CLEAR] to delete unnecessary characters at the end of the line.

Scoring the Music

Before playing this piece, you need to return to the Command Line and score the file. When you score a music file, the system checks for errors and then compiles the piece. Remember that you cannot play a newly programmed piece or a newly edited piece until it has been scored. Return to the Command Line by pressing (BREAK).

To score the piece, type S (ENTER). After a brief pause, the piece is scored. If your file contains a mistake, an error number appears in the upper left corner of the screen. The location of the error in the code is highlighted. Any time you correct an error or make a change in a music file, you must rescore the file before you play it again.

Playing the Music File

After scoring the music file, be sure that your stereo is properly connected to the Orchestra-90 CC Interface. Turn down the volume of the stereo. (You can adjust it later.) The command symbol for playing the piece is P, so type P (ENTER).

After the piece has played, you can listen to it again by repeating the P (ENTER) command.

Saving the Music File

You can save the music file onto cassette tape or diskette. The Orchestra-90 CC Interface lets you switch from cassette to diskette mode for the type of device you wish to use. These instructions show you how to save your piece on a cassette tape player.

First, be sure that you have a new cassette tape in the tape recorder and that the recorder is set to RECORD. Type **C** (ENTER) at the Command Line to be sure that Orchestra-90 CC is set up to work with cassettes.

Again at the Command Line, type:

W TTLS (ENTER) (W stands for "write.")

After you save TTLS, type N (ENTER) at the Command Line to clear the file from the screen and from the computer's memory.

To diskette users: To save it on a disk, refer to **WRITE** command, page 35.

An In-Depth Sample

This portion of the chapter describes the process of transcribing traditional music symbols to Orchestra-90 CC codes. If you have no prior musical education, keep in mind that there is a code for each note and a code for the time value of each note.

To interpret sheet music correctly, it helps to know standard musical notation. Included in this section are several charts and examples designed to help explain a few problems encountered during Music Language programming.

First, you will enter the information contained in the musical passage in Figure 3. The traditional music symbols are explained along with the Music Language substitutes.

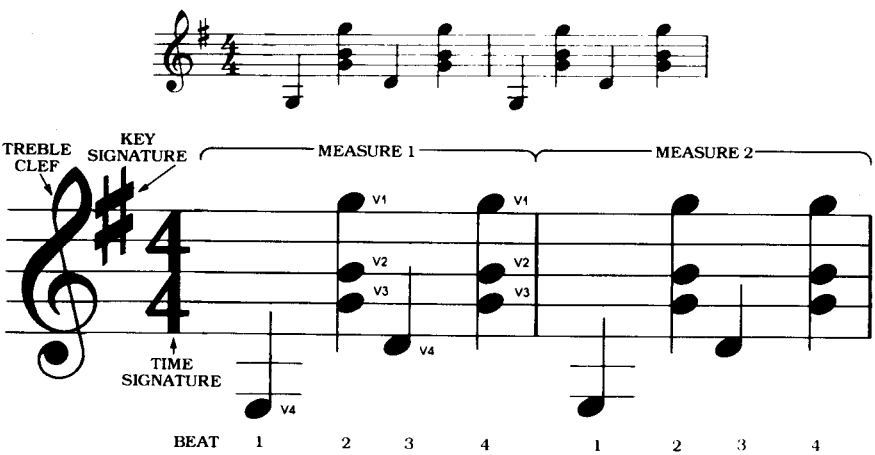


Figure 3. Music Sample #1.

The five, long, horizontal lines are known as *ledger lines.* Together they are known as the *staff.* Most of the information with which you will work can be found on the staff or on additional ledger lines added above or below the staff for extra high or low notes.

Reading from the left side of the ledger line, you first encounter the *treble clef* symbol. All the music in this sample is in the treble clef. The programming symbols for treble clef notes are found later in the manual, in Tables 1 and 2 (pages 42/43). The upper half of the diagram represents the treble clef, and the bottom half is the *bass clef.*

Key Signature. To the immediate right of the treble clef is a single #, known as a *sharp.* The lone sharp in the sample denotes a very important fact about this piece of music—the *key* in which it is to be played. One sharp denotes the key of G.

Begin programming this piece by telling Orchestra-90 CC to play in the key of G. Type K1# (ENTER).

> **Note:** If you have programming experience, you may be wondering about using line numbers. Orchestra-90 CC requires no line numbers for accurate programming.

Time Signature. Refer to Figure 3. The 4/4 on the staff is the *time signature.* In traditional sheet music, this consists of two numbers, one placed above the other, similar to a mathematical fraction. Some examples of time signatures are 4/4, 5/4, 3/4, and 6/8.

The top number shows the number of beats in a *measure.* A measure is a section of the staff between two vertical lines, called *bar lines,* which extend from the top line of the staff to the bottom line. This sample has four beats per measure. You can get the feel of the beat by counting out loud, "1 2 3 4," several times.

Next, examine the bottom number in the time signature, another 4. This 4 denotes that each beat in the music is equal to a quarter note. Therefore, there are four beats in each measure, and each one is equal to a quarter note. This sample contains only quarter notes. Study Figure 3 to be sure that you recognize quarter notes when you see them.

Together, the top number and bottom number for the example is 4/4 time, also known as *common time.* Do not enter a programming symbol for the 4/4 time signature. (Orchestra-90 CC does not require the actual time signature since it automatically accepts any number of beats per measure — up to the compiler limit of 32 notes per voice in a measure.)

Tempo. The next piece of information needed by Orchestra-90 CC can be placed on the same line as the key signature, but for the sake of clarity, you will use separate lines for each new item.

In Music Language programming, you must add a line to describe the tempo, or the speed at which the music is to be played. This programming line bears no resemblance to actual music notation.

Type NQ=80 (ENTER) . This indicates that a quarter note is equal to a time of 80 in hexadecimal, a number system commonly used in programming. If you start out with this tempo value, you can always change it later to speed up or slow down the piece by using tempo conversion, explained later in the manual.

So far, the music file consists of:

```
K1#
NQ=80
```

Voice. As mentioned earlier, you can program up to five *voices* per measure. By programming several voices per measure, you create a *polyphonic* sound. This sample is suited for four voices.

You can program Orchestra-90 CC to make each voice an individual sound, or *register.* Five standard registers are available, known in Music Language as A, B, C, D, and E. (These have nothing to do with the notes A-E.)

A = trumpet
B = oboe
C = clarinet
D = organ
E = violin

To assign the A, B, C, and D registers to Voices 1, 2, 3, and 4 respectively, type:

V1YA V2YB V3YC V4YD (ENTER)

This line indicates that Voice 1 (V1 in your entry) is to be set (Y) at Register A (A), or is to sound similar to a trumpet. Voice 2 is set at Register B to sound like an oboe. Voice 3 is set at Register C and should sound like a clarinet, and Voice 4 is set at Register D to sound like an organ.

Parts. The music will begin with a *part* number. A part contains one or more measures and can be repeated. (Measures cannot be repeated directly, only parts containing measures.) Type:

P01 (ENTER)

Parts consist of any two-digit number. Be careful not to identify two different parts by the same number.

Measures. The next essential Music Language code is the measure symbol. The sample contains two measures, shown by the vertical bar lines that divide the measures from one another. The Music Language measure symbol is followed by more information on the same line, so type it in, but do not press (ENTER) :

```
M01
```

Now, press the space bar, and type * (without pressing (ENTER)). The asterisk indicates that all the notes following it are in the treble clef. (The asterisk signals the compiler that all the notes following are in the treble clef, making it unnecessary to insert a "+" in front of each note.) The line now looks like this:

```
M01 *
```

Entering the Notes. Now you can enter the notes for Voice 1 in Measure 1. Orchestra-90 CC always considers the first line in a measure to be Voice 1, so you do not have to specify V1 in the code. (When you change voices or start with a voice other than 1, you should enter a voice code.)

In Figure 3, V1 has notes on beats 2 and 4 only. Beats 1 and 3 are silent and should be entered as *rests.* The Music Language's code for a rest is $. Orchestra-90 CC needs to be told the length of time that the rest should last. For this sample, the proper code is Q. Q is the quarter note time symbol. It indicates that all notes or rests following it have the time of a quarter note. In this case, the rest symbol will follow a quarter note time symbol, so the first beat becomes a quarter note rest. Type:

```
Q$
```

Next, add the notes for Voice 1. If you check Tables 1 or 2 (pages 42/43), you will see that the symbol for this note in the sample is B. Type B to add it to the end of the line so that the line now looks like this:

```
M01 *Q$B
```

V1 does not appear in the third beat of the measure, so add another rest, making the line appear as:

```
M01 *Q$B$
```

The code for the note on beat 4 is also B. Type B, and press (ENTER) to start a new line. The completed line should look like this:

```
M01 *Q$B$B
```

You can define V2 in the same way. Rests occur in the same places. The only difference in the lines is that the codes for the notes in V2 are harmonizing 6s instead of Bs.

To enter the code for Voice 2 in Measure 1, first press the space bar until the cursor is directly beneath the * in the previous line. Then, type:

```
V2Q$6$6 (ENTER)
```

You changed the voice, making it necessary for the V2 to appear in the line. Q$ signifies that all following rests and notes in this voice are equal in time to a quarter note. 6 is the code for the actual note to be played.

The code for Measure 1 should now look like this:

```
M01 *Q$B$B
    V2Q$6$6
```

Voice 3 has the same timing as the previous voices, with notes that harmonize with B and 6. The harmonizing note's code is 4. To enter V3, move the cursor directly below the V in the previous line, using the space bar. Then, type:

```
V3Q$4$4 (ENTER)
```

You have defined three of the four voices. These first three voices put together produce a *chord* when you play the piece.

Voice 4, unlike the first three voices, has notes on beats 1 and 3. Its rests fall on beats 2 and 4. The first note's code is -3. The note on beat

three is a 1. Enter Voice 4 by moving the cursor to align with the previous lines and typing:

```
V4Q-3$1$ (ENTER)
```

The measure is finished and should look like this:

```
M01 *Q$B$B
    V2Q$6$6
    V3Q$4$4
    V4Q-3$1$
```

Playing the Music. To listen to the music, type:

(BREAK)	(to leave Edit Mode and return to Command Mode)
S (ENTER)	(score the music)
P (ENTER)	(hear the finished music)

Return to Edit Mode by typing:

```
E (ENTER)
```

Measure 2 is an exact duplicate of Measure 1. To repeat Measure 1, or Part 01, use (↓) to move the cursor to the end of the file. Then, type:

```
R01 (ENTER)
```

This tells Orchestra-90 CC to repeat Part 01 (which contains only Measure 1). The finished sample looks like:

```
K1#
NQ=80
V1YA V2YB V3YC V4YD
P01
M01 *Q$B$B
    V2Q$6$6
    V3Q$4$4
    V4Q-3$1$
R01
```

Most sheet music clearly shows all rests. Correct transcription depends upon your awareness of timing. The number of beats per measure, the type of note that gets one beat, and the interpretation of rests are all factors that influence timing.

The next music sample gives you some practice with editing—changing the tempo, altering the voicing, and adding repeats.

Transcribing Additional Samples

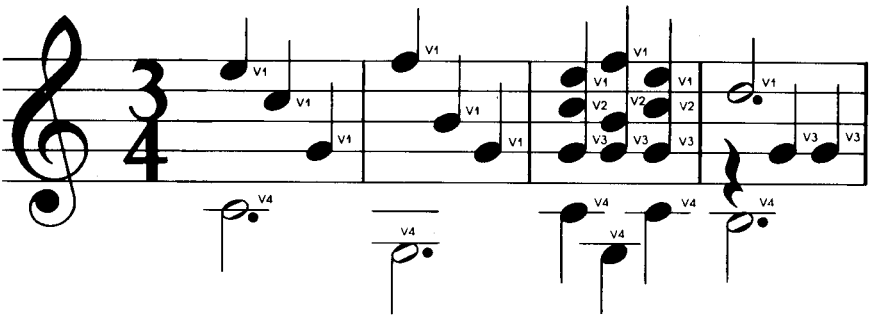


Figure 4. Music Sample #2.

This session describes and interprets the music shown in Figure 4. Study the sample music and the finished transcription. Remember that this transcription is only one possible interpretation. The final decision should be based upon what you believe sounds best.

Type N (ENTER) to start a new composition. Type E (ENTER) to move to Edit Mode. Now, enter, score, and play this piece. Score and play the music after each change you make to the original version.

```
K0#
NQ=80
V1YA V2YB V3YC V4YD
P01
M01 *Q974
    V4H.0
M02 *QA64
    V4H.-3
M03 *Q9A9
    V2Q767
    V3Q444
    V4Q0-30
M04 *H.8
    V3Q$44
    V4H.-1
```

Notice that there are no sharps or flats in the key signature. Your entry of K0# indicates this. Figure 4's time signature tells you that there are three quarter note beats in each measure.

