December 1980
USA $2.95/DM 9.80

kilobaud

MICROCOMPUTING™

Scroll Over, Beethoven

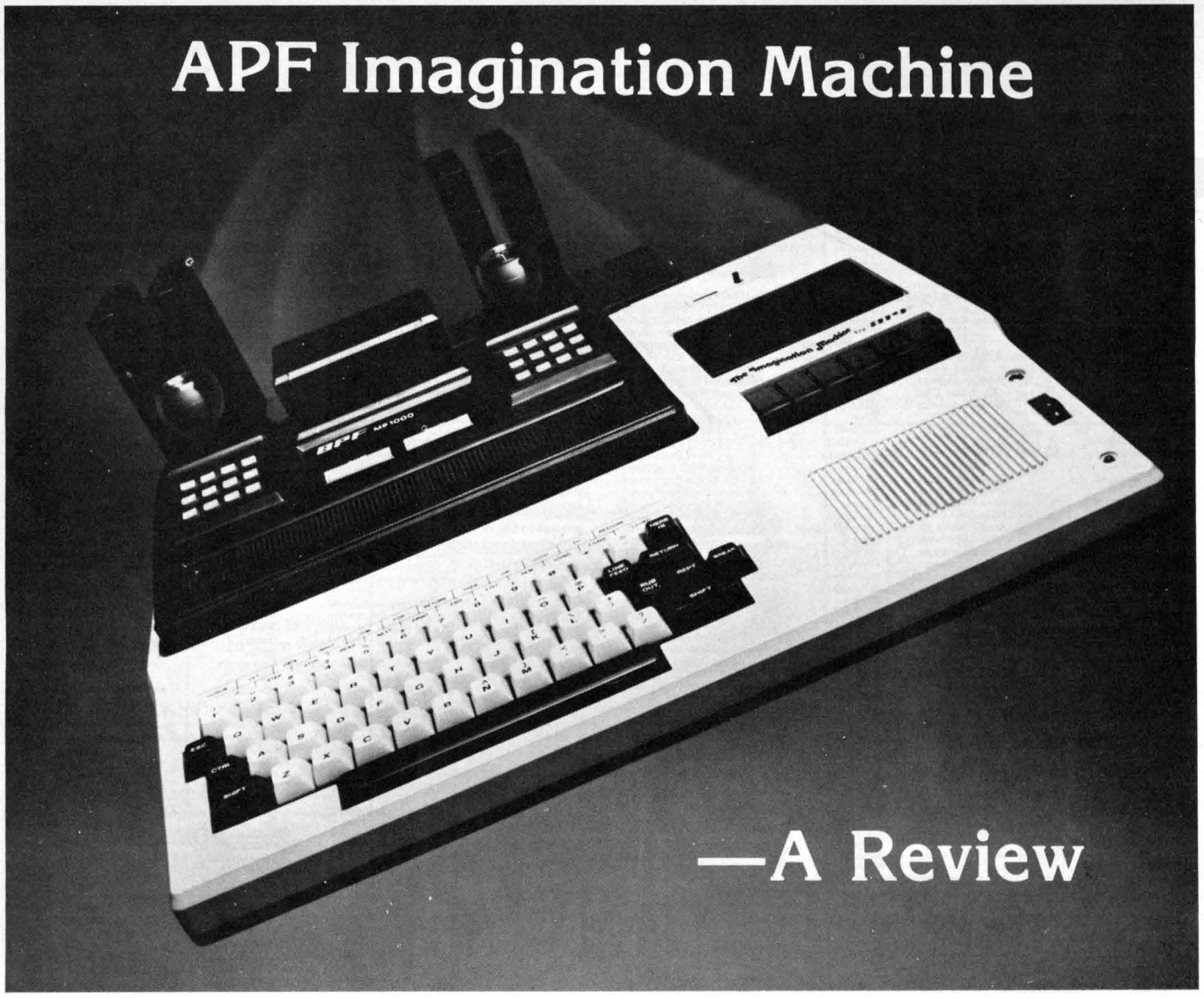
Tune Up Your Micro for Music

Special Music/Sound Generation Issue

ZX-80, Compucolor II and Imagination Machine Reviews ☐ Book Publishing with an Apple ☐ Connecting to The Source ☐ A Systems House Rides the Bumpy Rainbow ☐ Wine Making with the SWTP ☐ Software Reviews for North Star, Heath and CP/M

0 74470 65945 12

APF Imagination Machine



—A Review

The Imagination Machine may look like a single unit, but it actually consists of two parts, which sit on top of each other and are joined by a rigid connector. The MP1000 microprocessor unit (top left) contains the central portion of the computer. The MPA-10 base unit contains more RAM memory, the alphanumeric keyboard, cassette deck and expansion circuitry.

