Atari Logo InfoPac

ATARI INC.
CONSUMER PRODUCT SERVICE
PRODUCT SUPPORT GROUP
1312 Crossman Ave.
Sunnyvale, CA 94088

800-672-1404 inside CA
800-538-8543 outside CA

DEMOPAC #9
REV.1 JG/11/83

ATARI LOGO PRODUCT INFORMATION

Atari Logo was developed for Atari Inc. by Logo Computer Systems Inc. of Montreal, Canada. It is a derivative of and highly compatible with an earlier version of Logo developed for the Apple computer called Apple Logo. The release of Atari Logo is historically significant because it allows for the first time a "full" implementation of Logo on an inexpensive home computer.

Atari Logo has been designed to take advantage of much of the hardware capabilities of the Atari system. This has resulted in some enhancements over previous versions of Logo, most notably the availability of four programmable "turtles" with collision detection and a player-missile "shape" editor. Other enhancements include a 128 color spectrum, easy access to sound and controllers, and the ability to call assembly language subroutines. Also, Logo programs can be easily saved off to a storge device (disk, cassette, printer).

Although widely used in education, Logo is a powerful and sophisticated language. It was designed to have "no threshold" and "no ceiling". It is actually a subset of LISP, a language known for its use in the area of artificial intelligence research. The Atari Logo version is a full featured Logo and includes advanced computer science constructs such as list processing, recursion and local variables.

Logo is generally considered to be an excellent introductory programming language for children and adults. It has already been widely used in the educational field to teach the structure of computer programs and to illustrate the flow of logic within a computer program...concepts that are central to all programming languages.

Atari Logo is available in two forms....a consumer package and a special school package. The consumer package includes the following:

- *16K LOGO Cartridge
- *Introduction to Programming through Turtle Graphics Manual
- *ATARI LOGO Reference Manual
- *ATARI LOGO Reference Guide Manual
- *Retail price: $99.95

The school package is available in several different versions:

1) Econony Pac
  * 16K cartridge
  * ATARI LOGO Reference Guide. Retail price: $74.95

2) Manual Pac
  * Same three manuals listed in consumer package. Retail price: $29.95

3) Full Pac
  * Same as the consumer package. Retail price: $99.95

Both the Consumer package and the three school Pac's are currently available at your local Atari retailer, at software stores or from computer mail order houses. They are all fully compatible with all ATARI products. The Logo cartridge fits the cartridge slot on the 400/800 computers as well as the entire series of XL computers.

ATARI LOGO does have limitations. They include: no fill command; Logo language uses graphics mode 7 only; splits words at end of screen (unlike Atari PILOT or AtariWriter); no debugging aids (trace, remarks, etc.); pressing SYSTEM RESET erases all programs in memory (programs not erased in PILOT, BASIC, etc.); additional documentation may be necessary to learn advanced LOGO features, such as recursion and list processing (a list of available literature on the LOGO language can be provided upon request).

For additional information concerning Atari Logo, or any other Atari hardware and/or software product call Atari's toll-free Customer Service line at:

OUTSIDE CALIFORNIA - (800) 538-8543
INSIDE CALIFORNIA - (800) 672-1404

Thank you for your interest in Atari Home Computer products.

Reprinted with permission from the National Logo Exchange

The National LOGO Exchange

ISSN 0734-1717

Volume 2 Number 1 FORWARD 100! September 1983

Here Comes Atari Logo

by
Shoneen Gervich

Atari Logo is on the way! Boasting four turtles, floating point arithmetic, and a full array of list processing capabilities, it has the potential to become one of the most popular versions of Logo. Developed by LCSI to take advantage of the unique capabilities of the Atari computers, it features most of the "usual" Logo primitives plus some exciting new ones which give the Logo juggling act a completely different slant.

THUNDERING TURTLES

A herd of four turtles is available to carry out both individual and collective commands. That is, each turtle can move independently or in concert with the others, responding to primitives such as TELL, ASK, EACH, and WHO.

Moreover, these turtles actually look like turtles instead of the sometimes confusing triangles. Each has a round body with four legs and a tail. A turtle head points clearly in the direction of the heading.

TURTLE CONTROLS

Each turtle has three pens, each of which can draw in any of 128 different colors. Moreover, each of these colors is available for the background, and for the colors of the turtles themselves. The Atari palette will give your students color possibilities which are practically unlimited.