Measure 1 requires two voices—V1 for the melody notes and V4 for the bass notes. The melody notes are usually the highest notes in the measure, and it follows that the bass notes are the lowest. V2 and V3 are not used and therefore are not defined in Measures 1 and 2.

Measure 1 contains a dotted half note, H. , followed by the note symbol for Middle C, which is Ø. A half note lasts twice as long as a quarter note. A dotted half note has its value lengthened by 1/2. The total value of a dotted half note is equal to three quarter note beats. In 3/4 time, three beats fill the measure.

In Measure 2, the timing is the same, but the note symbols are different.

Measure 3 uses all voices and presents another four-part harmony using all four voices. All notes are quarter notes.

Measure 4 uses three voices. V1 consists of a single dotted half note. V2 is ignored. V3 has two quarter notes played after a quarter note rest ($): V3Ø$44

V4 is a dotted half note, -1.

Listen to this piece again, especially Measure 4.

The two notes in V3 have no articulation and sound like one continuous note. Change **V3** in Measure 4 to:

```
V3Ø$4'4
```

The ' is an articulation mark, which makes the notes sound more distinct.

To produce a *staccato* effect, add a semi-colon (;) after each note in V1, Measures 1 and 2:

```
MØ1 *Ø9;7;4;
MØ2 *ØA;6;4;
```

Score and play this new version. The traditional musical notation for staccato is a dot directly above or below the note.

Experiment with articulation and staccato to produce different effects in the music.

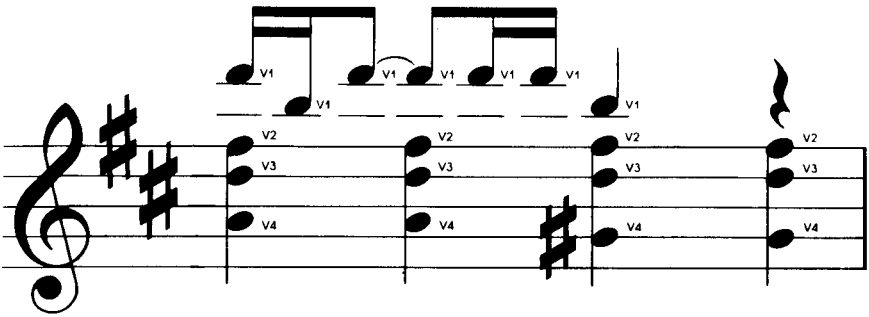


Figure 5. Music Sample #3.

This sample demonstrates an important feature of Orchestra-90 CC. The measure is in 4/4 time. After clearing the previous file from memory and returning to Edit Mode, enter the file exactly as shown:

```
K2#
N0=80
V1YA V2YB V3YC V4YD
P01
M01 *SF'D'IFF;SF;F;QD$
    V20AAAA
    V30888B
    V40554#4
```

Score and play the piece. The melody notes (V1) are so high that you can hardly hear them. You can make a change that brings the melody down to a comfortable frequency in one of two ways.

If you want to lower only V1, you can add a voice modifier after the part number:

```
P01 V1U-7
```

V1U-7 instructs the system to play the notes in V1 one octave lower than written. Make this change. Score and play it. The melody is more audible; however, the other voices are still too high. A better way to change this is to transpose the entire piece down several half-steps. This lowers all the voices equally.

Remove V1U-7 from the line, and replace it with <9. The file now looks like this:

```
K2#
N@=80
V1YA V2YB V3YC V4YD
P01<9
M01 *SF'D'IFF;SF;F;@D$
    V2@AAAA
    V3@8888
    V4@554#4
```

Score and play the sample again. The entire piece has been lowered by nine half-steps, or slightly over half an octave. Try replacing the 9 with an A, C, or any hexadecimal digit up to F. The tone quality of the voices improves, and distortion in the high notes is reduced.

V1 starts out with two sixteenth notes, F and D, each followed by ' (articulation to make these notes more distinct).

The third and fourth notes are eighth notes and have the symbol, F. The curved line in the music sample between these two notes indicates that they are tied, meaning that the two notes must sound without interruption. The second F has articulation to separate it from the two sixteenth notes that follow.

The two sixteenth notes, also F s , have articulation because they are not tied.

A quarter note, D, follows without articulation.

The last beat in V1 is a quarter note rest ($).

Both V2 and V3 consist of four quarter notes. V2 has four A notes, and V3 has four 8 notes.

V4 also has four quarter notes. The first two notes are 5 , and the third note has an *accidental* applied to it. In musical terms, a sharp preceding the note means that the note and all following notes of the same symbol are to be played 1/2 step higher than normal for the duration of that measure. Orchestra-90 CC carries the accidental automatically, so the first note is entered as **4#**, and the following note is entered just like it is written, **4**. The second **4** is played as a **4#** no matter which voice plays it.

After playing the original transcription, change V2, V3, and V4 as follows:

```
V2QA;A;A;A;
V3QB;8;8;8;
V4Q5;5;4#;4;
```

You added ; to every note. Score and play this new version, and you can hear a completely different effect. The chords have a shorter duration, accenting the four beats in the measure.

Using the Orchestra-90 CC System Commands 3

This chapter gives an in-depth description of system commands. The following discussion describes each command and gives examples of practical uses. Some of these commands are ones that you used during the sample session in the previous chapter. In Appendix A is a handy quick reference section, listing brief descriptions of commands and music symbols.

Enter system commands at the top line of the Orchestra-90 CC screen—the Command Line. All system commands consist of a keyword (or keyword abbreviation) and are sometimes followed by one or more operands. Separate operands from each other and from keywords with one or more spaces. You can abbreviate the keyword because only the first character is significant. However, a keyword cannot contain any spaces. If you enter an unrecognized command, ? appears in the Status Line.

One command exists that is not discussed in this chapter. This is the X command, which places you in Transfer Mode. The Transfer Mode is explained in Chapter 8, "USING ORCHESTRA-90 CC WITH COMPUSERVE" in page 73.

The Commands

APPEND *filename*

Add another file into memory.

Append, or add, an existing music file to the beginning of a music file currently in memory using this command. For example, if you had a file currently in memory and wanted to add another file from tape or diskette called FORTE, you would type:

A FORTE (ENTER)

After the appended file has been loaded, the cursor resumes blinking. You can append more files or edit the files in memory. You can also score several files at once to produce a medley. (This may require re-assigning some part numbers.)

BOTTOM

Go to the last line of the music file.

Type B (ENTER) to go to the end of the current file.

CASSETTE

Set up Orchestra-90 CC to run with cassette tape.

Type C (ENTER) to switch all I/O operations to cassette mode.

DIR : *number*

Display the diskette's directory.
Set up Orchestra-90 CC to run with a disk system.

If you have a disk system, you can verify the directory's contents by typing D, a space, a colon, and the drive number, followed by (ENTER). This command displays the names of only the Orchestra-90 CC music files on the diskette. It also sets up the system for using diskettes rather than cassette tapes. For example, to check Drive 1's directory, type:

D :1 (ENTER)

The directory for that diskette then appears on the screen.

EDIT

Press (SHIFT) (BREAK) to return to the Programming Line of a music file already in memory. This method of entering the Edit Mode leaves the last command entered in Command Mode intact. You can make changes, exit the Programming Mode, and search for the next occurrence of a string without retyping the appropriate search command. Exit the Programming Mode by pressing (BREAK) again.

You can also type E (ENTER) at the Programming Line to enter the Edit (programming) Mode. The music file text is now highlighted.

GET *filename 1 filename 2 filename 3 filename 4*

Read, score, and play the specified file(s).

This is a multi-function command that performs the R, S, and P commands for each of the filenames you specify. Overlap is performed automatically if necessary.

If you are a cassette user, please note that you must have the REMOTE plug connected to the recorder; otherwise, the tape continues running while the first file is playing. The G command automatically switches the cassette player on and off between pieces only if the REMOTE plug is connected.

Cassette users: Filenames can contain * , as in the R command. The G * command reads, scores, and plays any file on the tape.

Diskette users: Filenames can be followed by a drive number, just as in the R command.

Filenames may contain an * , as in the R command. The G * command reads, scores, and plays the next available file on the disk. (See the explanation of R * .)

KILL *filename* (diskette users only)

Delete a music file.

You can eliminate a music file from your diskette directory. To delete a file,type K *filename* (ENTER) . Then, check your directory to be sure that the file has been deleted.

LIST

Print out a music file.

If you have a serial printer, you can print out music files for future reference. Be sure that your printer is correctly connected to your computer and is on line and ready to print. Type:

L (ENTER)

The printer starts printing the contents of the music file in memory. Printing begins at the current programming line and continues to the end of the music file. To stop printing, press the reset switch. Be sure your printer is set to serial interface mode and the baud rate is 600. If your printer does not support 600 baud, use the variation command listed below.

Variations of the L command are:

L = 600 baud (or previous baud rate)
L 1= 1200 baud
L 2= 2400 baud
L 3= 300 baud
L 6= 600 baud

Adding a W to the L command indicates that a pause is required after each carriage return, as in:

L 2W (ENTER)

If you "lock up" the system by entering these commands without having your printer ready to print, press the reset button to return control to Orchestra-90 CC without losing the file in memory.

MULTI *filename 1 filename 2 filename 3 filename 4*

Multiple GET for perpetual play of specified files.

For diskette users, this command acts as a perpetual G command and is useful in background music application. Each file loads and plays. After the last specified file plays, the system automatically starts the process again and plays the same files again.

For example, to load and play three files repeatedly, you might type:

```
M WALTZ SONATA POLKA (ENTER)
```

A variation of the M command is **M ***, a special command for tape or diskette users.

Cassette users: M * gets, scores, and plays all files on the cassette.

Diskette users: M * gets, scores, and plays all files on the diskette. After the last file plays, it automatically starts again with the first file.

In either mode, * matches any characters, so the command, **M MUSIC ***, gets, scores, and plays only the files starting with the letters, "MUSIC".

NEW

Clear the music file buffer.

Type N (ENTER) to delete the file currently in memory and begin a new file.

OPTIMIZE

Optimize music file currently in memory.

Type O (ENTER)

This command deletes all unnecessary measure strings, spaces, and blank lines. The file is reduced in size but still can be edited, scored, and played normally. (The function performed by the O command is similar to "packing" a BASIC program.)

PLAY *part number*

Play from the part number specified.

This command tells the synthesizer to play the most recently scored music file, starting with the part you specify. If you omit the part number, the entire piece plays. Press (**BREAK**) to stop. An error code appears if the part cannot be found or if the text file was not recently scored.

For example, to play from Part 03 to the end of a piece, type:

P 03 (**ENTER**)

READ *filename*

Load a file into memory.

Cassette users: This command loads a music file into memory but does not play the file when it is located. To READ a file, type:

R *filename* (**ENTER**)

Orchestra-90 CC searches the tape for a matching filename. When found, the file is read into memory, replacing any previous file there. Typing an asterisk (*) anywhere in the filename tells Orchestra-90 CC to match any character in that position. For instance, the command, **R xx***, reads the next file with a name beginning with "xx." During the search, the Status Line displays the name of each file encountered. You can terminate the operation at any time by pressing the reset button.

Diskette users: Orchestra-90 CC searches the diskette(s) for a matching filename. When found, the file is read into memory, replacing any previous file there. If not found, an error code is displayed in the Status Line. The filename can be followed by a drive number. Typing an asterisk (*) in the filename tells Orchestra-90 CC to match any character in that position. For example, **R *** loads the next available file into memory. The next **R *** that you enter loads the following file, and so on, until the end of the disk directory is encountered or until you enter a specific filename using no asterisks, or until you use a **DIRECTORY**, **KILL**, or **WRITE** command. Reaching the end of the directory or entering one of the specified commands resets the program to start at the first file on the diskette.

SCORE

Compile a music file.

Before you can listen to a music file, use this command to score, or compile, the current file into the binary (coded) form required by the synthesizer. You cannot play a newly created music file until you score the file.

When using the Score command, the compiled binary code exists only in memory during play of a song. The compiled binary code cannot be saved to diskette.

After programming part or all of a music file, press (BREAK) to return to Command Mode. Then, to score the music file, type:

S (ENTER)

You can return to the Command Line at any stage of composition, score the music file, and then play the file. Remember that a file cannot be played before it has been scored. During the scoring process, the system evaluates your file, and if it contains a mistake, such as a non-existent note, an error code appears in the upper right portion of the screen. A white block appears at the point of error. (Refer to Appendix B for a list of error codes.) After you correct the error, press (BREAK) to return to the Command Line, and score the file again.

The S command automatically determines whether a four- or five-voice synthesizer is required for the music file in memory. If the command detects a V5 in the file, a five-voice synthesizer is used; otherwise, the standard four-voice synthesizer is used.

You might prefer the four-voice synthesizer for most music transcriptions because it offers optimum sound quality. The highest, "cleanest" notes are possible in this mode, and you can define more partial harmonics in your special register definitions.