Peter A. Stark
PO Box 209
Mt. Kisco, NY 10549

Since the 6800 is my favorite microprocessor, the ads for the Imagination Machine from APF Electronics, Inc. (which uses the 6800), immediately caught my eye. I couldn't wait to get my hands on one and try it. It turned out to be a very interesting system.

The Imagination Machine is a self-contained, desktop computer with a full alphanumeric keyboard, two separate game paddles with their own calculator-style keyboards and a built-in cassette deck. It connects to the antenna terminals of a color TV set, which provides full-color graphics as well as a 16-line by 32-character alphanumeric display. In addition, the Imagination Machine provides an audio output via the TV set's speaker.

It can be programmed in either BASIC or machine language, and programs can be entered from the keyboard, disks or cassettes or via plug-in ROM cartridges. A variety of game, home finance and utility cassettes and cartridges is available from APF dealers.

The basic Imagination Machine (which APF now calls the System I) costs $599 with cassette I/O; the System II version at $995 adds the $100 expansion interface, $150 disk interface and $400 minifloppy disk drive and provides a disk system at an even better price.

Though the Imagination Machine can be operated with just the supplied programs, there is enough to make a review of it from a more technical standpoint worthwhile.

The Hardware

Although photographs show the Imagination Machine as a single unit, it actually consists of two parts, which sit on top of each other. One unit—the smaller, black assembly at the top left—is the MP1000 microprocessor unit, which contains the central portion of the computer. The other—which is the larger, gray assembly at the bottom—is the MPA-10 base unit. It contains more RAM memory, the alphanumeric keyboard, cassette deck and expansion circuitry.

These two units are joined by a rigid metal bracket and connector assembly so that they cannot be moved apart. Each unit has its own power supply with an external power transformer, which plugs into a wall outlet.

To expand the System I machine, APF offers the BB-1 building block. This unit plugs into the back of the MPA-10 and, in turn, provides four additional plug-in slots. At this point, APF makes three modules that fit these slots—an 8K RAM cartridge, an RS-

232C serial interface and a minifloppy disk interface.

The MP1000 Microprocessor Unit

The MP1000 is actually a self-contained TV game (and was first introduced as just that). Except for the lack of general-purpose I/O equipment, it is a complete computer all by itself (I suspect that it would make an excellent—and inexpensive—color video board if interfaced to, say, an SWTP 6800 system). Like most TV games, this one has a pair of paddle controllers (which include a complete calculator-style keyboard as well) and a slot for plug-in game cartridges. Its 6800 microprocessor is not only a very powerful processor (not nearly well enough known, in my opinion), but is also an extremely simple one to learn and use.

A 6847 video display generator chip generates the color graphics and/or an alphanumeric display. In this system, the 6847 provides two graphics modes—low-resolution graphics with alphanumerics or high-resolution graphics without alphanumerics.

In low-resolution mode, the screen is divided into 16 lines by 32 characters. Each of the resulting 512 character positions can contain either an alphanumeric character —in several different color combinations —or can be further divided into four smaller boxes, each of which can be turned on or off.

In high-resolution graphics, the screen can provide either 128 by 192 or 256 by 192 graphics. In this mode, the screen is divided into 32 boxes across by 12 boxes down. Each of these 384 boxes can contain a specific pattern, which can consist of either 4×16 dots or 8×16 dots each. Each of these patterns is stored in a separate area of RAM, so that it can be moved from place to place on the screen without having to be redefined each time. This allows fast screen movement and is especially useful for preprogrammed games.

An MC1372 color video modulator provides an rf signal on TV channel 3. This is coupled to the TV set's antenna terminals through an FCC-approved antenna switch, so that the set can be used for either regular TV reception or computer use by just throwing the switch.

The data being displayed is held in a 1K RAM. In the low-resolution/alphanumeric mode, only 1/2K is used to hold the ASCII or graphics codes being displayed; then the other 1/2K can be used to construct another image to be rapidly swapped into the usable 1/2K (or stored on cassette).

In high-resolution graphics, the first 1/2K specifies which pattern is in each screen position, while the second 1/2K specifies the shape of each pattern.

All of this is done under control of a 2K ROM, which is pin compatible with the popular 2716, and which contains a variety of housekeeping and graphics routines, plus a game called Rocket Patrol.