The drawing speeds of the turtles can be controlled by the primitive SETSP and a number from 0 to 199. Thus, a turtle which has been told SETSP 5 will draw rather slowly, a feature perhaps useful with beginners, while one with SETSP 150 will draw much faster. Also, a line drawn at high speed will be less solid than one drawn slowly, giving an added texture capability.

The Atari Logo turtles can change shape, making it possible to create animated sequences. Up to 15 user-defined shapes may be in memory at any one time. Cars, planes, human figures, animals, and the like can come to life on the screen very easily by rapidly changing turtles from one shape to another, in a manner similar to that often done with the TI LOGO Sprites.

Possibly the most unique feature of Atari Logo is its collision detection. Beings called "WHEN Demons" check to see when specified turtles collide or when they cross lines. Predetermined lists of commands are then carried out. For example, if Turtle 1 and Turtle 3 collide, they could change shape or rebound or almost anything else imaginable.

The TOOT primitive will create sounds which can be played through the speaker of the television. Frequency, duration, and volume may be specified for two voices, making it possible to compose music, create special effects, and the like.

With Atari Logo, the heart-felt desire of almost every child to create a video game is now within reach.

NUTS AND BOLTS

Atari Logo comes in a plug-in 16K ROM cartridge which will fit the 400, 800, and 1200 models and their successors. When used with a 48K machine, such as the 800, Atari Logo has about one thousand more nodes of memory available in the workspace than any of the Apple versions.

Atari Logo comes with a reference card, and two books, "Introduction to Programming Through Turtle Graphics" and "Atari Logo Reference Manual."

A disk drive is recommended, in order to take full advantage of the language. Storage and retrieval tasks are straightforward.

The cost of an Atari Logo system is quite low. The language, a disk drive, and 64K computer can be put together for under $900. By using a tape cassette storage system and the 16K computer, you can drive the price below $400. This is about the lowest price you will find for a full Logo implementation.

EFFECTS

You may expect Atari Logo to have a significant effect on both teachers and students, and on the teaching of Logo.

The wealth of options available with Atari Logo will stimulate those already familiar with other versions of the language, and will serve as a beckoning enticement to those contemplating entering the Logo microworlds for the first time.

The multiple turtle capability makes possible new ways of thinking about solving problems. Turtle control becomes more complex, more challenging. Creativity and imagination become even more important.

Atari has made Logo affordable, powerful, and irresistible. It is up to us and our students to make the most of it.

(For more information, write: Atari Logo, PO Box 16525, Denver CO 80216 — Ed.)

Shoneen Gervich teaches children's classes in PILOT, Logo, and (only if she has to) BASIC, through various community agencies in the South Bay area of San Francisco.

NLX Back Issues Available

A limited number of sets of Volume 1 of The National Logo Exchange newsletter are available, covering the period from September 1982 through May 1983. Many beginning Logo teachers have found this set to be a helpful resource for ideas and activities, as well as for general Logo information. To order your set, send $25 to NLX Volume 1, PO Box 5341, Charlottesville VA 22905.

From The Editor

September always has a special zing to it, doesn't it? There is something invigorating about hallways alive with young people once more. Facing a class full of new friends and new challenges is stimulating. And, of course, having Logo in your lesson plans for the coming year is exciting!

This school year is going to be one of incredible growth for Logo. Thousands of teachers everywhere are starting to use it in their classrooms. More and more versions are being released for more and more computers. Several Logo books are being published, and more are on the way. Nearly every educational computing magazine now has a Logo section.

This growth has excited many people: teachers, parents, administrators, professors, software developers, publishers. And the children are the most excited of all!

With you, we welcome this growth. Our motto FORWARD 100! was chosen to reflect this attitude. FORWARD is the direction of progress, improvement, growth. 100% of our newsletter supports you, the Logo teachers. The exclamation mark represents the excitement Logo brings to learning. By working together and sharing Logo ideas, we can multiply this growth to proportions that cannot be imagined. Your students are the direct beneficiaries of this growth!

To all our earlier subscribers, we send a hearty WELCOME BACK! We appreciated meeting you last year, and look forward to working with you again this school year. To our many new subscribers, we extend a special HELLO! We hope that this year will be one of healthy challenges and professional growth for all of us.

