ROBOT CONTROL LANGUAGE(tm) WITH SAVVY(R)

PROGRAMMER'S MANUAL

GENERAL ROBOTICS CORPORATION Golden, Colorado

1994

Copyright 1986 General Robotics Corporation
All Rights Reserved

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in a form or by any means, electronic, mechanical, photocopy, recording, or otherwise, without the prior written permission of the publisher.

GENERAL ROBOTICS CORPORATION
14618 W. 6th Avenue, Suite 150
Golden, CO 80401
(303) 277-1574 or 277-1575
(800) 422-4265

The following are registered trademarks of the companies indicated:

RB5X: GENERAL ROBOTICS CORPORATION
Robot Control Language and RCL: GENERAL ROBOTICS CORPORATION
Savvy: Excalibur Technologies Corporation
Tiny BASIC: National Semiconductor Corporation
Apple II+, Apple IIe, and Apple DOS: Apple Computer, Inc.
C/PM: Digital Research

TABLE OF CONTENTS

Section I

A - Introduction.....A-1

- What you get when you order RCL with Savvy.....A-1
- What you need to get started.....A-2
- RB5X Options.....A-3
- What exactly is RCL with Savvy.....A-3
- Some useful Definitions.....A-5
- Savvy Definitions.....A-8
- Some Keyboard Preliminaries.....A-9

B - Getting Started.....B-1

- To Configure RCL for your System.....B-1
- Backing Up your Diskettes.....B-7

C - Sample Robot Task.....C-1

D - Appendices.....D-1

Appendix 1: RCL Robot Task "Party" and its Tiny Basic Translation

Appendix 2: Primary Tasks used to compose Robot Task "Party"

Appendix 3: Provided RCL Tasks

Appendix 4: Your own Applications

Section II

A - Overview.....A-1

B - Structure.....B-1

Action Commands.....B-1
Programming Commands.....B-1
Parameter Passing Functions.....B-1
Error Checking and Compile Routines.....B-1
Vocabulary Elements on Robot Tasks.....B-2
RCL and its Relationship to Tiny BASIC.....B-3
Compiling RCL into Tiny BASIC.....B-5

C - Implementation.....C-1

Instructions for Keying in the Sample Robot Task.....C-1
Writing New Robot Tasks.....C-3
Extending RCL and Defining New Actions.....C-5
Modifying Existing RCL Tasks.....C-6

D - Debugging and Error Trapping.....D-1

E - Miscellaneous Programmers Notes.....E-1

F - Appendices.....F-1

Appendix 1: Provided RCL Tasks

Appendix 2: Detail Provided RCL Tasks

Figures

1. Hardware and Language Relationships.....B-3

2. Language Structures and Relationships.....B-4

GETTING STARTED WITH IBM RCL

To Use IBM RCL you need and IBM PC, XT, AT or true compatible, DOS 2.0 or greater, 128K of RAM and at least one double-sided disk drive. You can also use IBM RCL on two floppy disks or a hard disk.

To start using RCL with a floppy disk, boot DOS first, then place your RCL disk in drive A* and type RCL.

When the computer displays the question "What Would You Like Me To Do Now?", that is your ready message. You are ready to write Robot programs. Turn to page A-3 in your RCL Tutorial. IBM users should ignore section B or the Tutorial. All other sections of the RCL Manual are applicable.

* Be sure you have made copies of your RCL disk following the disk copying instructions for your computer before you begin programming.

TROUBLE SHOOTING

If the computer does not seem to be communicating with the robot when downloading a program:

1. Check the connections on the RS232 cable to be sure both ends are securely plugged in. Sometimes static can occur on the connecter pins. Plugging and unplugging the cable a few times will take care of this.

2. Check to be sure the robot is operating properly and has sufficient charge. Plug a PROM in and see if the robot does what it is supposed to.

3. Check to see that you have included all of the necessary commands in your RCL program and see that they are in the correct order.

TROUBLE SHOOTING

BOTH THE RB5X AND SAVVY ARE RELATIVELY TROUBLE-FREE AND EASY TO USE. BUT, AS WITH ALL TECHNOLOGY, EACH ITEM HAS ITS OWN PECULIARITIES. IF YOU ARE HAVING TROUBLE GETTING YOUR COMPUTER TO TALK TO YOUR ROBOT, HERE IS A LIST OF THINGS TO CHECK.

1- HAVE YOU READ YOUR SAVVY TUTORIAL AND RCL PROGRAMMING MANUALS? MANY PEOPLE DO NOT READ THESE MANUALS BECAUSE THEY HAVE CONSIDERABLE PROGRAMMING EXPERIENCE. SAVVY IS INCREDIBLY EASY TO USE BUT IT IS DIFFERENT FROM OTHER PROGRAMMING LANGUAGES AT CONCEPTUAL LEVEL. THERE ARE MANY THINGS YOU DO NOT DO IN SAVVY THAT ARE CRITICAL TO OTHER PROGRAMMING LANGUAGES. DOING THESE THINGS FROM HABIT OR INTENTIONALLY CAN CAUSE SAVVY PROGRAMS TO ABORT OR CAN EVEN DESTROY YOUR DATA DISKS.

2- FOR APPLE SAVVY, IS YOUR SAVVY CARD WELL SEATED IN SLOT 7 OF YOUR COMPUTER?

3- IS YOUR SAVVY SYSTEM CONFIGURED FOR THE COMPUTER HARDWARE YOU HAVE? (SEE YOUR RCL MANUAL.) IF SAVVY COMES UP ON YOUR COMPUTER SCREEN WITH THE PROMPT "WHAT WOULD YOU LIKE ME TO DO NOW?" AND YOU CAN SUMMON THE STANDBY TASKS AS OUTLINED IN THE RCL MANUAL, YOUR SAVVY SYSTEM IS ALRIGHT.

4- IF YOUR SAVVY SYSTEM TELLS YOU THAT A DISK MAY BE IN TROUBLE, IT IS RIGHT 99% OF THE TIME. MAKE A NEW COPY OF THE DATA DISK FROM YOUR FACTORY-PROVIDED SOURCE DISKS AND START PROGRAMMING OVER AGAIN.

5- DOES THE RB5X WORK WHEN YOU PLUG IN THE TEST UTILITY PROM? IF SO, YOUR ROBOT IS FINE. IF NOT, CHECK TO SEE IF YOUR ROBOT'S BATTERIES HAVE SUFFICIENT CHARGE. FOR ANY OTHER PROBLEMS SEE YOUR RB5X REFERENCE MANUAL.

6- IS YOUR SUPER SERIAL CARD WELL SEATED IN SLOT 1 OF YOUR COMPUTER? MOST COMMUNICATIONS PROBLEMS BETWEEN THE COMPUTER AND THE ROBOT ARE CAUSED BY PROBLEMS WITH THE SUPER SERIAL CARD INTERFACE.