The five-voice synthesizer has slightly reduced sound quality, since the processor must now produce five voices in the time it normally produces four voices. The highest notes may *alias*, so use the < command to transpose the piece down a few notches. It might also be necessary to define fewer partials in your special instrument definitions to avoid aliasing.

The advantage of using five voices is that you can program and play five-part harmony. To many people, the addition of a fifth voice is a fair trade-off for more frequent aliasing on the higher notes. You

can download and play any music file created with Model III/4 Orchestra-90 on Orchestra-90 CC. Some transposition or reduction of higher defined partials may be necessary to reduce aliasing.

You can also remove a voice from a five-voice file and play it in four voices to improve the quality, or you can force Orchestra-90 CC to compile it with the four-voice synthesizer to preview how it sounds in four voices. The following commands let you to force Orchestra-90 CC to use a four- or five-voice synthesizer during compiling and play:

4 (ENTER) Score with a four-voice synthesizer

5 (ENTER) Score with a five-voice synthesizer

Both commands replace the regular S command and force Orchestra-90 CC to use either a four- or five-voice synthesizer. In other words, the 4 and 5 commands override the S command, which automatically selects the appropriate four- or five-voice synthesizer for the file in memory.

If you have a five-voice file in memory and compile the file by typing 4 (ENTER), the file plays in four-part harmony, ignoring Voice 5.

If not enough memory is available to hold the music file and the compiled code at the same time, the program pauses and displays the question, OVERLAP?, in the Status Line. If you answer Y, compilation resumes, and the code overlaps portions of the original music file. Answering anything else cancels the compilation and enters the Command Mode.

The OVERLAP function is useful when compiling large files. The end of the compiled code can overlap the beginning of the music file. **Do not overlap a file that has not been saved on diskette or tape.** Overlapping implies the probable destruction of the current file. Do not expect to be able to use your current music file in any way once you have overlapped it.

TOP

Go to the first line of the music file.

Type T (ENTER) to move the cursor to the beginning of the current file.

VERIFY *filename* (for cassette users)

Verify that file is written to tape.

After writing a music file to cassette using the W command (with the file still residing in memory), you can verify that the file has been saved by typing V and the filename and pressing (ENTER) . If the file has not been saved correctly, an error code is displayed. If an error code appears, repeat the W procedure, and verify again.

For example, to verify the saved file, ALLEGRO, type:

```
V ALLEGRO (ENTER)
```

WRITE *filename*

Save a file by the name assigned, using up to 8 characters.

You can save a file to tape or diskette at any time, regardless of whether or not you have scored the music file. The procedure is somewhat different for cassette users and diskette users.

First, type C (ENTER) to set Cassette Mode or D (ENTER) to set Disk Mode.

Cassette users: Use this command to save music files to cassette tape for later listening or editing. When using this command, be sure that you insert a new cassette tape into the cassette recorder. Set the recorder to record. Type W, followed by an original filename for the music file, and press (ENTER) . For example, to save a file called MOZART, type:

```
W MOZART (ENTER)
```

Diskette users: Use this command to save music files to diskette. To save a music file to diskette, type W, an original filename, a colon, the drive number (if you are using more than one disk drive), and press (ENTER) . The current file is saved with the filename you supplied. For example, to save a file called ALLEGRO to Drive 2 of your system, type:

```
W ALLEGRO:2 (ENTER)
```

The cursor resumes flashing after the file has been saved.

/ *string*

Go to the specified string in the following lines.

Use this command to search from the current cursor location in the music file for a specific phrase, word, or number. Type /, followed by the item for which you are searching, and press (ENTER) . (Hold down the (ENTER) key for a continuous search.) If the string or item is located, the Programming Line's cursor pinpoints it. The program returns to the Command Line. To perform another search for the same item, simply press (ENTER) again.

For example, to search a music file for a string such as P01, type /PØ1 (ENTER) at the Command Line.

– *string*

Go to the specified string in the previous lines.

This command is similar to the */ string* command; however, this command searches from the current cursor location in the music file through previous lines to the beginning. Like */*, – is an auto-repeat command. That is, the Command Line is not cleared after executing this command. This lets you search forward and/or backward for each occurrence of a string by holding down (ENTER) . After you have searched through the entire file, you must re-specify the command to start the search again.

For example, to search previous lines in a file for a string such as M27, you would type:

```
-M27 (ENTER)
```

@

Pinpoint a passage while the piece is playing. This command lets you pinpoint a wrong note or any specific passage in a piece upon playing the piece. To stop the piece while it is playing, press (BREAK) . Then, at the Command Line, type @ (ENTER) to move the programming area's cursor to the general area at which you stopped the previous play. This command is useful for finding errors heard during play. You must score the file again after using this command before playing it another time.

? (SHIFT /)

Display voicing at cursor position.

This command displays the "voicing" (register assignments and voice transposition) in effect at the current cursor position. Position the cursor anywhere in the music file. Press BREAK to return to Command Mode. Type ? ENTER, and the system displays the register and voice transposition variables on the line below the Command Line.

Also displayed are the I/O mode (C or D) and the synthesizer in use (four- or five-voice).

> **Note:** This command, like @, performs a partial compilation, making it necessary to rescore the piece before it can be played again.

! (SHIFT 1)

Play the current file from the cursor location forward.

This command lets you play a piece starting at the cursor's position in the music file. In Edit Mode at the Programming Line, use the arrow keys to move to the position of the piece at which you wish to start playing. Press BREAK to return to Command Mode. The cursor stays at the point in the piece at which you exited. To hear the remainder of the piece from the cursor position forward, hold down SHIFT and type ! , then press ENTER . This command ignores all notes prior to the cursor location during compilation; therefore, if the portion compiled contains a repeat of measures in the section prior to the cursor location, that repeat has no sound or duration.

The Orchestra-90 CC Music Language 4

This chapter describes the various features of the Music Language that you can use to enter or transcribe compositions with Orchestra-90 CC. The chapter explains the Music Language's symbols and functions and gives examples of using the language to make your music complete and accurate.

Special Notes about Music Programming

Always be sure that you are in the correct mode before you enter symbols or commands. The Programming Line(s) are at the center left of the screen. Command Mode always uses the upper left corner of the screen.

Be aware that when you program, the cursor always remains on the same screen line, and the file scrolls up and down around it.

Key Functions and Commands in Programming Mode

During music programming sessions, use the following keys or key combinations to generate the specified results.

BREAK — Terminates the current process and enters Command Mode.

CLEAR — Erases the character beneath the cursor and everything to the right of it on the line. If you press CLEAR and accidentally delete a line, type SHIFT BREAK to restore that line.

SHIFT BREAK — If you make a mistake while editing a line and have not moved the cursor from that line, this command cancels all changes to that line and restores the original line. The cursor returns to the beginning of the line.

SHIFT CLEAR — Deletes the entire line. The line cannot be restored.

ENTER — Accepts the current line as command or edit input.

Moves cursor to the left one position.

Moves cursor to the right one position.

Note: Left and right cursor movement wraps around at the logical end of line.

Deletes the character under the cursor and shifts the remainder of the line to the left. The cursor does not move.

Inserts a space under the cursor by shifting the character at the cursor and the remainder of the line to the right. The cursor does not move. Any character shifted past the end of the line is lost.

Moves the cursor to the line above the current line toward the beginning of the file.

Moves the cursor to the line below the current line toward the end of the file.

Joins the current line to the one above it, creating one long line. Position the cursor at the beginning of a line you wish joined with the one above it. Type SHIFT ↑. This command is effective only if the cursor is at the beginning of a line. Note that this command "undoes" only one previous SHIFT ↓ command. (See the explanation of SHIFT ↓ on next page.)

Inserts a new line between lines if the command is used with the cursor at the beginning of a line. If the command is used with the cursor on any other part of the line, the character under the cursor and all characters to the right move down to form a new line.

Makes a copy of the current line and inserts it at the end of the file. Position the cursor at the beginning of the line you wish to duplicate. Type (SHIFT) (?), and the line is added to the end of the music file.

Moves a specified line up one position toward the beginning of a music file. Position the cursor at the beginning of the line you wish to move. Type (SHIFT) (!) and the line moves up.

Converting Notes to Music Language

You will most often use Tables 1 and 2, especially when you begin composing. It might be helpful if you make a photocopy of these tables and place them near your computer for a convenient reference. Note the position of Middle C in both tables. The circled notes in Table 2 represent the Orchestra-90 CC Music Language Scale. The smaller letter above each circled note represents the actual note name in standard music notation.

Note Symbol	Musical Note Name
+G	E
+F	D
+E	C
+D	B
+C	A
+B	G
+A	F
+9	E
+8	D
+7	C
+6	B
+5	A
+4	G
+3	F
+2	E
+1	D
0	Middle C
−1	B
−2	A
−3	G
−4	F
−5	E
−6	D
−7	C
−8	B
−9	A
−A	G
−B	F
−C	E
−D	D
−E	C
−F	B
$	REST

Table 1. Note to Programming Symbol Conversion.

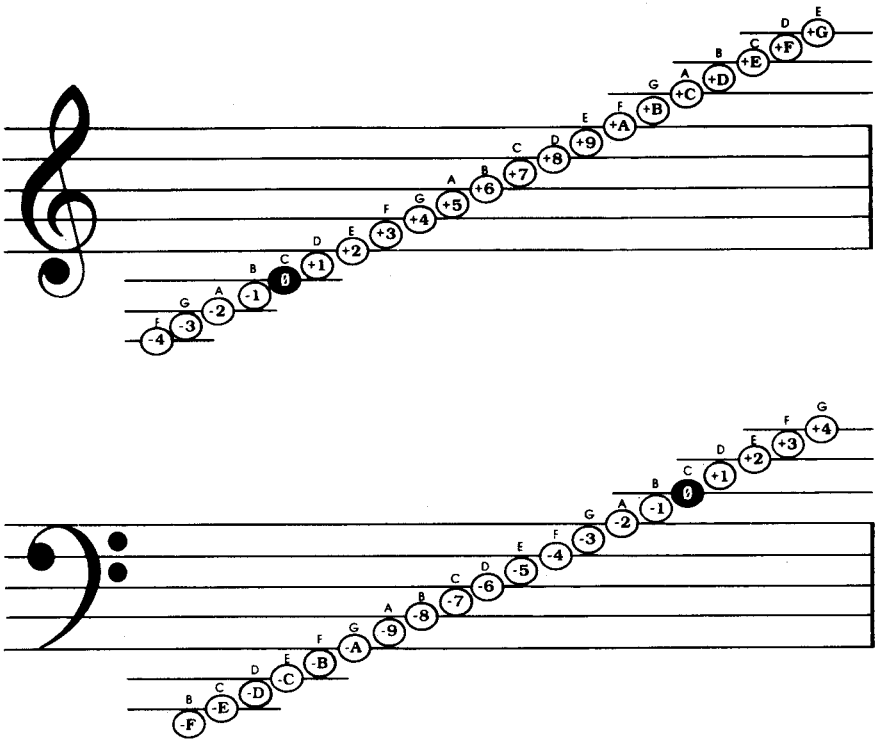


Table 2. Orchestra-90 CC Music Language Scale.

Note: The symbols inside the notes represent the Orchestra-90 CC scale. The small letters above the notes represent the musical scale and are for reference only.

Note and Rest Time Durations

Note	Name	Timing	Programming Symbol
	Whole Note	Held for a duration of four quarter note beats in 4/4 time.	**W**
	Half Note	Held for 1/2 the value of the whole note.	**H**
	Quarter Note	Held for 1/4 the value of the whole note.	**Q**
	Eighth Note	Held for 1/8 the value of the whole note.	**I**
	Sixteenth Note	Held for 1/16 the value of the whole note.	**S**
	Thirty-second Note	Held for 1/32 the value of the whole note.	**T**
	Sixty-fourth Note	Held for 1/64 the value of the whole note.	**X**
	Dotted Note	A period placed after the note duration symbol, H, Q, I, S, T, or X, increases the value of the following note by 1/2. For example, H.0 is a dotted half note played as Middle C.	**.**
	Triplet	A colon placed after a note duration symbol W, H, Q, I, S, T, or X makes the following note a triplet. For example, Q: says that any note following will be a quarter note triplet.	**:**
	Rests	Indicate a rest by following a note duration symbol with a dollar sign. For example, Q$000 indicates that there will be a rest for one beat, after which three Middle C quarter notes will play.	**$**

Articulations

Orchestra-90 CC normally plays successive notes in the *legato* manner, or tied (smooth, with no spaces between the notes) unless you use articulation to provide separation, distinction, and accent of notes. Articulation added to a note reduces its duration by a specific amount and replaces that amount with an equivalent rest, creating a space between that note and the following note.

Orchestra-90 CC offers four levels of articulation from an apostrophe ('), which creates the smallest amount of separation, to a comma (,), which provides the largest amount of separation.