As a personal note, I'd like to add that, after 13 years of marriage, my wife, Posy, and I have been blessed with a child! We are pleased that our son, Kyser, will be able to grow up with Logo, and with Logo teachers such as you!

FORWARD 100!

Tom

Atari Logo / PILOT Comparison

Atari Logo and Atari PILOT are both excellent user friendly languages that have been especially designed to introduce computer programming to beginners of all ages. It is the purpose of this comparison to clarify the differences and usage of the two languages.

The design and purpose of a language determine the types of activities it is good at and those functions which are more difficult to perform. The design and purpose of PILOT and Logo are, as we shall see, quite different.

PILOT stands for Programmed Inquiry, Learning Or Teaching. It was designed by Dr. John Starkweather at the University of California in San Francisco, to assist teachers in generating computer aided classroom materials. PILOT was designed to be easy to learn and to use. In addition, Dr. Dean Brown in his work at the Stanford Research Institute discovered that PILOT was a good first programming language for children as well. The eight or nine "core" PILOT commands allow the creation of interactive lessons and dialogues. Atari PILOT has been significantly enhanced and is a much more powerful language than the original.

Logo was also developed in a university setting. In 1970 Logo Group Leader Seymour Papert moved into MIT's Artificial Intelligence Research Laboratory and began implementing Logo on a large computer. By the late 70's, the first versions of Logo on microcomputers had arrived.

The design goal of Logo was different than that of most computer languages. Logo was not designed to solve a particular type of problem, it was designed as a tool for thinking. The MIT Logo group had studied the learning theories of the Swiss educator Jean Piaget, and modeled Logo after Piagetian learning theories. The self discovery aspect of Logo turtle graphics is a direct result of this influence. Atari Logo was developed by Logo Computer Systems, Inc. (LCSI) of Montreal and is headed by Seymour Papert. LCSI also implemented a version for the Apple computer called "Apple Logo".

From the above it is evident that PILOT was developed primarily as a tool for teachers and Logo as a tool for learning. Consequently, PILOT is the language of choice for the creation of interactive dialogues and computer aided instruction, whereas Logo is excellent for learning structured programming and creating graphic designs. A closer look at each language will reveal the tasks that each excels at.

TURTLE GRAPHICS

Both languages contain a graphics system called "turtle graphics". Turtle graphics features a graphics cursor called "the turtle" which is incorporated into a powerful but easy to use set of graphics commands. Along with the familiar cartesian coordinate system found in BASIC and other languages, a "polar" or relative system of referencing is featured. Graphics commands are given with reference to the turtle's position rather than fixed coordinates. A square may easily be drawn in turtle graphics by the statement REPEAT 4 [FORWARD 40 RIGHT 90] whereas in a cartesian system the following BASIC program would be required to produce the same square: PLOT 5,5:DRAWTO 5,45:DRAWTO 45,45:DRAWTO 45,5:DRAWTO 5,5. As is obvious from the above example, the turtle graphics description is more intuitive and easier to visualize than the cartesian.

In comparing the PILOT and Logo turtle graphics environments, Logo has some distinct advantages. The Logo turtle looks like a real turtle, whereas the PILOT turtle is invisible and can only be located by its "pen" trace. Logo features four turtles where PILOT has only one. The Logo turtles are "dynamic" and can each move about at different speeds. The PILOT turtle is static or is what some have called a "normal boring" turtle. The Logo turtles can change their color and can also detect collision with other objects, making for the creation of video games. The Logo turtles can also change their shapes which can lead to the creation of simple animation sequences. A plus for PILOT is the fill command which allows turtle designs to be filled in with solid colors. The Logo turtle does not have this feature.

One of the major drawbacks to PILOT turtle graphics, in addition to there being no visible turtle, is the graphic screen limitation. PILOT functions in the split screen mode only, making it impossible to view a complete full screen graphics display. Logo allows both split and full screen modes, <u>and</u> the ability to switch back and forth between text, split screen and full screen <u>while a program is running!</u> In addition, the PILOT GR: command must be used for each graphics command line while no special leading commands are needed for Logo graphics. However, it should be mentioned that PILOT turtle graphics execute <u>much</u> faster than Logo.