7- WHEN THE COMPUTER BEGINS TO "LOAD" THE PROGRAM INTO THE ROBOT, THE ROBOT'S FLASHING LED LIGHTS SHOULD COME ON. IF THEY DO NOT, THE ROBOT IS NOT HEARING THE COMPUTER FOR ANY OF THE REASONS GIVEN IN THIS TROUBLE SHOOTING SECTION, PROVIDED THAT YOUR PROGRAM IS CORRECTLY WRITTEN.

8- ARE THE SWITCHES ON YOUR SUPER SERIAL CARD SET FOR DATA TRANSMISSION AT 1200 BAUD? (SEE YOUR SUPER SERIAL CARD MANUAL). SOMETIMES THE SWITCHES GET DIRTY. SLIDING EACH SWITCH BACK AND FORTH A FEW TIMES CAN CORRECT THIS PROBLEM.

9- IS THE CABLE THAT CONNECTS THE SUPER SERIAL CARD WITH THE ROBOT PLUGGED IN CORRECTLY? THE PLUGS ARE CONSTRUCTED SO THEY CANNOT BE PLUGGED IN WRONG BUT ALL PINS MUST MAKE CONTACT AT BOTH ENDS. THE CABLE MUST BE PLUGGED INTO PORT 1 OF THE ROBOT. SOMETIMES STATIC BUILDS UP ON THE PINS. PLUGGING AND UNPLUGGING BOTH ENDS OF THE CABLE CAN CORRECT THIS PROBLEM

10-THERE ARE TIMES THAT YOU MAY HAVE ALL OF THE SERIAL CARD PROBLEMS AT ONCE. BE SURE TO TRY ALL OF THE REMEDIES SEVERAL TIMES IF ONE DOES NOT CORRECT THE PROBLEM. ABOUT 95% OF ALL COMMUNICATIONS PROBLEMS OCCUR IN THIS AREA AND DO NOT INVOLVE THE ROBOT OR THE SAVVY SYSTEM DIRECTLY.

11-DO YOU HAVE AN E-PROM PLUGGED INTO THE ROBOT? YOU CANNOT LOAD A PROGRAM FROM THE COMPUTER INTO THE ROBOT IF AN E-PROM IS PLUGGED IN.

12-IF YOU HAVE TRIED ALL OF THE ABOVE WITH NO SUCCESS, CALL THE FACTORY. IN THE HISTORY OF THE RB5X THERE HAS BEEN A LESS THAN 1% TECHNICAL FAILURE RATE OF ANY COMPONENT, BUT SOME ASPECT OF YOUR SYSTEM MAY FALL IN THAT RANGE. WE GUARANTEE ALL OF OUR PRODUCTS AND WE WILL FIND THE PROBLEM AND CORRECT IT.

ROBOT CONTROL LANGUAGE(tm) WITH SAVVY(R)

TUTORIAL

GENERAL ROBOTICS CORPORATION Golden, Colorado

1986

ROBOT CONTROL LANGUAGE WITH SAVVY
TUTORIAL

TABLE OF CONTENTS

A. Introduction A-1

What You Get When You Order RCL With Savvy A-1
What You Need to Get Started A-2
RB5X Options A-3
What Exactly is RCL with Savvy ? A-3
Some Useful Definitions A-5
Savvy Definitions A-8
Some Keyboard Preliminaries A-9

B. Getting Started B-1

To Configure RCL for Your System B-1
Backing Up Your Diskettes B-7

C. Sample Robot Task C-1

D. Appendices D-1

Appendix 1: RCL Robot Task "PARTY" and its Tiny BASIC Translation

Appendix 2: Primary Tasks Used to Compose Robot Task "PARTY"

Appendix 3: Provided RCL Tasks

Appendix 4: Your Own Application

A. INTRODUCTION

Congratulations on your purchase of Robot Control Language(tm) with Savvy(R) - RCL(tm), for short. Developed jointly by Robot Corporation and Excalibur Technologies Corporation, RCL is more than just another programming language. RCL is a complete software development system that virtually eliminates the language barrier between robots and people by allowing you to program your robot using common English words and phrases.

This Tutorial introduces you to the fun of programming with RCL and explains some of the terms you need to know as you begin creating your own RCL programs. As with other Tutorials, we suggest you use this one while seated at your computer, with your RB5X nearby, so that you can do the examples as we go along. For details about RCL not covered in this Tutorial, see the RCL Programmer's Manual that comes with this package.

WHAT YOU GET WHEN YOU ORDER RCL WITH SAVVY

To start, let's look at what you get when you order the RCL package.

In addition to this Tutorial, you receive a Robot Control Language with Savvy Programmer's Manual and the Savvy Personal Language Reference Folio. (We will discuss the relationship of RCL to Savvy a little later in this Tutorial.)

We also provide the Savvy circuit card, which you need to use RCL and which fits into Slot 7 inside your Apple computer.

If you ordered RCL I, you received one RCL diskette, which contains RCL with Savvy and some initial robot tasks we've developed; a Savvy One Master diskette, which enables you to use Savvy alone for other purposes on your Apple; and a "Let's Use Savvy" diskette. With this system, you store the robot tasks you create on the RCL diskette; it is a good idea to make several copies of the diskette before you begin designing robot tasks so you will have additional diskette space for storing the robot tasks you develop later.

If you ordered RCL II, you received one RCL diskette, which contains RCL and the initial robot tasks; one Savvy Pro Master diskette, which you will use with the RCL diskette (or with a blank data disk, if you wish to use Savvy with your Apple for other purposes); a Savvy "Demo" diskette; and a "Let's Use Savvy" diskette.

Copyright 1984 All rights reserved.

The RCL I diskette or the Savvy Pro and RCL II diskettes should always be in the disk drives when RCL is in use. They are standard floppy diskettes; the Savvy manual tells you how to handle and care for them.

RCL I has everything for programming robot tasks that RCL II has except the Savvy "Suspend" and selective file transfer capabilities, which were excluded to provide more space in the one-diskette system. For more information on these capabilities, consult your Savvy Personal Language Reference Folio.

WHAT YOU NEED TO GET STARTED

RCL I

To set up RCL I, you need the following hardware items:

Most important, of course, is one RB5X robot, which comes with its own Reference Manual.

You also need an Apple II+ or Apple IIe computer with a single disk drive and controller.

To establish communication between your computer and the RB5X, you must have a serial communications card -- either Apple's Super Serial Card or California Computer Systems (CCS) 7710A.