Musical Term	Explanation	Programming Symbol
Short Portato	An apostrophe placed after a note shortens it very slightly and creates the smallest possible separation between notes. This creates a slight accent on each note.	' (Apostrophe)
Long Portato	A quotation mark placed after a note shortens the note by an amount slightly more than short portato. The notes are accented more than in short portato.	"
Staccato	A semicolon placed after a note creates the staccato effect by shortening the note much more than does long portato. The notes have significant accent and distinction.	;
Intense Staccato	A comma placed after a note creates the intense staccato effect, substituting half the note's duration with a rest to produce the strongest articulation available. The notes have the most accent and distinction possible.	, (Comma)

Flats, Sharps, and Accidentals

Musical Notation	Name	Explanation	Programming Symbol
♯	Sharp	# placed after a note signifies an accidental sharp.	#
♭	Flat	& placed after a note signifies an accidental flat.	&
♮	Natural	% placed after a note signifies an accidental natural.	%
𝄪	Double Sharp	# #placed after a note signifies a double sharp.	##
♭♭	Double Flat	&& placed after a note signifies a double flat.	&&
♮♯	Natural Sharp	%# placed after a note signifies a natural sharp.	%#
♮♭	Natural Flat	%& placed after a note signifies a natural flat.	%&

Tempo Conversion

By pressing the **B** , **J** , **R** , **Z** , **2** , and **:** keys, you can change the tempo of a piece while it is playing. Release the keys to play the piece at the tempo in the Music Language code. This is an excellent method for experimenting and finding the tempo at which a piece sounds best. Examine the following Tempo Conversion Table. The more keys pressed down simultaneously, the slower the tempo. For example, pressing all six keys makes the tempo extremely slow.

Once you establish the correct tempo by experimentation, transfer the corresponding hexadecimal value to the music file. The Tempo Conversion Table illustrates each tempo change and gives the hexadecimal value you must enter in the code for that particular tempo.

For example, if your experimentation proves that your piece sounds best when you press down **B** , **J** , and **Z** , and **:** , you can return to the programming line while the code is displayed on your screen and change the tempo specification. In this case, the table shows that those keys correspond to hexadecimal value **AC**. Press (BREAK) to stop the music and return to Command Mode. Next, type E (ENTER) to move to Edit Mode. Go to the tempo specification in the Music Language code, and change it to **AC**. The example of this in code is **NQ=AC**.

Tempo settings below an 80 hexadecimal value can cause an undesirable shift in frequency and should be avoided. It might be necessary to adjust the time signature setting to get the tempo between 80 and FC. Increasing the note value of the time signature allows you to double the value of the tempo. For example, **NQ=C0** is preferable to **NI=60**.

Digits	HeX	Digits	Hex	Digits	Hex
BJRZ2:	FC	BJRZ :	BC	BJRZ2	7C
JRZ2:	F8	JRZ :	F8	JRZ2	F8
B RZ2:	F4	B RZ :	B4	B RZ2	74
RZ2:	F0	RZ :	B0	RZ2	70
BJ Z2:	EC	BJ Z :	AC	BJ Z2	6C
J Z2:	E8	J Z :	A8	J Z2	68
B Z2:	E4	B Z :	A4	B Z2	64
Z2:	E0	Z :	A0	Z2	60
BJR 2:	DC	BJR :	9C	BJR 2	5C
JR 2:	D8	JR :	98	JR 2	58
B R 2:	D4	B R :	94	B R 2	54
R 2:	D0	R :	90	R 2	50
BJ 2:	CC	BJ :	8C	BJ 2	4C
J 2:	C8	J :	88	J 2	48
B 2:	C4	B :	84	B 2	44
2:	C0	:	80	2	40

Table 3. Tempo Conversion.

Other Programming Symbols

( ) ***number***	A programming command enclosed in parentheses followed by a single hexadecimal digit (0-F) indicates that the enclosed sequence is to be repeated. For example, **(16;SD6")3** says to repeat the enclosed musical notation three additional times.
*	An asterisk placed in a music file line indicates that all the notes following are to be played in the treble clef.
–	Place a hyphen in front of a note to play only **that** note in the bass clef. This does not affect a previous treble clef (*) command.
/	Placing a slash at the beginning of a line lets you insert comments (for instance, a title) that is not considered when the piece is scored.
< ***number***	< followed by a single hexadecimal digit indicates that all the notes in the current and following measures are transposed down the number of semi-tones specified by the number.
> ***number***	> followed by a single hexadecimal digit indicates that all the notes in the current and following measures are transposed up the number of semi tones specified by the number.
@	@ indicates that all notes following are to be played in the bass clef.
+	Place a plus sign in front of a note to play only that note in the treble clef. This does not affect a previous bass clef (@) command.
SHIFT **0**	Inserting the left arrow with SHIFT **0** indicates that the notes following are to be played in percussion clef. (See "Special Instrumentation.")
J	J begins Register definition. J followed by the wave form type (Random or Sinesoidal) and nine digits specifies the wave form type, description, and volume. (See "Special Instrumentation.")

K	**K** followed by a digit, **0-7**, and **#** or **&** specifies the key signature and the number of sharps or flats in the key signature. For example, **K1&** indicates that the key signature contains one flat or is in the key of **F**.
M	**M** followed by a number indicates the measure number. For example, **M12** would begin Measure 12. The compiler actually looks for the letter **M** followed by any number of letters or numbers and finally a single space. Measure numbers are ignored by the system. The purpose of specifying measures is to tell the compiler the point at which each new measure starts. The following are examples of valid measure strings: **M01** **MEASURE** **M400001X** **M** **MUSIC** Measure numbers are normally helpful for you when referring from a transcribed piece back to the original sheet music. For example, you could pencil in measure numbers like **M01**, **M02**, and so on, above each measure in a piece of sheet music. Then enter the numbers into the music file. If you are dissatisfied with the played piece or want to rearrange the piece later, use the measure numbers as a quick reference. **Note:** When you optimize a file, all the measure strings in the music file are reduced to a single **M** followed by a space.
N	**N** followed by the **H**, **Q**, **I**, **S**, or **T** note symbols defines the time signature. For example, **NQ** indictes that a quarter note gets one beat. (It is feasible to simply begin with the abbreviation, **NQ = 60**, then adjust the tempo of the song after transcribing and playing the first one or two measures.)

O1	Unless otherwise specified by using the **O** code, an accidental affects all voices in a measure. By entering an **O1** anywhere in a measure, the accidental affects only the voice in which it occurs. The **O1** remains in effect throughout any number of measures until cancelled by **O0**.
O2	The **O2** code carries any accidentals forward into the next measure. This is necessary when you must split a particularly long measure into two or more measures to meet the 32 note per voice per measure limit of the compiler. **O2** must be set in each and every measure in which accidentals are to be carried forward.
O0	The **O0** code cancels both **O1** and **O2**.
P *number*	**P** followed by a two-digit value defines the beginning of a part numbered as the two-digit modifier. For example, **P20** indicates that a new part is begun and that the previous measure is not included in **P20**.
R	**R** followed by a part number indicates that the program is to repeat the notes in that part, using the current tempo and register assignments, then proceed to the next part. For example: **R20 V1YA V2YA V3YA V4YA NQ=80** **R20 V1YB V2YB V3YB V4YB NQ=70** indicates that the compiler should get the notes from Part 20 and repeat them twice, using new tempo and register assignments each time, then play subsequent parts. This process of getting only the actual notes explains the reason that you can change the tempo and register settings for each repeat. Any measure following a repeat must start with a new part number.

U	**U** followed by a plus or minus sign and a single hexadecimal digit **(0-F)** indicates that all the following notes in that particular voice are to be transposed up or down the number of whole steps indicated by the numbers. For example, **V4U-7** would lower Voice 4 seven steps, or one octave.
V	**V** followed by **1**, **2**, **3**, **4**, or **5** indicates the voice you are programming. For example, **V3** indicates Voice 3, which is differentiated from V1, V2, V4, and V5.
Y	**Y** followed by **A**, **B**, **C**, **D**, or **E** follows a voice specification and indicates that the current voice is to be a specific instrument. For example, **V2YE** indicates that Voice 2 is to sound like a violin. (See "Special Instrumentation.")
Z number	**Z** followed by the indicated one-digit number indicates that a certain voice is to be played either from the left or right stereo channel. (See "Special Instrumentation.")

Transcribing Sheet Music 5

The Music Language provides the means for transcribing standard musical notation into the symbolic form usable by the computer.

The use of the language can be shown best by example. Figure 6 is an excerpt in Music Language from J. S. Bach's *Capriccio.* Figure 7 is an excerpt from the sheet music of *Capriccio.* While the two figures may appear to have little in common, both contain essentially the same information.

```
010  /CAPRICCIO
020  /SOPRA LA LONTANANZA DEL SUO FRATELLO
     DILETTISSIMO
030  /J.S.BACH
040  P50/ ARIA DI POSTIGLIONE
050  K2&
060  NQ=E0 /POCO ALLEGRO
062  >2
064  V1YC V2YB V3YB
070  M1      *S6789IABSABA9I.8S7
080          V2@I$4Q1I12Q1
090          V3@Q.8I5Q48
100  M2      *(I6;SD6")3
110  M3      *S7654I365S43"I3;SA3"
120          V2@Q.4"I445%Q4
130          V3@I9768Q7B
140  M4      *I3;SA3"I3;SA9&8767I.5S6"
150          V2@Q$I$4"43Q4"
160          V3@Q$I$785&Q4
170  M5      *(I6;SD6")2Q6;
180          V2@Q4$$4
190          V3@Q6$$6
200  R50
210  P52
220  M6      *I8BA#BC8QB
230          V2*Q6I54S3#
240          V3@S3210*I12S1210@I.1S2
250  M7      V3@(I3;S+43")2I3;+4
260  M8      *I7#S89%IA987#Q8
270          V2*Q5I5654Q3
280          V3@S2345%I63456;S+16"
290  M9      *Q$I$SA3"Q3I$SA3"
```

```
300         V2@I6;S + 16"Q6I$S6D"QD
310   M10   *Q3I$SD6"Q6
320         V2@I$S18"Q8I$S18"I8;S18"
330   M11   *I$SD6"6789IAS67I.5S6"
340         V2@Q8I$213Q4
350         V3@Q$I$7654B
360   M12   *I6;SD6"Q6I$SD6"Q6"
370         V2@Q4I$S18"H8
380         V3@Q6$I$4Q1
390   R52
```

Figure 6. Music Language Transcription of *Capriccio*.

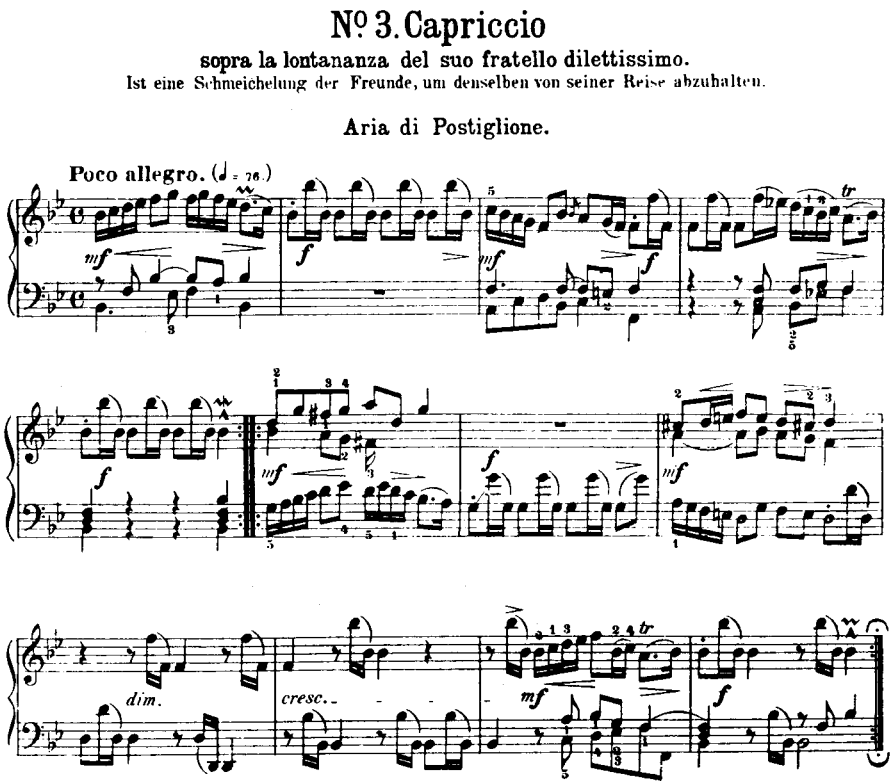


Figure 7. Sheet Music from *Capriccio*

In Figure 6, each line begins with a three-digit number followed by a space. These line numbers are added for the purpose of this discussion only and have no bearing on the music. Remember, the contents of each line effectively begin after the first space.