The two joystick controllers are connected to the data pins of a PIA parallel interface chip, and software scans their joystick and calculator-style keys. The joystick uses four switches, rather than pots, for the four directions, and so does not provide a proportional readout.

At the back of the MP1000 is a 30-pin socket for either game cartridges or the connector, which attaches the MP1000 to the base assembly of the Imagination Machine. The connector has all of the bus signals, including the full data bus, address bus, Read/Write and +5 volts.

(For interested owners, APF publishes a complete technical manual for $2 which provides all of the diagrams and technical information on what's in the Imagination Machine and how it works. This distinguishes APF from many other computer manufacturers—not only do they offer the technical manual, but also each accessory manual contains a full schematic diagram, and even program listings of their BASIC interpreter are available at a nominal charge. Unlike some other manufacturers who hold back information which a purchaser could find useful, APF hides nothing.)

MP1000 Software

The Rocket Patrol game contained within the MP1000 game's internal ROM is not particularly exciting. But APF has available about a dozen plug-in ROM cartridges (mostly $19.95 each) containing a wide variety of games, including Catena, Hangman, Tic Tac Toe, Doodle, Bowling, Micro Match, Brickdown, Shooting Gallery, Baseball, Blackjack, Backgammon, Roulette, Keno, Slots, UFO, Sea Monsters, Break It Down, Rebuild, Shoot, Pinball, Dungeon Hunt, Blockout and Boxing.

Some cartridges contain just one game; some contain two or three. Some games are for one player; some are for two or four. Several game cartridges, such as Baseball, have excellent color graphics.

The MPA-10 Base Unit

When you put the MP1000 game on top of the MPA-10 base unit and connect them together, you have the complete Imagination Machine.

The MPA-10 adds RAM and I/O capabilities to the processor inside the MP1000 and makes it into a full-fledged computer.

Inside the base unit is 8K of RAM, which uses eight 4115 $8K \times 1$ dynamic RAMs; these are similar to the $16K \times 1$ 4116-style dynamic RAMs used in the TRS-80, Apple and other popular computers. (In fact, with fairly minor changes, the 4115 could be replaced by 4116 ICs, and the memory expanded from 8K of RAM to 16K; though this would add 8K of memory for perhaps $30–$50, it would also require some wiring changes and probably void the warranty as well. In the long run, it is easier to buy the 8K expansion cartridge from APF for $99.95).

A PIA inside the MPA-10 provides a number of I/O functions. Eleven of its parallel I/O pins are connected to a full 53-key keyboard, which is scanned by software. The keyboard has a good feel to it, and is easy to use even for long periods. Software scanning of the keyboard is not an optimal approach—it takes up too much CPU time—but it is an inexpensive method which is also used on the TRS-80 and other computers, an alternative to a separate keyboard scanning controller such as the 2376 used in more expensive systems.

The only disadvantage here is that a properly interfaced separate keyboard scanner will store a keyboard character even after the key is released; thus, the character will be entered even if the computer wasn't actually looking for it at the time you press the key. In the software approach, on the other hand, there are times when you must depress a key for a half second or more when a BASIC program is executing and not looking for input just then.

Next to the keyboard is a built-in cassette recorder which has an interesting function. It is a two-track deck, with one track used for digital data and the other for audio. Both the audio and digital tracks, as well as the motor during recording or playback, are controlled by the computer via the keyboard PIA.

This makes for some interesting possibilities. For instance, APF program cassettes contain the program on one track and an announcer's voice on the other. As the cassette is being loaded, you hear the announcer describe the program and how to use it. It takes about 45 seconds to load a full 8K program (at an effective rate of about 1500 baud), and this certainly makes the time pass quickly.

Since the audio and motor can be computer controlled, this machine could easily be used for educational purposes where the computer could play selected lessons or musical selections under program control, skip ahead or even chain to additional program segments. Either or both tracks can be recorded on the Imagination Machine itself, so you can experiment with this concept yourself.

Although the cassette deck could be used for any purpose, given the right program, the BASIC software has a specific method of recording or loading cassette data. When recording a cassette, it first records the screen memory in the MP1000 game unit, followed by the complete 8K (or

16K) memory of the MPA-10 base unit. When playing back, it first reads the screen memory and displays it, followed by the 8K of program storage. Thus, you can set up a picture on the screen to be recorded to tape; when the tape is then loaded, the screen will automatically show that picture, and the audio track can simultaneously provide music or narration while the program is being read. APF makes full use of this in their program cassettes, and it is a very pleasing effect.

On the back of the MPA-10 are three connectors—two 30-pin connectors just like that on the MP1000 game, and a 50-pin connector with some additional signals for the expansion box.