You also need an RS-232 cable for loading the RCL programs you write using your Apple into the RB5X. The connector on one end should be a 25-pin D male connector. (If this sounds like Greek to you, ask your computer dealer for help in assuring that you have the right cable.) This goes into Port 1 on the back of your RB5X. The other end must match the RS-232 port on your Apple serial card.

Insert the CCS 7710A or the Super Serial card in Slot 1 in your Apple and set the card's baud rate at 1200. Slot 1 is commonly used for a printer; when using RCL, your computer communicates with RB5X as though it were a printer. Therefore, when loading robot tasks into RB5X you cannot also have a printer connected to your Apple.

When using the CCS 7710A card you must also make the following modifications to establish proper serial communication:

 Copyright 1984 All rights reserved.

1. Switch RS-232 lines 2 and 3 (inside the robot). This is done by positioning the switch on the RB5X's interface panel circuit board up, instead of down (normal).

2. Connect pin 4 on the CCS card to pin 6 on the CCS card.

3. Make sure there is no connection between pin 25 on the CCS card and the robot. Severe damage to the card and RB5X may result from this connection.

Please see your RB5X dealer if you need help making these cable modifications.

You may want to use an 80-column display card because it allows you to use both upper and lower case letters, making the screen easier to read. If you have an Apple IIe, any brand of card will do. If you have an Apple II+, you must use the Videx Videoterm card. This card should be inserted in Slot 3 of your Apple. Refer to pages 3-4 of your Savvy manual for further information on hardware set up.

RCL II

RCL II has the same hardware requirements as RCL I, with one difference. To use this system, you need from two to four disk drives or, optionally, a hard disk drive.

RB5X OPTIONS

You might want to add options -- such as the RB arm or voice/sound synthesis -- to your RB5X to make your programming in RCL more fun and more challenging. The sample program we use in this Tutorial, for example, works with an RB5X that has voice capability.

WHAT EXACTLY IS RCL WITH SAVVY?

RB5X's native tongue is Tiny BASIC. But as we all know, computer languages such as Tiny BASIC can be difficult for people to learn. There are complicated codes and symbols to memorize, and every character you enter must be in its proper place. No extra spaces, no transposed letters. Otherwise, it isn't Tiny BASIC and RB5X can't understand what you want it to do. And while Tiny BASIC is easier to learn than many programming languages, it can still be frustrating.

So RCL's primary responsibility is to act as a foreign language interpreter, translating what you say in your own words into the Tiny BASIC in which your RB5X is fluent.

Copyright 1984 All rights reserved.

RCL is based on Excalibur's personal language system called Savvy. So to understand RCL, you must have at least some understanding of Savvy.

Think of it this way: RCL is a subset of Savvy; it exists within Savvy, as your house exists within its neighborhood. RCL has its own personality, its own unique style of architecture. But it also draws some of its characteristics from its environment, from its neighborhood; in this case, from Savvy.

So what is Savvy? It really is a lot of things. It is an operating system -- you don't need CP/M or Apple DOS or any other operating system.

Savvy is a Virtual Resources Manager that lets your small microcomputer run huge programs as if it were a much larger machine.

Savvy is a language. More specifically, Savvy learns your language from you -- whether it is English, French, or Turkish. But since it needs some language to start with, Savvy learns an important primary vocabulary at the factory.

Savvy is an associative memory, which lets the system place items in its memory based on their description; therefore, the system will find what you want if you describe it. You don't have to assign or ask for an actual memory address, which is a complicated programming activity.

Savvy is an automatic program generator, which allows it to write "families" of programs for you, designing them for specific purposes.

But, most important, Savvy is a personality. You communicate with it using words instead of baffling, hard-to-remember codes.

Using Savvy as its basis, RCL then learns your language from you and offers RB5X owners a flexible, interactive, conversational method of writing programs for the robot.

In addition to learning your language from you, RCL also tries to figure out what you mean by what you enter at the keyboard. For example, if you enter the words MVOE FOREWARD on your computer keyboard exactly as we have written here, errors and all, RCL searches its memory trying to figure out what you really intended to enter. It then presents you with as many phrases as it has that resemble MVOE FOREWARD and asks you which one you want:

 Copyright 1984 All rights reserved.

```
I'm not sure what MVOE FOREWARD refers to.
Is it one of these:
   1. MOVE FORWARD
   2. MOVE BACKWARD
   3. MOVE RANDOMLY
   4. or None of the above?
Please type the number of the correct answer:
```

RCL also comes with some standard tasks that you can incorporate into the programs you write yourself for your RB5X. (MOVE FORWARD is one of those tasks.)

So RCL is an intelligent software development system -- able to draw conclusions, willing to overlook your spelling errors, and ready to learn your language from you -- that is just perfect for use with the RB5X Intelligent Robot!

SOME USEFUL DEFINITIONS

Before we actually step you through a sample program, there are some important definitions that will help you to use RCL.

Robot Tasks and Primary Tasks

In RCL, a "robot task" is a program that you write in order to train RB5X to do something -- for example, move through the living room using its sonar and tactile sensors to detect objects in its way.

A robot task is composed of smaller, primary tasks, some of which we have already written for you to use in the composition of your robot tasks and some of which you will create yourself. So unlike other languages where you have routines and subroutines as distinct from procedures and functions, in RCL, everything is a task. A robot task simply comprises a group of shorter tasks, one nested neatly inside the other.

A list of the provided RCL tasks appears at the end of this Tutorial, in Appendix 3. Take a look now at this list. You will notice that we have starred (*) some of these tasks. This designates the tasks with which everyone using RCL should be familiar, whether they are writing simple or complex robot tasks. The other, unstarred entries are tasks that will be used by more experienced users of RCL.

We use some of the starred entries in the "Sample Robot Task" later in this Tutorial.

Copyright 1984 All rights reserved.

Prompt

There are times when you are working in RCL that the system prompts you. This is a common programming term meaning that a message appears on the screen asking you for more information or instructing you on what to do next.

The RCL prompt you probably see most often is:

    What would you like me to do now?

as the system awaits your next set of instructions.

Menus

While you are working in RCL, you will notice that the system periodically displays lists of options, like a restaurant menu, from which you choose items for the system to accomplish. For example, if you use the Savvy SUMMON command, the system displays a menu and gives you several choices: you can either backup your diskettes, configure the system for your individual computer setup, or run diagnostics. You also have the choice of returning to the "What would you like me to do now?" prompt.

Section B, "Getting Started," contains several examples of RCL menus.

Replacement Symbols

You will notice as you look through the list of primary tasks that some of them contain angle brackets (< >) enclosing numbers. These numbers designate replacement symbols, characters that stand for information that you must supply.