Lines 10, 20, and 30 are informational only. A slash (/) appearing anywhere in a line causes the remainder of that line to be ignored.

For transcription purposes, consider a piece of music to be subdivided into parts, measures, and voices. A part defines one or more measures played in the same key, in the same tempo, and with the same registration. Logically, a part usually corresponds to a portion of the piece that can be repeated—that is, a phrase, stanza, chorus, and so on. Parts are defined by the letter P followed by either a two-digit number or a space. Numbered parts can be repeated; unnumbered parts cannot.

Measures are indicated by a character string beginning with the letter **M** and ending with a space. The characters following the **M**, usually measure numbers, are ignored by the compiler and serve only as a reference between the printed musical score and the transcribed Music Language text.

A voice is a separate strand of music, in harmony or counterpoint. A trio has three voices, a quartet four. You can define as many as five voices. They are identified as:

V1 (First Voice)
V2 (Second Voice)
V3 (Third Voice)
V4 (Fourth Voice)
V5 (Fifth Voice)

As a convenience, Orchestra-90 CC automatically begins each measure with an "implied" voice, V1, so that you never have to identify V1 in the code. You can, however, specify a different voice number whenever you wish.

In the example, the aria consists of two repeated sections, one five measures long, the other seven measures long. The first section is defined in line 40 as P50. The choice of the two-digit number is arbitrary and serves only to identify the part. However, each defined part must have a unique number. Line 200 defines a repeat of Part 50. Likewise, line 390 is a repeat of Part 52 (line 210).

Line 50 defines the key signature. You need only specify the number

of sharps (#) or flats (&). If you specify no key signature, C major (no sharps or flats) is assumed.

Line 60 defines the time signature and tempo. **NQ** indicates that each quarter note gets a beat. (**H**=half, **Q**=quarter, **I**=eighth, **S**=sixteenth.) =**E0** indicates the relative length of a beat.

Normally, you can determine the tempo setting experimentally by holding down various combinations of the **B** , **J** , **R** , **Z** , **2** and **:** keys while the piece is playing. Next, translate the key pattern, using the Tempo Conversion Table on page 43, to a two-digit number, which you then enter into the music file. You might also find it necessary to alter the time signature setting to achieve the desired tempo.

Line 62 defines transposition. The **<** or **>** character followed by a number defines the direction and number of half-steps the piece is to be transposed. (**<** = down, **>** = up.) You might find it desirable to transpose a piece down a few semi-tones to avoid the distortion present in higher notes.

Line 64 defines the tone color registers to be used by the different voices. The default register is **D** (organ).

In the sample, Voice 1 uses Register **C** (clarinet). Voices 2 and 3 use Register **B** (oboe).

The actual music begins in line 70 with the definition of Measure 1.

Each of the symbols representing a note in standard musical notation implies two pieces of information:

- Its shape, along with the time signature, defines the length, relative to a beat, that the note is to be held.
- Its position on the staff, key signature, and clef define the note to be played.

Transcribing this two-dimensional form into a single line requires two characters to represent each note.

The time value, or shape, is defined by a single letter:

W = whole
H = half
Q = quarter
I = eighth
S = sixteenth
T = thirty-second
X = sixty-fourth

You can modify the time value by using additional symbols. Dotted notes are indicated by a period (.) following the letter. Triplet time values are indicated by a colon (:) following the letter. For example:

Q means that all the notes following are quarter notes.
I. means that all the notes following are dotted eighth notes.

S: means that all the notes following are sixteenth note triplets.

You can combine note length modifiers:

H: .. means that all the notes following are double-dotted half note triplets. (Dot modifiers cannot precede a triplet modifier, so Q.: is not valid.)

The staff position of a note is defined by its relationship to a fixed point on the staff. Middle C is always location 0. Notes above it are defined by a positive (+) displacement, and notes below it are defined by a negative (−) displacement. See Tables 1 and 2 (pages 42 and 43). The displacement uses a hexadecimal-like number scale with the letters A to G representing the numbers 10 to 16. The maximum positive displacement is +G. The maximum negative displacement is −F. A dollar sign ($) defines a rest.

You can simplify coding note displacements by specifying a default displacement sign or clef. An asterisk (*) sets the default to + or treble clef, and all unsigned notes following are assumed to be +. @ sets the default to − or bass clef, and all unsigned notes following are assumed to be −.

Accidentals are indicated with:

 # sharp
& flat
% natural

Accidentals immediately follow the note affected in the Orchestra-90 CC music code. Accidentals stay in effect until the end of the measure or until modified by another accidental. You can use double sharps (# #) and flats (&&) and natural sharps (# % or % #) and flats (&% or %&). Double naturals (%%) and flatted sharps (&#) or sharped flats (#&) give unpredictable results. Avoid using them.

You should now be able to decipher all the symbols in lines 70, 80, and 90 and relate them to the first measure in Figure 7.

Line 100 introduces several new symbols. The parentheses indicate that the enclosed notes are to be repeated. The number following the right parenthesis is the number of times the information enclosed in the parentheses is to be repeated.

Line 100, although coded in the sequence, **M2 *(I6;SD6")3**, plays as if it were coded in the following sequence:

M2 *I6;SD6"I6;SD6"I6;SD6"I6,SD6"

Do not confuse this type of repeat, known as reiteration, with a repeat of a part. Repeats, such as the type in Line 100, can be used any number of times but cannot be nested. For example, (((Q3;3;)))3 is unacceptable.

You can change the articulation of notes by using "expression" modifiers. These modifiers change the way a note is played without altering its pitch or overall duration. Articulation introduces a small rest after a note to separate it from the note following. There are two major types of expression—staccato and portato. Each type has two forms.

Intense Staccato, indicated by a comma (,), shortens the note by 1/2 and adds a rest equal to 1/2 the note's duration.

Staccato, indicated by a semi-colon (;), shortens the note 1/4 its value and adds a rest equal to the note's duration.

Short Portato is indicated by an apostrophe ('). This shortens the note by an amount equal to 1/3 of a 1/128 note and adds a rest of the same value.

Long Portato, indicated by a quote ("), shortens the note by an amount equal to 2/3 of a 1/128 note and adds a rest of the same value.

In all cases, an expression modifier affects only the note it follows. Accidentals, if used, must precede expression modifiers.

So far, we have assumed that the music being transcribed is in bass/treble clef notation. To take advantage of music arranged for different instruments in different clefs (soprano, alto, tenor, etc.), you can define each voice as belonging to a different clef. Voice/clef definition is indicated by a U followed by a number that is the displacement from Middle C in the clef being defined to Middle C in the treble clef. For example:

V1 U-2 V2 U-6 V3 U-8 V4 U-C

defines Voices 1-4 as soprano, alto, tenor, and bass, respectively. Because all clefs are defined relative to the treble clef, each voice is transcribed exactly as if it were the treble clef, and no clef symbols (* or @) need be used.

Another useful transposition is **V4 U-7**, which lowers Voice 4 one octave to get a low bass sound. Since the most common clef transposition is "negative," the default sign is −. So, entering **U7** is the same as entering **U-7**.

Often, when transcribing part-music, the accidentals in one voice interfere with the accidentals in another voice. This is because the compiler applies every accidental to all voices.

Option 1, specified **O1**, limits the application of accidentals to the voice in which they appear.

Option 2, specified **O2**, carries any accidentals forward into the next measure. This is necessary when you must split a particularly long measure into two or more measures to meet the 32 note per voice per measure limit of the compiler. You must set **O2** in each and every measure in which accidentals are to be carried forward.

Option 0, specified **O0**, cancels both **O1** and **O2**.

This concludes the transcription study. You might want to enter, play, and then write the file for future reference. For practice, you could experiment with some of the programming commands and see the ways that they affect the file. Remember, when you are entering the voicing, you can be flexible. There is no set way of voicing a piece. Be creative and bold.

Special Instrumentation 6

An orchestra could not be an orchestra if only one type of instrument played all the time. Orchestra-90 CC lets you create orchestral effects by giving you the flexibility to arrange your music with up to five instruments, including percussion. The following information will help you enhance your final musical product by showing:

- Default instrumentation
- How to change the instrumentation
- How to use the percussion instrumentation
- Stereo mapping techniques

Changing Instrumentation (Tone Color Registers)

The default instrumentations provided with Orchestra-90 CC are based upon the spectral analysis of orchestral instruments and were chosen to provide a wide range of musical sounds. You can easily alter the *registers* to imitate other standard instruments or to create unusual, electronic sounds. You have the flexibility to use harmonic or percussive sounds or to combine the two by utilizing creative voicing.

A register definition consists of the following series:

The letter **J**
The initial of the register name (**A, B, C, D,** or **E**)
The initial of the wave form type (**R** or **S**)
The parameter list consisting of eight digits representing the weight of partials 1-8
A single digit representing volume

The standard registers are all harmonic or wave form type S (Sinesoidal). You can obtain percussive sounds from wave form type R (Random) as well as from type S.

The harmonic registers are defined by the sum of eight sine waves, or *partials*. Each partial is an integer multiple of the fundamental frequency. The first partial is the fundamental, the second is two times the fundamental, the third is three times, and so on. The eight digits in the parameter list define the relative strength or weight of each partial in the register.

This section provides a reference for you when selecting the appropriate instrumentation for the piece of music you are programming. It is easy and fun to experiment with each available instrument. Do not be surprised if you arrive at some unorthodox tones when experimenting with instrumentation. The following discussion illustrates the default registers for both harmonic and percussion instruments and also provides a guide you can use to create your own sound.

The standard default registers are:

JASDFA54E00E	**Define Register A, trumpet**
JBS48F8F200F	**Define Register B, oboe**
JCSE050F000A	**Define Register C, clarinet**
JDSF4080000B	**Define Register D, organ**
JES4F281400D	**Define Register E, violin**

A trumpet-like sound, standard default Register A, is defined as follows:

Symbol	Meaning
J	begin register definition
A	define Register A
S	Sinesoidal wave (harmonic)
E	Partial #1, fundamental, weight E (nearly maximum)
F	Partial #2, first harmonic, weight F (maximum)
A	Partial #3, second harmonic, weight A
5	Partial #4, third harmonic, weight 5
0	Partial #5, fourth harmonic, weight 0
0	Partial #6, fifth harmonic, weight 0
0	Partial #7, sixth harmonic, weight 0
0	Partial #8, seventh harmonic, weight 0
E	Volume E (nearly maximum)

Figure 8. Default Register for a Trumpet.

Each register definition must contain weights for all eight partials. Partials with a weight of 0 make no contribution to the final waveform. The weights of the partial harmonics #1 through #8 are defined using a single digit from the hexadecimal scale, 0-F, where 0 makes no contribution to the waveform and F makes the maximum contribution. (The last character in the register definition (0-F) is defined the same way.)

Because it takes a considerable amount of time to generate a harmonic wave, the compiler will do so only when the register definition changes. You will notice that the first piece compiled takes an unusually long time. This is because all five default registers are being generated. All registers are fully defined to default values unless you redefine them in the music file.

To redefine one of the default registers, simply enter a complete register definition at the beginning of the music file, such as:

JASEFA50000F

The only difference in this register as compared to the standard default register is that the volume has been increased to F, which is slightly louder than the default volume of E. You could also change the weighting of the partials. You can make a voice silent by changing all the partials to 0. Save special register definitions as part of the music file.

Percussion Instrumentation

You can create percussion and other special effects by defining any register(s) as **R**, indicating that a pseudo-random number generator is to be used. The eight digits in the parameter list define the four-digit random *seed* and the four-digit "randomizing" function, respectively. You can experiment by changing these eight digits to create your own unique effects. Because there are billions of combinations of seeds and functions, few generalizations can be made about the kinds of percussive effects available.

Following are two samples:

JBR000300058	Register B, scratchy sounds
JCR101000018	Register C, squeaky sounds

In both cases, the volume is 8.

The sinesoidal wave form can also be used for percussion:

JES80011001E

This register definition is used in later percussion examples. Just as you would not attempt to play a melody on a drum, do not try to play music on a percussive register.

Regardless of the waveform you use, there is a special way to play percussion that requires a unique clef.

The Percussion Clef, defined by ←, has 16 notes defined by the unsigned hexadecimal digits 0-9 and A-F. The ← character is produced by holding down (SHIFT) and typing **0**. This clef also inverts the function of the articulation.

The usual operation of articulation is to shorten a note by a very small amount and insert a small rest at the end. While in the percussion clef, the note is made very short, and a large rest is inserted at the end. This "reverse articulation" creates the short percussive sound of a drum and lets you enter drum rhythms using the same note values found in drum tablature. The ' and " symbols are the best to use for reverse articulation in the percussion clef. The following score demonstrates the features of the percussion clef.