One of the 30-pin connectors is used to join the MPA-10 base unit to the MP1000 game unit. This is done with a rigid J-shaped bracket that contains a flexible printed-circuit board, which contains bus drivers and bidirectional transceivers.

The second 30-pin connector is used for ROM cartridges. The same game cartridges that fit the MP1000 alone can also be plugged into the 30-pin connector on the MPA-10 base unit, so that the full Imagination Machine can run these games as well. In addition, the base unit comes with a 12K BASIC cartridge, which allows programming in that popular language.

Finally, the 50-pin connector is for the external building block expansion box.

Building Block Expansion

Once you add the expansion box (at about $100), you can plug in an 8K RAM cartridge ($99.95), an RS-232C interface cartridge (also $99.95) or floppy disk interface ($149.95) with either the APF minifloppy drive ($399.95 with cabinet and power supply) or any other Shugart-compatible 5-¼ inch drive.

The 8K RAM cartridge is equipped with a switch to enable or disable this extra RAM as desired. This is useful, since the cassette storage always writes the contents of the entire RAM space on tape. 8K of memory will therefore save and load faster than 16K. Since the BASIC and disk operating system are in ROM, and since BASIC source programs are stored in a compressed form, 8K of RAM will hold fairly sizable programs, and so the RAM expansion can be disabled for most programs.

The EIA RS-232C serial port cartridge contains a tiny switch for selecting baud rates between 110 and 9600 baud. It has the standard 25-pin plug and provides for modem handshaking.

Selection of the serial port is made by a form of the PRINT statement. PRINT = 1 routes all succeeding output to the serial port instead of the TV screen. PRINT = 2 outputs to the serial port and also accepts all input from the port; in this mode the computer can be fully operated from a remote terminal or, with a modem, via a telephone line (though not with graphics).

The Imagination Machine can also be used as a terminal to a remote computer; the machine-language program to achieve this is provided in the serial port's instruction manual.

The floppy disk interface supports one or two disk drives. The disk operating system is inside the BASIC ROM cartridge, so that no extra disk space is required for the operating system.

Each diskette provides 72K bytes of storage and has room for up to 16 files, each with a seven-character file name. A disk can store BASIC programs, sequential files or random files.

APF BASIC

Supplied with the MPA-10 base unit is a ROM cartridge which contains APF Level I BASIC. This cartridge contains a 4K ROM and an 8K ROM, for a total of 12K; but not all of that is used for BASIC itself. Also included is the disk operating system, as well as routines for scanning the keyboard and for cassette I/O. It also has a small monitor for directly entering or displaying machine-lan-

This Page Intentionally Left Blank

guage programs or data in memory. Using this monitor, it is possible to enter machine-language programs or to add machine-language subroutines to BASIC programs for faster execution speed.

In a way, APF BASIC show the disadvantages of having a BASIC in ROM. If it were in RAM, it could be expanded, patched or changed; but in ROM it's fixed and unchangeable. (Though a ROM cartridge is somewhat easier to change than having a permanent ROM a la the TRS-80 or PET. A new Level II BASIC cartridge should be available by the end of 1980.)

APF Level I BASIC is about midway between an integer BASIC and a floating-point BASIC. All numbers are carried as 13 decimal (BCD) digits, with nine to the left and four to the right of the decimal point. This is perfect for money calculations up to $999,999,999.9999, as well as most general-purpose calculations, but would not be suitable for scientific calculations.

Since calculations are done in BCD, there is no round-off error; you needn't worry that the cents will be wrong in a large financial calculation, as they might be in a binary floating-point computer.

In keeping with the unscientific nature of the machine, there is a minimum of arithmetic functions—just ABS, INT, RND and SGN (RND is not very good, since it returns only two-digit numbers from 0.01 to 0.99). There is no SIN, COS or SQR, which makes some games difficult to program. There is only an integer exponentiation, so that calculations involving noninteger powers also are difficult. In fact, a prerecorded APF cassette program which calculates loan interests takes ages to compute compound interest—and finally gives the wrong answer—because of this lack.

String variables can contain up to 100 characters, which should be long enough for almost any purpose. The HP string convention—wherein every string must first be dimensioned for its maximum string length—is used. This makes it possible to save space by defining each string a different length, but makes programming nonstandard.