For example, in the task "DEFINE A VARIABLE called <1>", the number 1 is the replacement symbol. You must supply the name of the variable -- for example, DEFINE A VARIABLE called "A". Note that when you supply the answer in place of the replacement symbol, unless the answer is a number (value) or unless it is a response to the direction prompt in the commands for the RB arm (OPEN, OUT, etc.), it must be enclosed in quotation marks. You type the first set of quotations (") and when you press <RETURN>, Savvy supplies the ending quotation marks.

Loops

Another concept you must understand is loops. If you have done any programming, you should recognize this term. Even if you haven't, you may have heard it and have a vague idea of what it means.

 Copyright 1984 All rights reserved.

In its most fundamental sense, a loop is a procedure that a computer program -- or, in this case, a robot task -- repeats over and over again until it meets some condition or set of circumstances, allowing the program to stop the repetition. In other words, a loop within a robot task instructs your RB5X to repeat a procedure over and over again until some condition is met that allows it to stop. And if you don't designate this condition, your RB5X repeats and repeats until you switch it off. Our "Sample Robot Task" later in this Tutorial includes several loops and illustrates this concept.

In RCL, there are three kinds of loops: limited loops, counted loops, and general loops.

The limited loop is identified in RCL as the "BEGIN A LIMITED LOOP" task, and is ended with the command "REPEAT THE LIMITED LOOP until <1> is (<,>,=,<>) <2> to <3>." In a limited loop (which Tiny BASIC programmers will recognize as a DO/UNTIL loop), the robot task repeats the same procedure over and over again until it encounters a condition that allows it to leave the loop. This condition is defined by the "REPEAT THE LIMITED LOOP" command.

In a counted loop (which Tiny BASIC programmers will recognize as a FOR/NEXT loop), the robot task repeats a procedure for a specified number of times. The task "BEGIN A COUNTED LOOP" instructs the robot task to start this procedure and begin counting the times it is repeated. For this task, your computer system prompts you for the name of the loop, where it begins, and where it ends. So the task reads "BEGIN A COUNTED LOOP called <1> beginning at <2> ending at <3>". Remember that 1, 2, and 3 are replacement symbols for information you must supply. The number 1 represents a variable name that will be used in Tiny BASIC, so it must be a single letter enclosed in quotation marks. The numbers 2 and 3 designate variable values that RCL counts from the first value to the second, one number at a time. For example, the loop that begins with "BEGIN A COUNTED LOOP called "P" beginning at 1 and ending at 5" and is closed with "REPEAT THIS COUNTED LOOP for the counter "P"" will be executed five times.

General (unconditional) loops (which Tiny BASIC programmers will recognize as a GOTO construction) will repeat from "BEGIN A LOOP" to "REPEAT THIS LOOP" until some condition within the loop causes the task to exit and proceed with the instructions that follow. The primary task "TEST IF the variable <1> is (=,<,>,<>) <2> compared to <3>" and the primary tasks that begin with EXIT are most often used to set up a condition within the loop.

Copyright 1984 All rights reserved.

SAVVY DEFINITIONS

Since RCL is a subset of Savvy, you occasionally need to use some commands that are basic Savvy functions and are not really additional features of RCL. We have listed some of these commands -- called primaries -- here, but you should examine your Savvy manual for details on using the Savvy primaries.

ASSOCIATE

This command enables you to teach RCL your language. Specifically, you can tell it to "associate" one phrase with another. For example, using this task, you can teach RCL to think of the two tasks "CLEAN THE CARPET" and "VACUUM THE FLOOR" as meaning exactly the same thing.

DEFINE

This is the command you use to create robot tasks in RCL. The "Sample Robot Task" beginning on page 21 begins with DEFINE.

DETAIL

This command allows the user to list the various components of a robot task. For example, if you look at the "BEGIN A LOOP" detail in Appendix 2, you see the four separate tasks that comprise "BEGIN A LOOP."

EDIT

This command allows you to change a robot task you already created using the DEFINE command. In using this command, you have access to several important functions, including L (which allows you to list the task you are editing), <#>--a line number (which enables you to automatically move the cursor to the line number specified), D (which deletes the line above your cursor), and Q (which allows you to leave, or quit the EDIT task and return to the mode where you can enter more robot tasks). The "Sample Robot Task" in Section C describes how to use these functions.

FORGET

This command cancels an association that you may have previously created using the ASSOCIATE command. For example, if you told the system to recognize "CLEAN THE CARPET" and "VACUUM THE FLOOR" as meaning the same thing, and later decide they should have two separate meanings, you would use this command.

 Copyright 1984 All rights reserved.

RENAME

This command allows the user to rename a task throughout the RCL system. This is most often used when customizing RCL to your particular needs.

SUMMON

The SUMMON command allows you to display on the screen the EXPLANATION of STANDBY TASKS menu (RCL II) or the UTILITIES menu (RCL I). From these menus, you can choose to make backups of your diskettes, configure RCL to match your computer system, or run a diagnostic check of your system.

SOME KEYBOARD PRELIMINARIES

In general, your Apple keyboard is similar to the keyboard of a standard typewriter, with a few extra keys added that correspond to specific computer functions.

Your Apple User's Manual contains detailed information about the operation of the computer itself, including the functions of the various keys. But it is important that you understand how RCL handles certain keys.

RESET Key

The RESET key interrupts the computer's CPU and leaves no way for you to return to what you were doing. As a rule, you should **NEVER** use the RESET key as it will cause irreparable damage to your RCL diskette or diskettes.

RETURN Key

Any time you type an instruction in RCL, you need to end it by pressing the RETURN key. This signals the system to look at what you have entered. In the sample program in this Tutorial, we use <RET> to remind you to press the RETURN key.

Left-Pointing Arrow

If you make a mistake while entering RCL instructions, use the left-pointing arrow to back up the cursor to the mistake. Then type your correction. Each time you press this key, the cursor backs up one character, erasing that character.

Right-Pointing Arrow

The right-pointing arrow acts as a "delete" key. When you press this key, you delete everything you have typed since you last pressed the RETURN key.

Copyright 1984 All rights reserved.

Escape Key

Use the Escape key, labeled ESC on the left side of your keyboard, to stop an operation. Pressing this key returns you to the "Finish, Resume, or Suspend?" prompt. Press the first letter of the choice you wish to make to continue from there. See your Savvy Personal Language Reference Folio for a detailed description of the "Finish, Resume, or Suspend?" function.

Upper-Case Letters

You should remember to type all of your RCL instructions in capital or upper-case letters. The system then responds or prompts you in all lower-case letters. (That is, unless you are working with an Apple II+ with a 40-column display. In this case, everything on the screen appears in upper-case letters.) The shift key does not work with Savvy; to shift between upper and lower case input press CTL and A at the same time.

 Copyright 1984 All rights reserved.