JER80011001A	defines Register E, random wave form, volume A.
JCS80011001F	defines Register C, sine wave form, volume F. Other registers default. Articulation on all notes.
P00<7	is an unnumbered part, piece played one octave down.
M01 V1YD	***I0'1'2'3'4'5'6'7'8'9'A'B'C'D'E'F'Q$** plays a standard scale using default Register D.
M02 V2YE	***I0'1'2'3'4'5'6'7'8'9'A'B'C'D'E'F'Q$** plays the same scale using a random wave form without percussion clef.
M03 V2YE	**←I0'1'2'3'4'5'6'7'8'9'A'B'C'D'E'F'Q$** plays the same scale using a random wave form with percussion clef.
M04 V3YC	**←I0'1'2'3'4'5'6'7'8'9'A'B'C'D'E'F'Q$** plays the same scale using a sinesoidal wave form and percussion clef.

Figure 9. Percussion Demonstration.

In reviewing the demonstration, Measure 1 (**M01**) sounds normal. **M02**, though, sounds very unusual because the notes are not shortened using reverse articulation. **M03** and **M04** are examples of the proper use of register definition, reverse articulation, and percussion clef. Once you define a percussion register, you might wish to play the entire scale using the percussion clef and select the notes that sound best. Some notes in any percussion register may have no sound at all or may not be suited for a particular piece.

Remember that the left arrow (←) defines default percussion clef with reverse articulation. All notes following are percussion notes until the program encounters * or @.

Always define a music voice following a percussion voice by * (treble clef) or @ (bass clef). Typically, use the last voice in any measure for percussion, and define the first voice in the following measure as either * or @.

Stereo Mapping

Stereo mapping lets you balance or position voices in each stereo channel or "ping-pong" voices between channels. The mapping symbol is **Z** followed by a number, 0—9. The default is **Z0**.

Symbol	Channel A		Channel B			Synthesizer
Z0	V1	V2	V3	V4	V5	3, 4, or 5
Z1	V1	V3	V2	V4	V5	3, 4, or 5
Z2	V2	V3	V1	V4	V5	3, 4, or 5
Z3	V1	V4	V2	V3	V5	3, 4, or 5
Z4	V2	V4	V1	V3	V5	4 or 5
Z5	V3	V4	V1	V2	V5	4 or 5
Z6	V1	V5	V2	V3	V4	5
Z7	V2	V5	V1	V3	V4	5
Z8	V3	V5	V1	V2	V4	5
Z9	V4	V5	V1	V2	V3	5

The simplicity of the table hides the difficulty of the implementation. The compiler actually performs the mapping in two parts. Voice mapping takes place at measure boundaries. This means that if the mapping symbol appears somewhere other than at the beginning of a part, a voice may become disconnected from its register. To put it another way, if all voices are playing the same register, you can specify mapping at measure boundaries; otherwise, the results are unpredictable.

Technical Information 7

Theory of Operation

This synthesizer uses a sampling technique commonly used in professional digital synthesizers.

The Sampling Theorem demonstrates that any wave form, no matter how complex, can be reconstructed from a rapid succession of discrete voltages. The reconstructed wave form is actually a stepped approximation of the original. If the steps are small enough, they are not noticed. The size of the steps is determined by the sample rate and the frequency of the wave being reproduced.

Consider one cycle of a simple sine wave. If its amplitude were measured as 100 equally spaced intervals, the values could be recorded in a table. You could reproduce it by serially setting a voltage equal to each entry in the table. Since the table represents exactly one cycle, the frequency of the synthesized wave would be equal to the sample rate divided by 100 (the number of entries in the table). At 1000 samples per second, the entire table would be accessed 10 times per second, producing a 10 Hertz sine wave. Doubling the sample rate would double the frequency of the synthesized sine wave.

Using the same table and sample rate, you could effectively double the frequency of the synthesized wave by accessing every other table entry. This gets you through the entire table twice as fast. Similarly, taking every third or fourth table entry will triple or quadruple the frequency. This is the basis of the synthesizer, with the exception that, to get musically accurate frequencies, it is necessary to skip a fractional number of table entries.

Theoretically, the highest frequency that can be synthesized is equal to half the sample frequency. This is called the *Nyquist frequency.* As this limit is approached, the synthesized wave form becomes more and more distorted because there are fewer samples in each cycle. As the distortion increases, a secondary tone, or alias, is produced. If the Nyquist frequency is exceeded, the synthesized frequency is replaced by its alias. This phenomenon is not limited to the fundamental frequency. It affects each component of the synthesized wave form.

Aliasing cannot be filtered out—it can only be avoided. If aliasing is a problem, redefine the registers with fewer partials or transpose the entire piece down a few half-steps.

Compiler Organization

This section describes certain features of the compiler in more detail. A small amount of technical information will help you understand the way that the compiler works and the way that certain parameters are interpreted.

The output of the compiler consists of two lists: a Part List and a Note List. Each element in the Part List contains a pointer to a Note List element and parameters, such as tempo and registration, that describe how the Note List is to be played. Each element in the Note List contains note frequencies and durations for each of three, four, or five voices.

During compilation, the music file is processed, and the data collected is placed into either a Part Buffer or a Note Buffer. At the appropriate time, the contents of the buffers are processed into the corresponding list.

When the compilation begins, all buffers are cleared, and the default values are entered. If written out, the default values would look like this:

P00	Un-numbered part
K0#	Key of C major, no sharps or flats
<0	No transposition up or down
NQ	Time signature based on a quarter note
=C0	Each beat has a value of C0 hexadecimal
V1 YD U0	Voice 1 uses Register D, no clef transposition
V2 YD U0	Voice 2 uses Register D, no clef transposition
V3 YD U0	Voice 3 uses Register D, no clef transposition
V4 YD U0	Voice 4 uses Register D, no clef transposition
V5 YD U0	Voice 5 uses Register D, no clef transposition
JASEFA50000E	Define Register A, trumpet
JBS48F80000F	Define Register B, oboe
JCSE0500000A	Define Register C, clarinet
JDSF4080000B	Define Register D, organ
JES4F280000D	Define Register E, violin
M V1 *	Begin a measure with Voice 1 in treble clef

The contents of the Part Buffer are added to the Part List whenever a part or repeat symbol is processed or whenever the end of the file is reached. The contents of the Note Buffer is added to the Note List whenever a measure, part, or repeat symbol is processed or whenever the end of the file is reached.

Referring to Figure 6, the part symbol in line 40 causes the contents of the Part Buffer to be added to the initially empty Part List and the Note Buffer to be added to the initially empty Note List. A new Note List element is started, and the Note List pointer in the Part Buffer is changed to point to it.

Lines 50-64 modify most of the default values in the Part Buffer. In line 70, the measure symbol causes the Note Buffer to be processed and the note frequencies to be transferred to the Note List. The Note Buffer is cleared, and any accidentals are reset. (Since the Note Buffer is empty, nothing of interest has happened yet.) The notes defined in lines 70, 80, and 90 are then placed in the Note Buffer.

The measure symbol in line 100 causes the Note Buffer to be processed using the key signature and transposition parameters in the Part Buffer, and the calculated note frequencies of Measure 1 are transferred to the Note List. The Note Buffer is cleared, accidentals are reset, and the rest of the line is transferred to the Note Buffer.

This process of accumulating notes in the Note Buffer and processing them only when a new measure is defined continues until line 200. The repeat symbol causes the Note Buffer to be processed, and Measure 5 is added to the Note List. It also causes the Part Buffer to be added to the Part List and a new Note List element to be started. The Note List pointer in the Part Buffer is changed to point to the same Note List element to which Part 50 pointed.

The part symbol in line 210 acts in much the same way as that in line 50. Similarly, line 390 is like line 210, and the end of the file forces out the remaining buffer contents.

The result of the compilation is a Note List with the following elements:

<1> (empty)
<2> measures 1, 2, 3, 4, 5
<3> (empty)
<4> measures 6, 7, 8, 9, 10, 11, 12
<5> (empty)

and a Part List with elements and Note List pointers as follows:

<00> points to <1>
<50> points to <2>
<00> points to <2>
<52> points to <4>
<00> points to <4>

Notice that Note List elements 3 and 5 have nothing pointed to them. These are the elements created when the repeat symbol is processed. Any notes defined between a repeat and a part symbol are processed into one of these "unclaimed" elements and are never played. In all other respects, repeats are like unnumbered parts.

Repeats can alter any of the parameters in the Part Buffer and thereby repeat sections of music with a different tempo and/or registration. For example, the following code plays a sample scale using each of the different tone color registers.

```
P01  YA  S012343210$  Register A
R01  YB               Register B
R01  YC               Register C
R01  YD               Register D
R01  YE               Register E
```

The default values, NQ=C0 and V1*, are assumed.

Parameters

Parameters take effect when data is processed from the buffer to the list. All parameters remain in effect until changed, either explicitly or by default.

Parameters affecting the key signature (K and O1) or transposition (<, >, and U) take effect at the beginning of the current measure and stay in effect until changed.

Accidentals take effect immediately and stay in effect until changed or until the end of the current measure unless Option 2 (O2) appears within the measure. (Accidentals are also reset by K.)

Parameters affecting the tempo (N and =) or registration (Y) take effect at the beginning of the current part and remain in effect until changed.

Parameters affecting the clef (*, @, and ←) or note duration (W, H, Q, I, S, T, or X, with or without modifiers, . or :) take effect immediately and remain in effect until changed or until the beginning of the next Measure. Measures always begin with a default V1.

Register definition (J) takes effect at the beginning of the piece. You can redefine each register only once per music file.

The stereo mapping parameter (Z) has two different actions that take effect at different times. Voice mapping takes effect at the beginning of the current measure and remains in effect until changed. Register mapping takes effect at the beginning of the current part and remains in effect until changed.

Using Orchestra-90 CC with Compuserve 8

Subscribing to the CompuServe Information Service gives you the opportunity to be a member of the Orchestra-90 Special Interest Group. The Orchestra-90 CC Special Interest Group (SIG) gives you access to 300 available music files (in ASCII format) stored in a data base.

XFER (Transfer Mode)

With a 300-baud modem connected, you can use the Transfer Mode of Orchestra-90 CC to access CompuServe or other Bulletin Board Systems. Transfer Mode lets you directly download music files into your computer. No additional conversion is required. The file may then be played and saved to tape or diskette while still "on-line" with CompuServe. You can also upload your own music files directly into the database. To enter Orchestra-90 CC's Transfer Mode, type:

X (ENTER)

Any character typed in this mode is sent to your modem.

Transfer mode is indicated by light characters on a dark green background with a non-blinking cursor. All alphabetic characters are transmitted and displayed in uppercase. Transfer Mode makes use of several special keys, described below.

(⬇) functions as a control key. As long as (⬇) is held down, all typed characters are transmitted as ASCII control characters.

In Transfer Mode, the keys on the keyboard belong to one of three classes: display, shift, or function. The display keys are the standard alphanumeric keys plus a few additions:

Key	Display	Code
CLEAR	(escape)	$1B
↑	^	$5E
←	(backspace)	$08
→	(tab)	$09
SHIFT 0	_ (underline, but displays as left-arrow)	$5F
SHIFT ↑	\	$2F
SHIFT ←	[	$5B
SHIFT →	]	$5D

The last four require that you use the (SHIFT) key in conjunction with the second key to produce the desired character.

Following are the function keys:

(BREAK) terminates Transfer mode and returns you to Command Mode. If any data exists in the capture buffer, it becomes the current file.

(SHIFT) (BREAK) prepares to transmit the current file in memory by asking for a "prompt" character. Entering (↓) @ means "no prompt." Any other character key you press becomes the prompt character.

After transmitting each line, Orchestra-90 CC waits to receive the prompt character from the host computer before sending the next line.

Pressing (ENTER) in response to the prompt is similar to a "no prompt" except that Orchestra-90 CC waits for its line-ending carriage return to be echoed by the host.

During file transmission, the screen color changes from green to red. Transmission continues until the entire file is transmitted or until the transmission is terminated by hitting any function key.

(SHIFT) (CLEAR) manually opens the capture buffer for downloading and changes the screen color from green to red to indicate that the buffer is open. The capture buffer remains open until you press another function key, until the host sends a "close buffer" command, or until the capture buffer is full. All available memory is used when the capture buffer is open.

Note: The capture buffer automatically responds to buffer "open" and "close" commands as transmitted by various bulletin board systems and time-sharing systems (such as CompuServe). Orchestra-90 CC recognizes the following control characters:

Code	Character	Function
$08	control H	backspace
$09	control I	tab
$0A	control J	linefeed
$0D	control M	carriage return
$12	control R	open capture buffer
$14	control T	close capture buffer

Pressing the reset button breaks a printer lockup without losing the file. Resetting also breaks a cassette I/O lockup.

Accessing and Using CompuServe's Orchestra-90 CC SIG

1. After you have logged into the CompuServe system, you can access the Orchestra-90 CC SIG by typing:

   G PCS-43 (ENTER)

2. After logging into the Orchestra-90 CC SIG, access the data base by typing:

   DL2 (ENTER)

3. Then, to obtain a list of the music files, type:

   BRO (ENTER)

The file names then display with brief descriptions of all files in the data base. Orchestra-90 CC music files have the extension, .A85, indicating that the files are in ASCII format and play on Orchestra-90 CC.