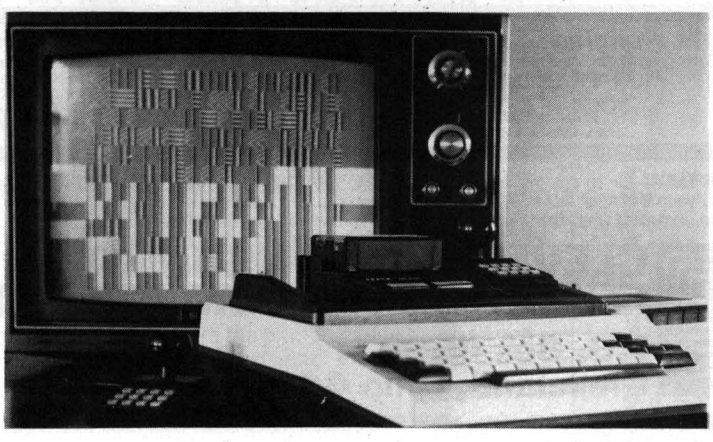
For example, the statement DIM A$(20) dimensions not an array, but a single string A$ with 21 characters. The subscripts start with 0, so dimensioning at 20 actually means 21 characters; but it is not possible to dimension a string of length 1 with DIM A$(0).

One-dimensional string arrays are dimensioned as DIM A$(5,10), which means that there are six strings of 11 characters each. There are no two-dimensional string arrays (though there are such numeric arrays).

There are very few string functions—only ASC, CHR$ and LEN. Since strings are dimensioned as arrays, it is possible to get characters in a string by using the subscripts. For example, if A$ is dimensioned as ten characters such that A$ = "ABCDEFGHIJ," then A$(4) would be everything to the right of the fourth character. By juggling subscripts and using the LEN, ASC and CHR$ functions, it is possible to simulate the more traditional LEFT$, MID$ and RIGHT$ functions, but it's not easy.

An input to a string is somewhat unusual in that it does not erase a string prior to inputting into it. Thus, the string may contain

More on APF's Graphics



Richard Esposito
Bertram Thiel
62 Meadow Rd.
Frostburg, MD 21532

The manual that accompanies the IM-1 is written for the person already familiar with BASIC and with the inner workings of data storage in a microcomputer. When the high-resolution mode is entered with the program statement

POKE 8194,158 : POKE 8193,60

the RAM used for screen characters becomes RAM used for defining the fine graphics characters (see Table 1) that are numbered 0 through 31.

	C1	C2	C3	C4
R0				
R1				
R2				
R3				
R4				
R5				
R6				
R7				
R8				
R9				
R10				
R11				
R12				
R13				
R14				
R15				

Table 1. High-Resolution Character.

Line 1 of the screen memory defines characters 0 and 1, line 2 for characters 2 and 3, line 3 for characters 4 and 5, and so on. The first 16 bytes of each line are for even characters, and the second 16 are for odd characters.

Memory position 512 is row 0 of character 0; position 513 is row 1 of character 0; and so forth to address 527. Memory position 528 is row 0 of character 1, position 529 is row 1 of character 1, and so on.

Reread the previous paragraphs again and visualize the screen as a map of RAM from position 512 to 1023. To define all characters you will have to poke all of these positions. This is a formidable task at first, but bear with us; it will get a little worse before the fog lifts.

Now for the final hurdle. By now, you should be asking yourself: "Just what number do I poke into these memory locations?" The answer is, "Why, the decimal equivalent of a coded-base 4 number!" We'll bet you thought you had it made once you mastered bases 2, 8 and 16. This one is easy. Each of four colors has an equivalent number that defines it: green = 0, yellow = 1, blue = 2 and red = 3. Just determine

not only the new string, but also parts of the old string. Remembering to always set the string to a null string before an input solves the problem, but it means that programs running on other systems require some translation before they will run in APF BASIC.

In other ways, APF BASIC is convenient. Variable names can be up to five characters, though only the first two are used by the machine. A variety of statements is present, including GOTO, GOSUB, RETURN, ON GOTO, ON GOSUB, IF THEN, FOR-NEXT, END, STOP, REM, DATA, READ, RESTORE, INPUT, PRINT and even PRINT USING.

The keyboard (and display) have uppercase letters only. Although there is a CTRL key, it does not generate the traditional control characters. Instead, pressing CTRL and another key generates entire keywords. For instance, the CTRL-Y combination generates the word PRINT on the screen. To cut down on program storage, keywords are stored as single characters in memory, and the CTRL key lets them be entered that way directly from the keyboard.

To allow BASIC to sense whether a key is pressed—without waiting for a carriage return—BASIC's KEY$ statement allows the program to sense not only keys on the main keyboard, but also the calculator keys on the joysticks and the joystick paddles themselves.