B. GETTING STARTED

One of the first things you should do when you open your RCL package is to configure the diskettes to your individual computer system and then make copies of the diskette or diskettes.

In RCL I, the one-diskette system, the system software takes up about 50% of the diskette space, leaving you only 50% for your own program development. So it is a good idea to make several copies of your RCL I diskette and to use these copies for writing different RCL programs for your robot.

With RCL II, the Savvy system software is contained on one diskette, leaving the second diskette for RCL commands and your program development. However, it is still a good idea to make copies of both the system diskette and the RCL diskette so that if your system ever malfunctions while your diskettes are in the drives, you can recreate the work you did up until the system failure.

TO CONFIGURE RCL FOR YOUR SYSTEM

RCL I

To configure RCL I for your system:

1. Make sure the Savvy circuit card is in Slot 7 inside your Apple and that the serial card is in Slot 1.

2. Insert your RCL diskette, with its label up and the notch to the left, into the Apple disk drive and close the door.

3. Switch the Apple on.

4. At the "What would you like me to do now?" prompt, use the Savvy SUMMON command to display the UTILITIES menu. Type:

   SUMMON<RET>

   The system then prompts you with the message:

   the standby task

Copyright 1984 All rights reserved.

5. Type:

"<RET>

and the system displays the following menu:

--UTILITIES MENU--

1. Make an exact copy of this diskette.

2. Perform system DIAGNOSTICS.

3. CONFIGURE for different hardware.

4. ALL DONE WITH HELP...Go back to 'What would you like me to do now?'

ENTER the NUMBER of the desired action:

6. Choose the CONFIGURE option by pressing:

3

There is no need to press the RETURN key.

The system then displays the following screen:

WHICH CRT (MONITOR) WILL YOU BE USING?
(ASTERISK SHOWS PRESENT CONFIGURATION)

*1) 40-COLUMN NATIVE APPLE SCREEN WITHOUT LOWER-CASE.

2) 40-COLUMN NATIVE APPLE SCREEN WITH LOWER CASE. (SPECIAL EQUIP. OR II-E REQ'D)

3) 80-COLUMN VIDEX VIDEOTERM. (SPECIAL EQUIP. REQ'D)

4) 80-COLUMN NATIVE APPLE SCREEN. (II-E REQ'D)

ENTER THE NUMBER FOR DESIRED CRT:

7. Press the number key that corresponds to your configuration. The system then displays the following menu:

 Copyright 1984 All rights reserved.

WHICH PRINTER INTERFACE WOULD YOU LIKE?
(ASTERISK SHOWS PRESENT CONFIGURATION)

1) APPLE PARALLEL

*2) APPLE SUPER SERIAL

3) CCS 7728 PARALLEL

4) CCS 7710 SERIAL

5) NO PRINTER PRESENT

ENTER THE NUMBER FOR DESIRED PRINTER.

8. Remember that you won't actually be using a printer. RCL simply thinks of the RB5X as a printer. Press the number key that corresponds to your printer setup (either #2 or #4). The system then returns you to the "What would you like me to do now?" prompt.

Once you have configured your diskette to match your system, you are ready to move on to the section, "Backing Up Your Diskettes."

RCL II

To configure RCL II for your system:

1. Make sure the Savvy circuit card is in Slot 7 inside your Apple and that the serial card is in Slot 1.

2. Insert the Savvy system diskette with its label up and the notch to the left into Disk Drive 1, and close the door.

3. Insert your RCL II diskette with its label up and the notch to the left into Disk Drive 2, and close the door.

4. Switch the Apple on.

5. At the "What would you like me to do now?" prompt, use the Savvy SUMMON command to display the EXPLANATION of STANDBY TASKS menu. Type:

   SUMMON<RET>

   The system prompts you with the message:

   the standby task

6. Type:

   "<RET>

Copyright 1984 All rights reserved.

and the system displays the following menu:

EXPLANATION of STANDBY TASKS

1. ARCHIVE is used to make copies of disks. It can also be used to do special disk related operations.

2. CONFIGURATOR is used to tell SAVVY what hardware is in your system and to make clean data disks.

3. DIAGNOSTICS is used to test the integrity of the data on your data disk. It can also clear the 'your disk may be in trouble' message.

4. All done with UTILITIES...Go back to 'What would you like me to do now?'

ENTER the NUMBER of desired action:

7. Choose the CONFIGURATOR option by pressing:

2

There is no need to press the RETURN key.

8. The system then displays the following menu:

-- CONFIGURATOR MENU --

1) Make a CLEAN new DATA-DISK.

2) Change the Hardware Configuration. (Monitors, Printers, Disk-Drives)

3) Extend the length of the DATA-DISK.

4) Shorten the length of the DATA-DISK.

5) All done with the CONFIGURATOR...go back to 'What would you like to do now?'

ENTER the NUMBER for the desired action:

Choose the second option by pressing:

2

There is no need to press the RETURN key.
The system then displays the following message:

 Copyright 1984 All rights reserved.

You may VIEW the CURRENT configuration of the hardware by responding with only a carriage RETURN to the questions.

HOWEVER, if you make any configuration changes, I will automatically RESTART to make your change become effective.

Press any key to continue.

9. Press any key to continue. The system then displays the following menu:

Which CRT (Monitor) will you be using?
(Asterisk shows present configuration.)

1) 40 column native APPLE screen without lower-case

*2) 40 column native APPLE screen with lower-case (Special equip. or II-e req'd.)

3) 80 column VIDEX videoterm (Special equip. req'd.)

4) 80 column native APPLE screen. (II-e req'd.)

ENTER the NUMBER for desired CRT:

10. Press the number key that corresponds to your system and the following menu appears:

Which printer interface would you like?
(Asterisk shows present configuration.)

1) APPLE parallel

*2) APPLE Super Serial

3) CCS 7728 parallel

4) CCS 7710 serial

5) No printer present

ENTER the NUMBER for desired PRINTER:

11. Remember that you won't actually be using a printer. RCL simply thinks of the RB5X as a printer. Press the number key that corresponds to your system (either #2 or #4) and the following menu appears:

Copyright 1984 All rights reserved.

What drive will you be using for DATA?
(Asterisk shows present configuration)

*1) Floppy diskettes

2) Corvus hard-disk

The MASTER-DISK you are now using is presently configured for a data-disk on Drive #2.

ENTER NUMBER for desired data-drive:

12. Press the number key that corresponds to your system and the following message appears:

    I always use diskette Drive #1 for the SAVVY MASTER diskette.

    How many of the REMAINING Drives shall I use for DATA storage (1-3)?

13. Press the number key that corresponds to your system and the following message appears to confirm your choice:

    For DATA I'll use Drive #X.

    Press any key to continue.

    where the X stands for the number 1 through 3 that you choose.