Downloading Music Files

After selecting a music file from the database, you can download it and save it on tape or diskette by entering DOW plus the name of the file you wish to download. For example, the command to download the file, CARMEN.A85, is:

DOW CARMEN.A85 (ENTER)

The first time you use the DOW command (or UPL command for uploading) you are offered the following transfer protocols:

1 — XMODEM or MODEM7 protocol
2 — CompuServe 'B' protocol
3 — CompuServe 'A' protocol
4 — DC2/DC4 capture protocol

Select #4 to match Orchestra-90 CC's protocol. After you select this transfer protocol, the system uses it until you exit the database, so the above menu is shown only once.

CompuServe automatically opens your terminal buffer and transmits the file to your system. The background on your screen turns red during file transmission. Upon reaching the end of the file, CompuServe closes your terminal buffer. At the end of transmission, CompuServe sends the signal to close your buffer. The background on the screen returns to dark green, indicating that you are still in Transfer Mode.

Do not log off the CompuServe system with the downloaded music file in memory. You might be tempted to log off and save on the connection time once the buffer closes. However, when the log off command is executed, the "disconnect" usually generates a few characters. These characters are of no consequence **unless** one of them happens to be the control character that opens the Orchestra-90 CC buffer. If that character transmits during the disconnect sequence, Orchestra-90 CC automatically opens its buffer, clears the buffer's previous contents (the music file you just downloaded), and replaces the contents with the meaningless characters.

One way of avoiding this situation is to save the file before you log off. Instructions for doing so follow. Another solution is to initiate the log off command and **immediately** press (BREAK) to return to Command Mode. This takes Orchestra-90 CC out of Transfer Mode and moves the file from the buffer to the command area, in which it is safe. You can then play or save the file.

Playing the File

Press (BREAK) to return to Command Mode. The music file is visible and ready to be scored. To score the file, type:

S (ENTER)

To play the file, type:

P (ENTER)

Unless you logged off CompuServe, you are still connected to the system. If you like the file, you can save it to tape or diskette, or you can type X (ENTER) to return to Transfer Mode and continue your session. Remember that any new file you download replaces the current file in memory.

Saving the File to Tape or Diskette

If you are unsure about whether Orchestra-90 CC is in Cassette or Disk Mode, simply type ? (ENTER) at the Command Line. The Status Line displays C or D.

Type C (ENTER) to set Cassette I/O Mode, or type D (ENTER) to set Disk I/O Mode. The command:

W *filename* (ENTER)

Writes the music file to cassette tape or diskette, depending upon the mode you selected. In either mode, the filename can be a maximum of eight characters in length.

After saving the music file, you can type X (ENTER) to return to Transfer Mode and finish your session with CompuServe.

Uploading Files to CompuServe

Log on to CompuServe, and access the Orchestra-90 CC SIG as previously explained. Type XA2 (ENTER) to access the music file data base.

Press (BREAK) to return to Command Mode, and load your music file into memory from cassette or diskette using the command:

R *filename* (ENTER)

This step is not necessary if the music file was in memory before you accessed CompuServe.

Optimizing the File

To reduce the uploading and downloading times as well as the storage size of music files, Orchestra-90 CC allows the music file in memory to be optimized. This is mandatory for file submission to the Orchestra-90 CC SIG. Optimizing removes all unnecessary spaces and measure strings. This can drastically reduce the size of the file. To optimize a file, type:

O (ENTER)

You can still play and edit the file normally.

Type X (ENTER) to return to Transfer mode.

Type UPL *filename.*A85 (ENTER)

Press (ENTER) one extra time to indicate the end of the description.

Next displayed is ENTER PROMPT CHARACTER TO USE. CompuServe accepts the next key pressed as the prompt character. Do not press (ENTER) after you type the character. You can use any character you like, for example, :. Wait a few seconds after typing the desired character.

CompuServe responds by sending the first : and waiting for you to start transmitting your file in memory.

To start transmitting the file in memory, press (SHIFT) (BREAK) . Orchestra-90 CC now asks, PROMPT? Press the same character key that you previously selected—in this example, :. Orchestra-90 CC automatically starts transmitting the music file. The background of the

screen turns red during file transmission. Upon reaching the end of the file, the background returns to dark green. Now, indicate to CompuServe that you are finished transmitting. Press (↑) Z to indicate the end of data.

CompuServe responds by asking for KEYWORDS. These words are used later to search the data base. For example, you might respond with:

```
BACH, FUGUE, 4VOICE, CLASSICAL (ENTER)
```

Next, CompuServe asks for a DESCRIPTION. You can use as many as 500 words to describe your music file. A typical description might be:

```
Bach's Fugue #10, arranged on Orchestra-90
/CC in 4 voices by Bryan Eggers (ENTER)
```

Press (ENTER) one extra time to indicate the end of the description.

Your file transfer is merged into the database by the SYSOP as soon as possible.

With CompuServe, the ORCH-90 SIG lets you set a Line Length parameter as a default option. By setting your Line Length to 32, the SIG formats all transmitted text so that no words are "split" at the end of a line. Use the following command at the "Function:" prompt to set your Line Length at 32:

```
OP;LL;32;P;T (ENTER)
```

This command has no effect on the music files you download.

APPENDICES

Command Summary A

Edit Mode

Press E (ENTER) or (SHIFT) (BREAK) to enter edit mode.

Control Key	Function
BREAK	Return to Command Mode.
CLEAR	Erase to end of line.
SHIFT + CLEAR	Delete current line.
ENTER	Accept the current line.
←	Move cursor left.
→	Move cursor right.
SHIFT + ←	Delete character.
SHIFT + →	Insert space.
↑	Move cursor up.
↓	Move cursor down.
SHIFT + ↑	Join line.
SHIFT + ↓	New line.
?	Duplicate line.
!	Move line up.

Command Mode

Command	Operand	Function
/	**<any string>**	Search forward for next occurrence of string (auto repeat).
–	**<any string>**	Search backward for next occurrence of string (auto repeat).
APPEND	***filename***	Append the named file into memory at the beginning of the current file.
BOTTOM	(none)	Position cursor at the end of the file.
CASSETTE	(none)	Set I/O mode to Cassette.
DIR	**0, 1, 2,** or **3**	List all Orchestra-90/CC (or "music") files on the diskette named in the operand. Default = 0. Or, set I/O mode to Diskette.
EDIT	(none)	Enter Edit Mode.
GET	***filename...***	Read, score, and play each of the files named in the operand.
KILL	***filename***	Delete the named file from the diskette.
LIST	(none)	Print the current file on the printer.
MULTI	***filename ...***	Perpetual GET.
NEW	(none)	Erase current file.
OPTIMIZE	(none)	Delete all unnecessary measure strings, spaces, and blank lines.
PLAY	**Part number** or **blank**	Play the current piece starting with the part number in the operand or at the beginning if no operand.
READ	***filename***	Read the named file into memory.

Command	Operand	Function
SCORE	(none)	Compile the current file.
TOP	(none)	Position the cursor at the beginning of the file.
WRITE	***filename***	Write the current file with the name in the operand.
XFER	(none)	Enter Transfer Mode.
reset	(none)	Reset button. Break a printer lockup or cassette I/O lockup.

Time Value Modifier	Musical Equivalent	Name
.		dotted note
:		triplet

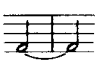
Symbol	Modifier	Definition
(	(none)	Begin reiteration.
)	**hex digit**	Reiterate the number of times specified by the modifier.
*	(none)	Unless otherwise indicated, all the notes following are assumed to be + or treble clef.
/	(none)	Comments. The remainder of the line is ignored.
<	**hex digit**	All the notes in the current and following measures are transposed down the number of semi-tones specified by the modifier.
=	**2-digit hex**	Set tempo to value of modifier.
>	**hex digit**	All the notes in the current and following measures are transposed up the number of semi-tones specified by the modifier.
@	(none)	Unless otherwise indicated, all the notes following are assumed to be - or bass clef.
←	(none)	All the unsigned notes following are assumed to be percussion clef. This is the left arrow produced by pressing (SHIFT) 0
J	**register name, R or S, nine hex digits**	Define the named register with the named wave form type, description, and volume.
K	**digit 0-7** and # or &	Define key signature by number and type specified by modifier.
M	**any string** and a **space**	Define the beginning of a measure. The current measure is ended. Accidentals are dropped. The key signature is restored.
N	**H, Q, I, S,** or **T**	Define time signature, set note type in modifier to one beat.

Symbol	Modifier	Definition
O	**hex digit**	Set the options to the value of the modifier.
P	**2-digit hex**	Define the beginning of the part named by the modifier. The current measure is ended. The current part is ended.
R	**2-digit hex**	Repeat the part named by the modifier. The named part must be previously defined. This symbol group can be followed by a tempo group and/or register group. A measure following a repeat must be defined by a new part number.
U	**signed hex**	Transpose all the notes following that belong to the current voice up or down the number of whole steps indicated by the modifier.
V	**1, 2, 3, 4,** or **5**	All the notes following are added to the previous notes of this measure belonging to the voice named by the modifier.
Y	**A, B, C, D,** or **E**	Set the current voice in the current part to the register specified by the modifier.
Z	**hex digit**	Map the stereo channels as specified by the modifier.

Note Modifier	Musical Example	Name
#	♯	accidental sharp
&	♭	accidental flat
%	♮	accidental natural
##	𝄪	double sharp
&&	𝄫	double flat
%#	♮♯	natural sharp
%&	♮♭	natural flat

Expression Modifier	Example	Name
, (Comma)		intense staccato
;		staccato
' (Apostrophe)	(none)	short portato
"	(none)	long portato

Time Value Symbol	Musical Equivalent	Name of Note	Rest ($)
W		whole note	
H		half note	
Q		quarter note	
I		eighth note	
S		sixteenth note	
T		thirty-second note	
X		sixty-fourth note	

Note Signs	
Dotted notes	A dot after a note adds 1/2 the value of the preceding note. Examples in 4/4 time: dotted quarter note 1-1/2 beats dotted half note 3 beats
Triplet notes 	Three notes of equal time value played in the time of two of the same kinds of notes.
Hold	Notes sound longer than their actual values.
Tied notes 	The first of two tied notes must have no articulation. The two notes sound as one continuous note. An accidental on the first note is passed to the note in the next measure.
Staccato 	Expression modifier used to make a note sound separate or detached from the other notes. Use a comma or a semicolon after the note symbol to produce this effect.
Articulation	Expression modifier used to make a note sound clear and distinct, especially within a series of similar notes. Use an apostrophe or quotation mark after the note symbol.

Performance Signs		
Repeat signs	=	
First and second endings	=	
Repeat preceding measure	=	
Common time	=	 
Cut time	=	 
Gradually slower	=	*rit.*
Finish	=	*Fine*
Repeat from the beginning to the end of the measure marked *Fine*	=	*D.C. al Fine*
Repeat from the sign 𝄋 to the end of the measure marked *Fine*	=	*D.S. al Fine*
Repeat from the beginning to the coda sign ⊕ and then skip to the coda	=	*D.C. al ⊕Coda*
Repeat from the sign 𝄋 to the coda sign ⊕ and then skip to the coda	=	*D.S. al ⊕Coda*
Repeat from this sign	=	𝄋
Coda mark when playing second time after *D.C.* skip from this sign to the coda	=	⊕

Error Messages B

When an error is detected, an error number is displayed. Whenever possible, the cursor is positioned at or near the location in the file at which the error was encountered.

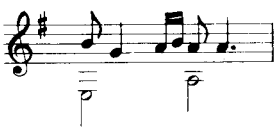
Error Message	Meaning
Err 1: MEMORY OVERFLOW	There is not enough memory to score the current file, even after overlapping. There is not enough memory to edit the current file. Consider expanding your system or splitting the file into two or more smaller sections. There is not enough memory to read or get the requested file. Expand the memory in your system.
Err 2: SYMBOL OUT OF CONTEXT	The command operand contains an invalid hexadecimal digit. The tempo symbol, =, is not followed by two hexadecimal digits. The compiler was expecting a hexadecimal digit or a note symbol (0-9 or A-G) but none were found.
Err 3: PARAMETER ERROR	The number of sharps or flats in the key signature is too large or is not followed by a sharp (#) or flat (&) symbol. The note value in the time signature is not **H**, **Q**, **I**, **S**, or **T**. The voice specified is not **1**, **2**, **3**, **4**, or **5**. The register specified is not **A**, **B**, **C**, **D**, or **E**.