Cassette program storage is controlled by CLOAD and CSAVE commands, which load or store all of RAM. There are no explicit commands for maintaining data files on cassettes, but since CSAVE saves all of RAM, it saves not only the program, but also all dimensioned variables. Hence a program and its data can be saved for later reuse. Once the cassette is CLOADed, the program can be run again; the only precaution is that it must be started with a GOTO rather than a RUN statement, since RUN automatically resets all variables to 0 before starting. Several APF financial programs (such as an excellent Checkbook/Budget Manager cassette) make use of this feature to build a data set month after month.

APF BASIC can access machine-language programs or data via PEEK, POKE and CALL statements. This allows BASIC to control I/O equipment such as the cassette recorder—a very necessary function since the BASIC interpreter forgets to turn off cassette audio after a cassette is loaded, and there is enough hum in the audio system to make it objectionable. It is also easy to link to machine-language subroutines; the technical reference manual describes this procedure extremely thoroughly. A serious programmer needs that manual, since the main BASIC instruction manual's 21 pages don't explain much of the system; it only makes sense to someone who already knows some BASIC. (A BASIC Tutor program cassette is available for $49.95 to teach BASIC to a beginner.)

APF BASIC has two extensions—one for graphics, the other to play tunes via the TV set's speaker.

Music is played by a MUSIC command. Each note of three octaves is assigned a number from 1 to 7 for the middle octave, /1 to /7 for the lower octave or *1 to *7 for the

which color you want in each of the four pixels of a line of a character and solve the formula

```
C1*64 + C2*16 + C3*4 + C4
```

and you end up with a decimal number that represents the four colors in the four columns of the row of that defined character.

Here's an example. The character we wish to define is number 5, whose memory location starts at 512 + 16*5, or 592. The row of that character is number 8, whose memory position is thus 592 + 8, or 600. The colors we want are red (3) in column 1, green (0) in columns 2 and 3 and yellow (1) in column 4. Using the formula, we get

```
3*64 + 0*16 + 0*4 + 1*1 = 193
```

Thus, we would use the statement POKE 600,193 to define those four pixels. Sixteen POKE statements define the whole character if each row was different from the others, as in defining a chess piece shape or a kata kana character.

However, many or all lines are often the same so that a character definition can be simplified with a FOR-NEXT loop. Thus,

```
FOR X = 592 TO 607
POKE X,193
NEXT X
```

will define character 5 as consisting of three vertical stripes—a thin red one, a thick green one and a thin yellow one.

Applications

When you are in high-resolution graphics mode, what is displayed on the screen is addressed by RAM memory from address 0000 to 383 (base 10). That's 32 characters across the screen and 12 down (as contrasted to 16 down in alphanumerics mode). All you have to do now is poke into that section of memory the appropriate defined character. Thus,

```
10  FOR X = 0 TO 371 STEP 12
20  POKE X,5
30  NEXT X
```

will place the character we created down the left edge of the video field.

If you think you now have a handle on the high-resolution graphics, study the program in Listing 1 and try it out. Note that here we define the special graphics *before* we enter the high-resolution mode, so that you will be able to watch them being defined by your program at the top of the TV screen just before entering the second part of the program that gives a random sprinkling of the characters over the field.

Then, while still in high-resolution mode, the KEY$(X) waits for a key to be depressed; when it is, it redefines character number 4 to solid yellow. A second KEY$(X) allows escape from the high-resolution mode with POKE 8193,52 : POKE 8194,30.

If you forget to escape the high-resolution mode with these two POKE statements when you are programming and you think you have lost control, don't hit the reset or break buttons. Hit break and carefully type in the two POKE statements (you won't see them on the screen, but they are in the computer). Then hit enter and you're back to the land of reality.■

```
100  REM HI RES GRAPHICS DEMONSTRATION
110  REM ESPOSITO/THIEL DEC1979
120  REM CHAR#0 = VERT STRIPES
130  FOR I = 0 TO 15
140  POKE 512 + I, 198
150  NEXT I
160  REM CHAR#1 = SOLID GREEN
170  FOR I = 0 TO 15
180  POKE 528 + I,0
190  NEXT I
200  REM CHAR#2 = HORIZ STRIPES
210  FOR J = 0 TO 3
220  FOR I = J TO J + 12 STEP 4
230  POKE 544 + I, 85*J
240  NEXT I
250  NEXT J
260  REM CHAR#3 = SOLID RED
270  FOR I = 0 TO 15
280  POKE 560 + I,255
290  NEXT I
300  REM CHAR#4 = ZIG-ZAG
310  FOR J = 0 TO 3
320  FOR I = 0 TO 8 STEP 8
330  POKE I + J + 576,4↑J
340  POKE I + J + 580,4↑(3 – J)
350  NEXT I
360  NEXT J
370  REM SETUP FINE RES MEMORY
380  REM WITH GREEN CHAR#1
390  FOR I = 0 TO 383
400  POKE I,1
410  NEXT I
1000 REM ENTER HI RESOLUTION MODE
1010 POKE 8194,158 : POKE 8193,60
1020 REM RANDOM CHARACTER FIELD
1030 FOR I = 0 TO 383
1040 POKE I, INT(RND(0)*5)
1050 NEXT I
1060 IF KEY$(X) = "" THEN 1060
1070 REM RE-DEFINE CHAR#4 TO YELLOW
1080 FOR I = 576 TO 591
1090 POKE I,85
1100 NEXT I
1110 IF KEY$(X) = "" THEN 1110
1120 REM EXIT HI RESOLUTION MODE
1130 POKE 8193,52 : POKE 8194,30
```