14. Press any key to continue and the following message appears, signaling you that configuration is complete.

    I will now RESTART to make these changes become effective immediately.

    If you changed your CRT configuration, the following message also appears:

    Reminder...you changed your video.
    You may need to move a cable.

The system then returns to the "What would you like me to do now?" prompt, and you may now move on to the section, "Backing Up Your Diskettes."

 Copyright 1984 All rights reserved.

BACKING UP YOUR DISKETTES

RCL I

Once you finished configuring your diskette to match your computer system, your screen displays the "What would you like me to do now?" prompt. To make backups of your diskettes:

1. Use the Savvy SUMMON command to display the UTILITIES menu again. Enter:

   SUMMON<RET>

   The system then prompts you with the message:

   the standby task

2. Type:

   "<RET>

   The system displays the UTILITIES menu:

   --UTILITIES MENU--

   1. Make an exact copy of this diskette.

   2. Perform system DIAGNOSTICS.

   3. CONFIGURE for different hardware.

   4. ALL DONE WITH HELP...Go back to 'What would you like me to do now?'

   ENTER the NUMBER of the desired action:

3. To choose the backup (or copy) option from this menu, press:

   1

   The system then displays the following message:

   This procedure allows you to make an exact copy of this ORIGINAL. You will need a spare diskette to continue.

   Do you still want to copy?

4. Type:

   Y

Copyright 1984 All rights reserved.

for yes and the system displays the following prompt:

Insert NEW diskette into drive

Press any key to continue.

5. Remove your RCL diskette and place your spare diskette into the disk drive. Close the disk drive door.

6. Press any key to begin the copying process. The system then displays the following message to signal you that copying is taking place:

   WORKING!

7. The next message you see is:

   Insert ORIGINAL diskette into drive.

   Press any key to continue.

   Remove your new diskette and place the original into the drive. Close the disk drive door. The system then displays the "WORKING!" message.

8. The system displays these last two messages several times before the copying process is complete. Continue to swap diskettes and press any key until you see the following message, which indicates that this process is complete:

   Good copy.

   The copying process is now complete.
   Insert the disk you want to use now.
   Press any key to continue.

9. Insert your copy into the disk drive and press any key to return to the "What would you like me to do now?" prompt.

10. To make additional copies, insert your original into the disk drive and repeat the procedure until you have several copies of your RCL diskette to work with. Keep the original in a safe place, using your copies to write robot tasks.

11. At the prompt, you are ready to proceed to the section of the Tutorial titled "Sample Robot Task" and to begin entering this task.

 Copyright 1984 All rights reserved.

Once you have finished configuring your diskettes to match your computer system, your screen displays the "What would you like me to do now?" prompt. To make backups of your diskettes:

1. Use the Savvy SUMMON command to display the EXPLANATION of STANDBY TASKS menu again. Enter:

   SUMMON<RET>

The system then prompts you with the message:

   the standby task

2. Type:

   "<RET>

   and the system displays the following menu:

   EXPLANATION of STANDBY TASKS

   1. ARCHIVE is used to make copies of disks. It can also be used to do special disk related operations.

   2. CONFIGURATOR is used to tell SAVVY what hardware is in your system and to make clean data disks.

   3. DIAGNOSTICS is used to test the integrity of the data on your data disk. It can also clearn the 'your disk may be in trouble' message.

   4. All done with UTILITIES...Go back to 'What would you like me to do now?"

   ENTER the NUMBER of desired action:

3. Choose option 1 by pressing:

   1

   There is no need to press the RETURN key.

   The system then displays the following menu:

Copyright 1984 All rights reserved.

-- ARCHIVE MENU --

1) Copy diskettes.

2) Show the diskette drive numbers.

3) Copy between hard-disk and diskettes.

4) Use the SPECIAL disk utilities and diagnostic routines.

5) All done with ARCHIVE...Go back to 'What would you like me to do now?'

ENTER the NUMBER of desired action:

4. Again, choose option 1 by pressing:

1

There is no need to press the RETURN key.

The system then displays the following screen:

COPY DISKETTES

This procedure allows you to copy data from one diskette, called the ORIGINAL, to another one, called the DUPLICATE.

Press ESCAPE (ESC) key to start over.

Do you need more instructions?

5. Answer this question by pressing:

Y

for YES and then follow the instructions as they are presented on the screen.

It usually takes about 90 seconds for the system to copy a diskette. As the copy is being made, the system also verifies the copy. If a copy cannot be verified, a message displays so that you can try again. When a successful copy is made, the screen displays the message:

Good copy

Press any key when ready to continue.

 Copyright 1984 All rights reserved.

6. After you press another key, you see the message:

```
REMOVE your new copy from Drive #1 then
RESTORE diskettes to original drives.

Press any key when ready to continue.
```

   In fact, you should use your new copies and preserve your original Savvy diskette and your original RCL diskette. So once you have placed your Savvy duplicate in Disk Drive 1 and your RCL duplicate in Disk Drive 2, press any key to continue. The system then displays the ARCHIVE menu.

7. Select option 5 from the ARCHIVE menu to return to the 'What would you like me to do now?' prompt. Press:

```
5
```

   There is no need to press the RETURN key.

You are now ready to move on to the next section, "Sample Robot Task."

Copyright 1984 All rights reserved.

C. SAMPLE ROBOT TASK

Now that we have discussed the basics of RCL, let us step through a sample robot task.

For this demonstration, we will assume that you have RCL I and an RB5X with the voice/sound synthesis option.

SCENARIO

> Assume you have invited friends to your home to enjoy snacks and conversation. As an added treat, you want your RB5X to greet each guest on arrival. As each one enters, you press a bumper on the robot, RB5X begins flashing its lights and rolls over to your friend, stops and introduces itself saying, "Hello, I am RB5X: the Intelligent Robot."

Before we actually write this robot task, you should consider the various elements that comprise it: specifically, movement, bumpers sensors, flashing lights, sonar, and voice. The robot task is then broken down into smaller tasks controlling these various elements.

A handy way to organize your approach to this scenario is to use the following format for laying out the logic or sequence of your robot task:

Copyright 1984 All rights reserved.

Building Blocks	Programming
Bumper sensor	The robot is programmed to respond to your pressing one of its bumper sensors by flashing its lights, moving, and using its sonar.
Flashing lights	At the touch signal from you, the robot begins to flash its lights.
Movement	At the same time, RB5X begins to move toward your friend.
Sonar sensor Bumper sensor	RB5X is programmed to stop moving as soon as it senses someone with its sonar or touches them with its bumper.
Voice synthesis	The robot greets your guest with the phrase, "Hello, I am RB5X: The Intelligent Robot" as soon as it stops moving.