PROGRAMMING STRUCTURE AND EDITING

Both languages encourage structured programming, although Logo is superior in this respect. Structured programming is defined here as the ability of a program to be divided into independent smaller programs (sometimes called modules or procedures) and tested independently of the other program segments. PILOT is somewhat modular while Logo is completely modular. PILOT program units are called modules while Logo units are called procedures. PILOT uses line numbers like BASIC. Logo, like Pascal, uses no line numbers at all. Logo is fully extensible in that a defined procedure is treated exactly like a built in command or "primitive". This allows the user to in effect "customize" a program by building a vocabulary tailored to a specific situation. PILOT, on the other hand, allows other modules to be called with a simple command "U:" but does not treat modules as extensions of the language. Many computer scientists would include local variables as a requirement for defining structured programming. Logo includes local variables, PILOT does not.

One of the most useful features of Logo is recursion: the ability of a procedure to call itself. Recursion is more than mere repetition. With recursion, a complex problem can be described in simpler terms of itself and its inclusion in Logo allows for simple and elegant solutions to complex problems. The use of recursion, list processing, and local variables make Logo quite suitable for studying more advanced computer science concepts. Logo has been used in some colleges as preparation for learning Pascal.

One of the most unique features of PILOT is ease with which complex pattern matching can be performed on user input. Decisions can then be made on the presence or absence of a "match". Using only a few simple commands, reasonably sophisticated interactive instructional programs can be created.

Editing in PILOT is done as in BASIC with the exception that PILOT has a renumber and auto number function. As mentioned before, Logo is a procedural language and has no line numbers. It does, however, contain a built-in editor which enables one to add, delete and insert lines much like a mini/text editor. The titles and/or definitions of all procedures and variables can be examined with single word commands.

DEBUGGING

PILOT is clearly superior to Logo with respect to debugging programs. PILOT has a trace function which slowly prints the program lines and line numbers to the screen as they are executed by the computer. This feature makes it very easy to follow a program's excecution sequence and to identify and correct bugs. PILOT also allows for two types of remark statements to be inserted into program listings. Logo does not have a trace function or a remark command.

DATA TYPES AND VARIABLES

PILOT, like BASIC, supports two data types, string and numeric. However, unlike BASIC, strings never need be dimensioned. PILOT automatically dimensions all strings to 256 characters. All PILOT strings are indicated by a "$" preceding each string name whereas the "#" precedes each numeric variable. A numeric variable is designated by the "#" followed by a single capital letter (A-Z). Therefore a maximum of 26 variables can be assigned at any one time.

One of the nicest features of PILOT variables is the ease in which they can be combined with text literals. When using a print command, many languages require that the literal text be separated from the variable by using quote marks, brackets, etc. PILOT allows one to freely mix text literals and variables in its print statements, making it especially simple and straight forward for beginners to write programs using variables with screen or printer output. Logo, on the other hand, requires all text literals in print statements to be set off with brackets while the variables must remain outside the brackets. This is cumbersome and more difficult to use than PILOT's method.

Logo contains two data types: words and lists. Numbers are words that happen to contain digits. All data types in Logo are called "objects" and are in many ways treated identically. No distinction is made in the designation of Logo variables. Logo automatically takes care of memory allocation for all variables, so as in PILOT, no dimensioning is required. For all practical purposes, there is no limit to the length of variable names.

Variables in Logo may be "local" or "global". A local variable only exists while the procedure containing it is being executed. When the procedure ends, the variable has no value. Thus, several procedures can use the same variable name and each procedure may produce a different value for the same variable without causing Logo to become confused. The use of local variables is a very powerful feature which allows for Logo procedures to pass variables to each other and to call themselves recursively. Global variables are constant and at any one time have the same value whenever they appear in a program. Both BASIC and PILOT have only global variables.

SCREEN OUTPUT

Screen output is another area where PILOT excels. When a PILOT program is run in the text mode, the screen is automatically cleared. This removes any unwanted text and makes for a very readable screen without having to explicitly clear the screen each time a program is run.

Along with clearing the screen, PILOT, unlike almost all other languages including Logo, does not break words when displaying them on the screen. If a word cannot be displayed without being broken, PILOT removes the entire word to the next screen line and displays it in its entirety. This is especially useful when a program requires input from the user and cannot control the positioning of the text display. Logo breaks words at the second to last screen column and inserts a right arrow if the word continues on the next line. The arrow may be informative but it makes for messy screen displays.