Listing 1.

APF's Imagination Machine features a typewriter keyboard, two game-style controllers, a built-in cassette deck, a built-in microphone jack, a cartridge connector plus the expansion box to add on a printer, telephone modem, minifloppy disk or additional memory.

upper octave (with + or – signs for sharps and flats). Placing these symbols into a string variable or constant after the word MUSIC results in playing the indicated tune. Though the resulting monotonic music is quite plain, it does add an interesting touch to some programs. Individual tones are also used in many APF programs to provide keyboard feedback.

Low-resolution graphics are handled through five special commands. A COLOR command assigns one of eight colors to the current cursor position, while the SHAPE command can paint one of 16 shapes at that position. The cursor position is chosen by a PLOT command.

In addition, HLIN and VLIN commands can plot horizontal or vertical lines.

High-resolution graphics are also possible, but must be done through PEEKs and POKEs to the graphics RAM. It is possible to fully control this mode from BASIC, as well as PEEK into various locations to see what is on the screen, where the cursor is and so on. But APF BASIC is so slow that any such graphics would require overly long waits. To achieve reasonable graphics speeds, you really have to add machine-language routines or write everything in machine language. (Only one APF game cassette—Word Factory—is written in machine language; all the others are in BASIC and have very slow graphics movement.)

Software

The Imagination Machine can essentially run four kinds of programs—those available in a cartridge, those sold on disk or cassette and those you write yourself.

The cartridge programs are the same games (listed above) which are sold for the MP1000. All of these are written in machine language, and provide quite good graphics of the TV-game kind.

With one exception, APF cassette programs are written in BASIC. The exception is a game called Word Factory, which is a spelling game for children in the 6-9 age group. (But note that even here, the BASIC cartridge must be inserted to provide the CLOAD command needed to load the cassette.) Word Factory is interesting for kids, except that its vocabulary is necessarily limited. Since it is written in machine language and is not documented, even a customer with excellent programming ability would have trouble adding more words.

BASIC cassettes currently available include:

Music Composer/Player Piano. Stores four simple tunes which can be played, or new ones substituted.

Typing Tutor. Provides some finger exercise for the typing student and monitors his performance.

Checkbook/Budget Manager. A well-done program that provides a month-by-month checkbook balancing and budgeting function and tells you how well you are doing within each budget category.

Math Tutor. Not reviewed.

Perception I. A game where two players compete in recognizing shapes, letters and numbers.

Space, Size and Surface Guide. A program for computing areas and costs to cover them—with grass seed, tiles, paint and what have you. For the price of this program, you could buy three APF calculators, which would be more useful.

Personal Business Machine. Calculates interest rates, mortgage payments and the like. Runs very slowly and gives the wrong answers.

Budget Manager II. Keeps a detailed record of expenses in 13 different categories (such as food, rent and auto) and provides totals and summaries. Like most of the home finance programs, this one would be much more useful if a printer were available to print the results.

Artist and Easel. Used to "draw" six different still pictures on the screen and display them either one at a time or in sequence.

Bar Charts. Not reviewed.

Spelling Duel. An interesting game for kids as well as grown-ups, similar to Scrabble but played on the TV screen.

Electronic Files. Not reviewed.

Adventure Castles. Not reviewed.

Billboard/Message Center. A very interesting program which implements a new language specifically for setting up interesting video displays. A variety of commands, such as CURTAIN to create a mock curtain on the screen to cover and uncover text, makes for an interesting display and

ideal sales presentations. (With a camera and good-quality TV set, this program could make varied and eye-pleasing slides for illustrated lectures.)