Now, let's begin to compose our first robot task.

Before your friends arrive:

1. Make sure the Savvy circuit card is in Slot 7 inside your Apple and that the serial card is in Slot 1.

2. Make sure the voice card is in Slot J1 or J2 inside the robot.

3. Insert the RCL diskette, with its label up and the notch to the left, into the Apple disk drive and close the door.

4. Switch the Apple on.

5. When the "What would you like me to do now?" prompt appears, type:

   DEFINE<RET>

 Copyright 1984 All rights reserved.

6. The next CRT display should be:

   DEFINE a task called

   Let's call this robot task PARTY. So type:

   PARTY<RET>

   NOTE: If at any time what you expect to appear on the screen does not appear, press the ESCAPE key to return to the "Finish, Resume, or Suspend?" prompt. If you then press F (for Finish), you can return to the "What would you like me to do now?" prompt and begin again.

7. The next display should be:

   0--Task PARTY
   1 Does

   where line 0 gives the task title and line 1 is the beginning of what the robot task does. RCL automatically numbers the lines of your robot task.

8. The first thing you must do is prepare the robot's hardware to begin accepting commmands. You need to do this any time you want to write and then download programs from your computer to the RB5X. Type:

   PREPARE THE ROBOT<RET>

9. Next you must let the voice hardware know that it too must be ready to accept commands. Type:

   PREPARE THE VOICE<RET>

10. Next, you must load the command that instructs the robot to introduce itself to your guests. Type:
(SEE NEXT PAGE - EXPANDED NO. 10)
    X LOAD THE INTRODUCTION <RET>

    The X before this task indicates that it is a sample task that comes on your RCL diskette; you must include the X.

11. Line 4 contains the title of your task and must be included so that when the RB5X reaches the end of the task, the robot will have a point of reference for returning to the start and repeating the greeting again. Type:

    CALL<RET>

    The system answers with:

    this robot task line

Copyright 1984 All rights reserved.

Name this line by typing:

"BEGINNING OF PARTY<RET>

Don't forget the first pair of quotation marks. RCL enters the second set for you after you press the RETURN key.

12. Line 5 marks the beginning of the first loop in your robot task. With this general loop, you say to the robot, "You sit and wait patiently until I give you the signal to begin your greeting." Type:

    BEGIN A LOOP<RET>

13. Once you've instructed RB5X to begin a loop, you must also tell the robot how to leave (or exit) the loop. In this case, you want RB5X to sit and wait patiently until you press a bumper. So, type:

    EXIT IF ANY BUMPER TOUCHED<RET>

14. A general loop really consists of three tasks: one that tells it when to start (BEGIN A LOOP), one that tells it what to do within that loop or how to get out of the loop (EXIT IF ANY BUMPER TOUCHED), and one that closes the loop and tells it to repeat the procedure. To program this last task, type:

    REPEAT THIS LOOP<RET>

15. Now, when you touch any bumper on the robot, you want it to begin flashing its lights and to roll up to your guest, using its sonar and bumpers to tell when it has reached your friend. So now type:

    TURN ON THE FLASHING LIGHTS<RET>
    MOVE FORWARD<RET>

16. Here's a good place for another general loop. You want the robot to continue moving forward until either it senses something with its sonar or until it touches something with a bumper. So, type:

    BEGIN A LOOP<RET>

17. This loop has two tasks in it telling it when to stop. Type:

    EXIT IF ANY BUMPER TOUCHED<RET>
    EXIT IF SONAR<RET>

18. After you press the last RETURN, the system prompts you for the sonar distance you wish:

    distance value is less than

 Copyright 1984 All rights reserved.

10. Next, you must load the command that instructs the robot to introduce itself to the guests. Type:

    LOAD THE PHRASE <RET>

The computer screen will display:

    LOAD THE PHRASE called:

Now type in "SAY THE INTRODUCTION <RET>

The computer screen will display:

    LOAD THE PHRASE called "SAY THE INTRODUCTION" with phonemes:

Now type in the following phonemes exactly as they are shown including the quotation mark, and all periods. Do not worry about where the letters wrap around the screen. Just type in the the phonemes in a continuous string. If the phonemes are not entered in the correct manner, the robot will not speak.

    "H.H.EH.L.L.OO.O1.PA1.AH.E.E.PA1.AE.AE.M.PA1.D.THV.U.PA1.AH1.AH1.R.PA0.B.Y.Y.PA1.F.AH.Y.V.PA1.EH1.EH2.K.PA0.S.PA1.I.N.T.EH.L.L.UH3.D.J.EH3.N.T.PA1.R.O1.B.AH1.T.STOP. <RET>

The computer screen will display:

    LOAD THE PHRASE called "SAY THE INTRODUCTION" with phonemes: "H.H.EH.L.L.OO.O1.PA1.AH.E.E.PA1.AE.AE.M.PA1.D.THV.U.PA1.AH1.AH1.R.PA0.B.Y.Y.PA1.F.AH.Y.V.PA1.EH1.EH2.K.PA0.S.PA1.I.N.T.EH.L.L.UH3.D.J.EH3.N.T.PA1.R.O1.B.AH1.T.STOP."

This indicates at what distance the sonar is programmed to detect obstacles in the robot's path. In this case, you want the robot to stop just less than a foot from your guests, so type:

95<RET>

(The value 95 is the minimum distance read by the sonar, and is just less than a foot. For further understanding of sonar values, see the section on "Sonar Machine Code Algorithm" in Chapter D on "Software" in the RB5X Reference Manual.)

19. Once again, close the loop by typing:

    REPEAT THIS LOOP<RET>

20. When either a bumper is touched or the robot's sonar detects an object, you want the robot to stop. Type:

    STOP ALL MOTION<RET>

21. After it stops, you want RB5X to introduce itself. Type:

    SPEAK<RET>

    The system then prompts with the message:

    the phrase called:

    Enter:

    "SAY THE INTRODUCTION<RET>

    This is the phrase you loaded earlier in your robot task with the X LOAD THE INTRODUCTION command. Don't forget the first set of quotation marks. RCL enters the second set for you.

22. Once it has spoken its piece, RB5X should return to its starting point to await another guest. So tell RB5X to turn around by entering:

    SPIN CLOCKWISE<RET>

    The system then prompts you with:

    this many degrees:

23. Since you want the robot to turn around and return to the start, type:

    180<RET>
    MOVE FORWARD<RET>

Copyright 1984 All rights reserved.