PILOT also has the ability to display large colorful text by accessing graphics modes one and two. Logo is restricted to the standard text mode

SOUND

Both Logo and PILOT can generate a wide variety of sounds through the T.V. speaker. These sounds can be used to create music or add sound effects to programs. Logo allows two voices with frequency, volume, and duration while PILOT allows four voices and duration but no volume. The Logo sound commands are more flexible but not as easy to use as PILOT's.

Atari Logo Resource Guide

The Logo Resource Sheet has been compiled to provide additional resources for users of Atari Logo. Most of the sources mentioned refer to versions of Logo that are similar to Atari Logo and those resources that refer specifically to Apple Logo will be the most compatible. The primary difference between Apple and Atari Logo is that Apple Logo has additional list processing commands while Atari Logo contains an enhanced set of turtle graphics commands.

Currently, only two books have been announced specifically relating to Atari Logo. They are Dan Watt's book Learning With Atari Logo and David Thornburg's Computer Art And Animation: A User's Guide to Atari Logo (both available in early '84). Other Atari Logo books are forthcoming and they will be included in future updates of this resource Guide.

Three books included here deserve special mention: Papert's Mindstorms, Abelson's Turtle Geometry and Abelson's Apple Logo. Mindstorms is the most comprehensive expression of the Logo philosophy of education and computing while Abelson's Apple Logo is the most useful manual to date for the serious student of the Logo language. (Do not confuse Apple Logo with Abelson's Logo for the Apple II. Apple Logo is the version that is most compatible with Atari Logo. Logo for the Apple II is for the MIT versions of Logo.) Abelson's Turtle Geometry is a profound excursion into the realms of turtle graphics and has been used as a college level text. Many of the resources included here are very useful in preparing a Logo curriculum and for introducing Logo to beginning students both in and out of a classroom setting.

BOOKS

Abelson, Harold. Apple Logo, New York: BYTE Books/McGraw-Hill, 1982. The best single reference work to date. Features a complete description of Apple Logo which is 90 % compatible with Atari Logo. The book emphasizes the more advanced features such as list processing, recursion and local variables.

Abelson, Harold and Andrea diSessa. Turtle Geometry, Cambridge,MA: MIT Press, 1981. A serious college level text on turtle graphics. Proves that turtle geometry is not just kid stuff.

Beardon, Donna. One, Two, Three, My Computer and Me: A Logo Funbook for Kids, Reston, VA: Reston Publishing Co., 1983.

Beardon, Donna, Kathleen Martin and Jim Muller. The Turtle's Sourcebook, Reston, VA: Reston Publishing Co. 1983. Formally distributed by the Young People's Logo Association, this book is filled with turtle graphics worksheets and and activities.

Bitter, Gary and Nancy Watson. Apple Logo Primer, Reston VA: Reston Publishing Co., 1983.

Burnette, J. Dale. Logo: An Introduction, Morristown, NJ: Creative Computing, 1983. A short collection of turtle geometry explorations.

Goldenberg, E. Paul. Special Technology for Special Children, Baltimore: University Park Press, 1979. Describes the use of Logo and computers with special-needs children.

Minnesota Educational Computing Consortium (MECC.) Apple Logo in the Classroom. MECC Distribution Center, 2520 Broadway Dr., St. Paul MN 55113. A Logo curriculum for children in grades 5 thru 9 - includes teachers manual and worksheets.

MIT Logo Group. Bibliography of Logo Memos.Mit Artificial Intelligence Laboratory, 545 Technology Square, Cambridge, MA 02139. Capsule descriptions of over 60 publications describing more than ten years of Logo research at MIT.

Papert, Seymour. Mindstorms: Children, Computers and Powerful Ideas, New York: Basic Books, 1980. The philosophy of Logo by its chief proponent. A must for anyone who wants to understand the Logo educational philosophy.

Ross, Peter. Introducing Logo: For the Apple II, TI 99/A, and the Tandy Color Computer, Reading, MA: Addison-Wesley, 1983. Covers list processing and structured programming as well as turtle graphics. Includes several chapters of projects and activites to enhance problem solving capabilities.

Thornburg, David. Discovering Apple Logo, Reading, MA: Addison-Wesley, 1983. The study of how Logo can relate to art and patterns in nature. Covers such topics as fractals and the golden mean.

Watt, Daniel. Learning With Logo, New York: BYTE Books\McGraw-Hill, July 1983. A hands on guide filled with projects for beginning and intermediate users. Includes interactive games and list processing.