Error Message	Meaning
Err 4: **INVALID PART NUMBER**	The part or repeat specified is not a two-digit hexadecimal number or the part is not blank. **P00** is considered an unnumbered part. The part is already defined. The repeat is not previously defined. The piece cannot be played because the part cannot be found or because the piece needs to be scored.
Err 5: **MEASURE OVERFLOW**	The measure being compiled contains more than 32 notes per voice. Split the measure in half and carry forward any accidentals. **Note:** Staccato and portato articulations generate a note and a rest, so count them as two notes.
Err 6: **NOTE OVERFLOW**	Dotted whole notes, 7-dotted triplets, and other unusual combinations cannot be compiled.
Err 7: **FILE I/O ERROR**	The host operating system reported an **I/O** or other error during the last file operation. **READ** or **GET**: File not found. **WRITE**: Disk or directory full. **READ, WRITE, GET,** or **DIR**: Unrecoverable **I/O** error. **DIR**: Invalid drive number.

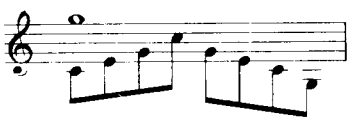
Error Message	Meaning
OVERLAP?	If there is not enough memory to hold both the source file and the compiled object code at the same time, the program pauses and displays the question, OVERLAP?, in the Status Line. If you answer **Y**, compilation resumes, and portions of the source file are overlapped by the object code. Answering anything else cancels the compilation and enters Command Mode. This function is useful when compiling large files, as the end of the object code can overlap the beginning of the source code. **Do not overlap a file that has not been saved on diskette or tape because this function implies the probable destruction of the current file.** Do not expect to be able to use the file in any way once this command is given.

Miscellaneous Programming Examples C

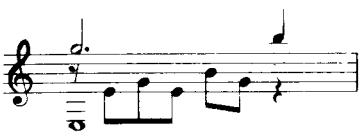
The following music samples are included to further educate you in the art of converting sheet music to Orchestra-90 CC programmed music. The key signatures are not included in the code given. The samples are in 4/4 time.



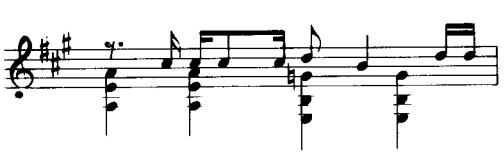
***I6Q4S5'6'I5'Q.5**
@V2H52



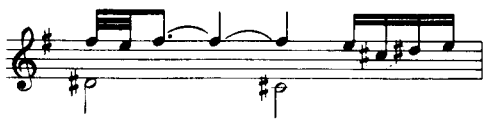
***WB**
V2I0247420-3



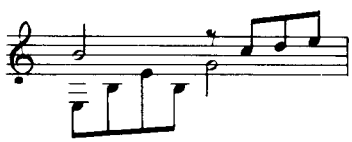
***H.BQD**
V2I24264
V3W-5



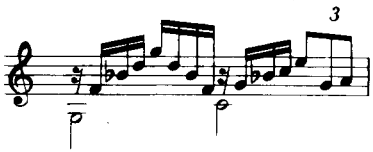
***I.$S7'7'I7'S7'I8Q6S8'8'**
V2Q5'54%4
V3Q2'2-1'1
@V4Q2'25'5



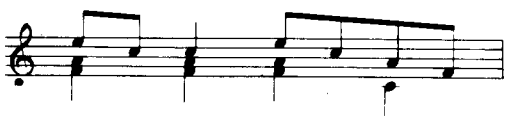
***TA'9I.AQAAS9'7#'8#'9'**
V2H1#0#



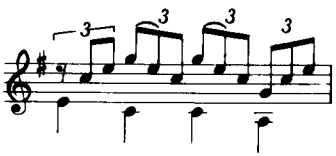
***H6I$789**
V2I-5-12-1H4



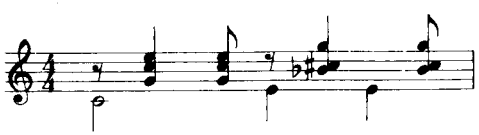
***S$3'6&'8'B'8'6'3'$4'6&'7'**
I:945
V2H-30



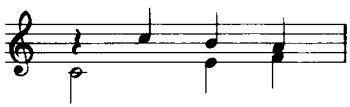
***I97'Q7I9753**
V2Q5'5'5'$
V3Q3'3'3'0



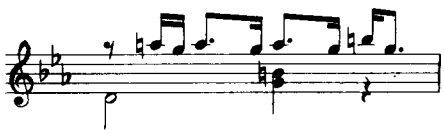
***I:$7'9'B9'7'B9'7'4'7'9'**
V2Q20'0-2



```
*I$Q9'I9$QB'IB
V2I$Q7'I7$Q7#'I7
V3I$Q4'I4$Q6&I6
V4H0Q2'2
```



```
*Q$765
V2H0Q23
```



```
*I$SC%'BI.CSB'I.CSB'D
%'I.B
V2H$Q6%
V3H1Q4
```

Percussion Samples

The following sample voicing lines demonstrate some of Orchestra-90 CC's possible percussion effects. When arranging a piece, you would enter one of these voicing lines for each voice you wanted as a percussion instrument. The instrument registration line is entered at the very beginning of the music file. All the following examples are labeled for Voice 4 (V4) but could be applied to any voice number. (The following is not a song but individual program lines used to demonstrate the various effects available with just one instrument register. Experiment with the other four instrument registers to see the effects that you can invent on your own.)

JES80011001E	(defines Register E, sine wave, Volume E)
V4YE	(assigns Register E to Voice 4)
NQ = 60	(plays the samples at this tempo)

4/4 Time Signature

← V4SA"B"C"D"ID"8"QD"IB"A"
← V4QD"E"8"IE"E"
← V4QC"9"IF"D"E"9"
← V4IE"SF"C"ID"E"SF"C"IF"D"SF"C"
← V4QC"F"I8"9"F"8"E"
← V4QC"I8"9"F"8"E"
← V4QC"9"D"IA"E"
← V4QE"I9"F"Q7"I8"9"
← V4QB"IA"8"E"8"6"C"
← V4Q8"I8"8"9"9"9"
← V4Q8"I9"D"8"D"A"
← V4Q8"SA"A"I9"F"SF"8"I8"E"SA"C"

3/4 Time Signature

← V4I8"8"QC"ID"E"
← V4IE"SE"B"I8"B"QD"
← V4(ID"SE"E"ID")1
← V4S6"7"8"9"IE"8"E"8"
← V4S:(E"F"F")3IF"F"
← V4I8"8"B"S:C"D"E"ID"E"

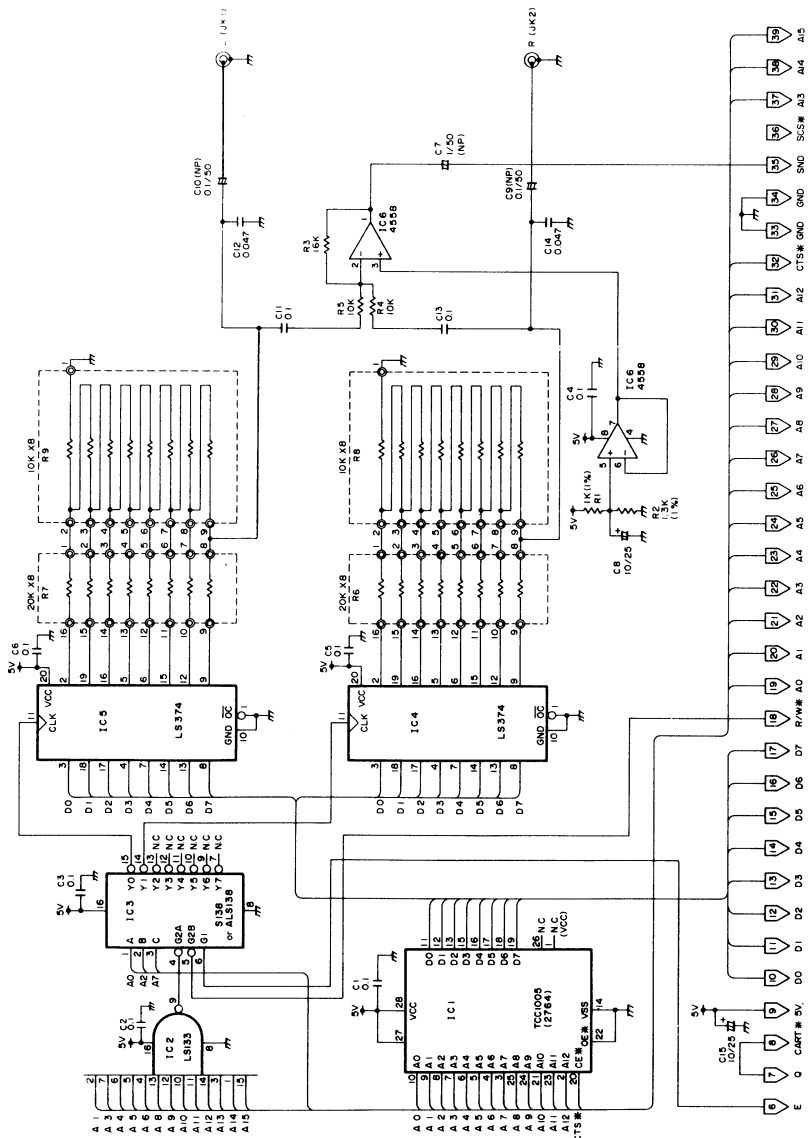
Additional Instrument Register Definitions D

These are the register definitions for Orchestra-90 CC's interpretation of some instrumental sounds. Keep in mind that the sounds produced by these register definitions will not necessarily sound similar to the instrument. Experiment with these sounds to find ones that fit your needs.

JASA2840000F	/	CHIMNEY FLUTE
JASE2020000F	/	FOREST FLUTE
JASCEEA0402F	/	MELOPHONE
JAS8E480402F	/	SOLO OPEN FLUTE
JAS46802200F	/	VIENNA FLUTE
JAS48EA6402F	/	GRAND VIOLIN
JAS68C60202F	/	'CELLO
JAS48A20206F	/	VIOLA D' AMORE CELESTE
JASACE20402F	/	VIOLIN
JAS82C44206F	/	VIOLINCELLO
JASC0A22202F	/	BASSOON
JASACEF8604F	/	BUGLE
JAS0A0A0A08F	/	CLAIRION HARMONIQUE
JAS20286402F	/	DULZIAN
JAS262A260CF	/	EGYPTIAN BAZU
JAS84E82408F	/	ENGLISH HORN
JASEACC0000F	/	FRENCH HORN
JAS02A20806F	/	ORIENTAL REED
JAS42048A06F	/	ROHRSCHALMEI
JASA6220000F	/	SACKBUT
JASFC600402F	/	SAXOPHONE
JASFF8FFF0FF	/	SOLO TUBA
JASCFCE6000F	/	STENTOR HORN
JASEFFEA602F	/	TROMBONE
JASCEE222E1F	/	WALDHORN
JAS26ACCC04F	/	BANJO
JAS40060004F	/	BELLS
JAS26020202F	/	CELESTA HARP
JAS2C020002F	/	CHIMES
JAS02000002F	/	GLOCKENSPIEL
JASA4022240F	/	HARP
JAS600C0000F	/	MARIMBA
JAS0A0A0E0EF	/	ZINK

JAS20022202F / VOX HUMANA
JAS26E06402F / SARRUSOPHONE
JASFFFFF0FF / OPHECLEIDE
JASFCA00000F / SOLOSAXOPHONE
JASC0C0A202F / MELODY CLARINET
JAS0E080400F / CORNO OCTAVE
JAS20040008F / MINIATURE BELLS
JAS28222202F / ETHEREAL VIOLINS
JASA8642202F / PIANO

Schematic Diagram E



NOTE (1) ALL RESISTANCE VALUES ARE INDICATED IN "OHM" (K=10^3 OHM, M=10^6 OHM)
(2) ALL CAPACITANCE VALUES ARE INDICATED IN "µF" (P=10^{-6} µF)

Schematic subject to change without notice. For most accurate Schematic (and parts) contact Radio Shack, National Parts Dept., Fort Worth, TX 76101

In UK, contact Tandy Electronics, National Parts Dept., Bilston Road, Wednesbury, West Midlands WS10 7JN.

In Australia, contact Tandy Australia Limited, National Parts Dept.; 91 Kurrajong Avenue, Mount Druitt, N.S.W.2770.

SERVICE POLICY

Radio Shack's nationwide network of service facilities provides quick, convenient, and reliable repair services for all of Tandy 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 Tandy computer products are broken, Radio Shack reserves the rights to refuse to service the equipment or to void any remaining warranty on the equipment.
2. If any Tandy 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 Tandy computer equipment to original manufacturer's specifications will be charged to the customer in addition to the normal repair charge.

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 AVENUE MOUNT DRUITT, N.S.W. 2770	PARC INDUSTRIEL 5140 NANINNE (NAMUR)	BILSTON ROAD WEDNESBURY WEST MIDLANDS WS10 7JN

2A5

Printed in Korea
811015410A