24. RB5X should move forward until you press a bumper, so a loop would be useful here. Type:

    BEGIN A LOOP<RET>

25. The robot should continue to move forward until you instruct it to stop, by pressing one of its bumpers. Type:

    EXIT IF ANY BUMPER TOUCHED<RET>

26. Don't forget to close the loop:

    REPEAT THIS LOOP<RET>

27. Once RB5X has returned to the starting point, you need to have it facing front again, so type:

    SPIN CLOCKWISE<RET>

    At the prompt

    this many degrees:

    type:

    180<RET>

28. To have the RB5X turn off its lights, type:

    TURN OFF THE FLASHING LIGHTS<RET>

29. You now want the robot to sit and wait until the next guest arrives. So, to return to the beginning of the entire program, type:

    JUMP<RET>

    At the prompt

    to the line called

    type:

    "BEGINNING OF PARTY<RET>

    Don't forget to type the first set of quotation marks. RCL adds the second.

30. To redisplay the entire task on the screen, type:

    L<RET>

 Copyright 1984 All rights reserved.

This displays on your screen the RCL robot task listing you see in Appendix 1. (To list only part of the task type the number of the last line you want to see, for example, to list lines 1-15, type 15<RET>.)

31. To signal the end of this robot task, type:

    Q<RET>

    This stands for Quit. The system displays:

    END (Task is 15% full.)

    Once you have ended the task, your screen displays the "What would you like me to do next?" prompt.

32. To load this robot task into the RB5X, switch your robot on (this means the top of the ON/OFF switch is pushed in) and connect the RS-232 cable to the port on the serial communications card in your Apple and Port 1 on the robot.

33. Type:

    BUILD AND LOAD<RET>

The system displays

    Build and Load an RB5X robot task.

34. At the prompt:

    Please enter the name of the robot task:

    type:

    PARTY<RET>

35. The system responds with

    Would you like the task checked for errors? (Y/N)

    Type: Y or N and <RET>

    If you type Y, the system takes a short time to check your program for errors then displays:

    Now building robot task PARTY... One moment, please.

    Note that the process of building and loading the task may take several minutes. You may also use BUILD and LOAD as separate commands; when BUILD is used alone you must enclose the name of the robot task that follows the command (e.g., PARTY) in quotation marks.

Copyright 1984 All rights reserved.

The BUILD command begins the process of translating RCL commands into the Tiny BASIC code RB5X understands.

Only one robot task can be translated at a time, and only one robot task at a time remains in the Tiny BASIC code form on your RCL diskette.

For example, you just built the robot task PARTY; it now exists in Tiny BASIC code form on your RCL diskette. If you create another robot task, DANCE, you must build DANCE before you can download it to RB5X, but if you want to reload the robot task PARTY, you do not have to build it again. Therefore, you must build every robot task before you load it into your robot, unless it is the task you last built when you last used RCL.

36. When the system finishes building the task, the system displays the message:

    Ready to load into the RB5X. Press any key to continue.

    After you press a key, the system displays:

    Now loading ... One moment, please

    It also displays a status line indicating the loading of sonar, voice, and the robot task.

    When the computer finishes loading this task into your RB5X, the "What would you like me to do now?" prompt again displays.

37. Disconnect the robot from the RS-232 cable, position it where you want it to wait for your guests, and press any bumper for it to begin its greeting.

38. So now you have a complete robot task in RCL. But suppose you want to change your task? RCL allows you to add, change, or delete lines in your robot tasks using the Savvy EDIT command mode.

    Suppose you want your robot to honk its horn as it begins to approach each of your guests. To have RB5X do this, you need to add a horn-honking command to your robot task, PARTY. Since you just loaded PARTY into your robot, your screen now displays the "What would you like me to do now?" prompt.

    Type:

    EDIT<RET>

    to go into the Savvy edit mode, and at the prompt:

    the

 Copyright 1984 -- All rights reserved.

Type:

PARTY<RET>

39. Your screen now displays:

0 --- Task PARTY
1 Does

which is the beginning of the task listing for PARTY.

Type:

L<RET>

to list the primary tasks that make up this robot task.

40. Looking down the list of primary tasks, you see that line 9, which contains the primary task, MOVE FORWARD, is the line where RB5X begins to move toward your guest. It is after this line that you want to add a command that makes the robot honk its horn.

Type:

9<RET>

Your cursor moves to line 10, and your screen displays the text of lines 0 through 9. You can use this procedure to move to any line in your robot task when you are in the Savvy edit mode.

41. To add the horn-honking command, type:

HONK<RET>

RCL prompts you with:

the horn for this many seconds.

42. Type:

3<RET>

You have now added the horn sound to your robot's approach.

43. To view your new robot task PARTY, type:

L<RET>

Copyright 1984 All rights reserved.

44. Now type:

    Q<RET>

    to exit the edit mode and return to the "What would you like me to do now?" prompt.

45. Now you can load the new task into your RB5X and run it using the procedures beginning with step 32. Switch RB5X off, then on again or press the software reset button on the interface panel before you BUILD AND LOAD to clear the previous PARTY robot task from its memory.

46. You also use the Savvy edit command mode to delete or change robot task lines as follows:

    When the system displays the "What do you want me to do now?" prompt, type:

    EDIT<RET>

    At the prompt:

    the

    type:

    PARTY<RET>

47. You are now again in the Savvy edit mode. Type:

    L<RET>

    to list your robot task.

48. Let's delete line 10, the HONK command we previously inserted. Type:

    10<RET>

    to move the cursor to line 10.

49. Type:

    D<RET>

    to delete the line.

50. To change a robot task line, you follow the above procedure for deleting the line, then type in the new line and <RET>.

 Copyright 1984 All rights reserved.

51. To delete an entire robot task, you must delete each of its lines separately. The task name, however, exists on your RCL diskette even after you delete all the task lines. Because you cannot completely remove old tasks from the RCL diskette, we encourage you to reuse old task names or to rename them using the Savvy RENAME command.

52. RENAME is used following the "What do you want me to do now?" prompt. Type:

    RENAME<RET>

    at the prompt:

    old name

    type the name of the robot task you want to change and <RET>.

53. At the prompt:

    to new name

    type the new name of the robot task and <RET>.

To better illustrate the difference between what you wrote in RCL and what RCL told RB5X, we have included in Appendix 1 a copy of the "Party" robot task and the Tiny BASIC translation. As you can see, the RCL text is just 24 lines long -- the Tiny BASIC text stretches to 115 lines of complicated computer code that only machines like RB5X should be forced to learn!

We have also included the detail (which we got by using the Savvy task DETAIL) for each of the primary tasks used to compose the robot task we just wrote. These tasks are listed alphabetically in Appendix 2.

You are now ready to design your own robot tasks for your RB5X. Use the sample scenario form in Appendix 4 to outline the building blocks for your own scenarios and plans.

To start composing own your robot task, follow steps 1 through 8 of the Sample Robot Task using your new scenario data.

Copyright 1984 All rights reserved.