MAGAZINES

BYTE Magazine, Logo Issue, August 1982. A good general introduction to the various aspects of Logo programming and philosophy.

Classroom Computer News, April 1983. An excellent all Logo issue which features an excellent article by Tom Lough entitled "Is There Logo After Turtle Graphics?".

Computing Teacher, The; November 1982. An all Logo issue featuring "Creating a Logo Environment" by Tim Riordin.

Kilobaud Microcomputing, September 1981. Another all Logo issue which includes "Logo and the Great Debate" by Ricky Carter.

The following magazines have regular columns on Logo and/or turtle graphics:

Compute! "Friends of the Turtle" by David Thornburg.

Watt, Daniel. Learning With Logo, New York: BYTE Books\McGraw-Hill, July 1983. A hands on guide filled with projects for beginning and intermediate users. Includes interactive games and list processing.

MAGAZINES

BYTE Magazine, Logo Issue, August 1982. A good general introduction to the various aspects of Logo programming and philosophy.

Classroom Computer News, April 1983. An excellent all Logo issue which features an excellent article by Tom Lough entitled "Is There Logo After Turtle Graphics?".

Computing Teacher, The; November 1982. An all Logo issue featuring "Creating a Logo Environment" by Tim Riordin.

Kilobaud Microcomputing, September 1981. Another all Logo issue which includes "Logo and the Great Debate" by Ricky Carter.

The following magazines have regular columns on Logo and/or turtle graphics:

Compute! "Friends of the Turtle" by David Thornburg.

Computing Teacher, The; "The Logo Center" by Kathleen Martin and Tim Riordin.

Softalk "Logo, the Voice of the Turtle" by Jim Muller.

NEWSLETTERS

Follk, Friends of LISP/Logo & Kids, a non-profit organization formed by a group at San Francisco State University. 436 Arballo Dr., San Francisco CA 94132

Logophile, published by the College of Education, MacArthur Hall, Queen's University, Kingston, Ontario K71 3N6

National Logo Exchange, published by Posy Publications, P.O.Box 5341, Charlottesville, VA 22905

Polyspiral, published by the Boston Computer Society, Three Center Plaza, Boston MA 02108

The Logo and Educational Computing Journal, published by Interactive Educational Foundation, 1320 Stony Brook Road, Stony Brook, NY 11790

Turtle News, published by the Young People's Logo Association, 1209 Hillsdale Drive, Richardson, TX 75801.

PAPERS

The MIT Logo Group has published a series of memos and reports. They can be obtained by contacting the MIT Logo Group, 545 Technology Square, Cambridge MA 02139. Some of the titles are as follows:

Abelson and diSessa, "Student Science Training Program in Mathematics, Physics, and Computer Science." Logo Memo #29, MIT 1976.

Feurzig, Papert, Bloom, Grant, and Solomon, "Programming Languages as a Conceptual Framework for Teaching Mathematics." Report #1889, Bolt, Beranek and Newman, Cambridge, MA 1969.

Papert, "Teaching Children to be Mathematicians vs. Teaching About Mathematics." Logo Memo #4, MIT 1971.

Papert, "Uses of Technology to Enhance Education," Logo Memo #8, MIT 1973.

Papert, Abelson, diSessa, Watt, "Assessment and Documentation of a Children's Computer Laboratory." Logo Memo #48, MIT, 1977.

Papert, diSessa, Watt, Weir, "Final Report for the Brookline Logo Project, Parts I, II, and III." Logo Memos #53 and #54, MIT 1979.

Papert and Solomon, "Twenty Things to do with a Computer." Logo Memo #3, MIT 1978.

Papert and Weir, "Information Prosthetics for the Handicapped." Logo Memo #51.

Weir, "The Uses of Logo for the Diagnosis of Children's Abilities in Areas for Spatial Reasoning, and the use of Logo for Remediation." Internal working paper, MIT Logo Group, 1979.

Weir, "Evaluation of Cultivation of Spatial and Linguistic Abilities in Individuals with Cerebral Palsey." Logo Memo #51, MIT, March 1980.

Weir, and Emmanuel, "Using Logo to Catalyse Communication in an Autistic Child." Department of Artificial Intelligence Memo #15, University of Edinburgh, Scotland, 1976.