BASIC Tutor. Not reviewed, but should be essential for the beginner, since the BASIC language manual supplied with the system is more of a reference manual for the experienced user.

Computer Lab. Not reviewed. Teaches basics of computers and assembly language.

Jumbled Up Things. Not reviewed.

Most APF program cassettes are either $19.95 or $29.95; the BASIC Tutor is priced at $49.95. Undoubtedly, other cassettes and cartridges will be available in the future from APF and others.

Disk programs scheduled for early release include an accounts receivable package, mailing list program, machine-language editor and assembler and a Visicalc-like program for general record keeping.

In addition to purchased software, it is, of course, also possible to write your own. Since the Imagination Machine has a fairly capable BASIC, a variety of business, entertainment and educational programs can easily be written and saved on cassette. Unfortunately, APF string usage is completely different from most other small systems, and so it would be difficult to take a program from a magazine and convert it to the Imagination Machine.

Writing BASIC programs is somewhat difficult for another reason as well. Since the display is only 32 characters wide, and the BASIC interpreter inserts a number of extra spaces around keywords, virtually every BASIC statement will wrap around the end of the line. This makes BASIC programs very hard to read. For instance, while it is possible to LIST the programs from APF cassettes on the screen, in many cases the screen will contain only two or three statements, spread out over many lines each. This is compounded by the fact that there is no way to temporarily pause while listing—thus, you never know how many lines to LIST in order to fill up, but not exceed, one screen-full. Each time you resume, you must type in a new LIST command with the required line numbers.

For this reason an external terminal or printer, connected via the RS-232C port, is almost indispensable for any kind of serious programming, whether it be in BASIC or machine language.

APF BASIC has another quirk that is sometimes awkward. When running, a program can be interrupted by pressing the break key on the keyboard. However, this key does not work while the program is looking for an input. Thus, in simple programs which have a lot of input statements in a row, it may not be possible to hit the

✓ Reader Service—see page 257

break key at just the right time for it to be recognized.

On the other hand, the Imagination Machine is very easy to program in machine language. There is a monitor in the BASIC ROM with which you can examine or modify memory; this makes the job easy. The technical reference manual has detailed programming information. Moreover, the 6800 processor is probably the easiest one to learn and use. A number of schools use it in programming courses for that reason.

The DOS

The DOS, or disk operating system, is contained inside the BASIC ROM cartridge and provides one of the simplest—and fastest—systems yet. It essentially has four commands:

INIT initializes a blank disk to format the sectors and set up an empty directory. Unlike many other disk systems, initializing a disk takes only a few seconds.

DIR displays the directory of a disk.

SAVE "file-name" saves a BASIC program on the disk. The special form SAVE "file-name" K is used to delete a file.

RUN "file-name" loads a file from disk for running.

The disk can also be used for data files, though file storage is not as convenient as on some other systems.

BASIC's OPEN and CLOSE statements open and close a disk file; up to four different files can be open at the same time.

Data files can be either sequential or random, but in both types the file is divided into 256-byte records, and a file can contain a maximum of 127 records. Thus, the maximum file size is just under 32K. Though this is an adequate size, the fact that each record is a self-contained entity and data fields cannot cross record boundaries sometimes makes packing data into a record somewhat difficult, if it is necessary to squeeze every last byte into a file.

Summary

The APF Imagination Machine is an interesting system not just for the casual programmer, but even for a person interested in the technical aspects of what makes it tick. It contains quite a few features which other systems don't have—such as the built-in cassette recorder with separate data and audio tracks. Its ability to control the audio playback and motor during program execution makes it quite useful for educational programs.

A good variety of accessories and interfaces is available to expand the system at moderate cost; certainly the System II with its expansion box, disk interface and disk drive provides a lot of equipment for its price.

Since APF provides full technical information—including complete diagrams, printed-circuit board parts layouts, timing diagrams, program listings and other information—interfacing other equipment or writing more fundamental programs is not difficult. (And for someone interested in learning about computers, the Imagination Machine could provide an excellent teaching and experimental tool.)

On the other hand, the software has some rough edges. APF BASIC has quirks which make conversion of programs difficult. It is slow and lacks many string and arithmetic functions that are often needed. Hopefully, the new Level II BASIC cartridge will be a significant improvement.

Although some of the program cassettes are quite good, many are trivial and not interesting after the first 15 minutes. Since they are expensive (the complete set of 18 costs almost $500), it becomes important to choose well and carefully. (But this is a common problem with other "appliance" computers as well.)

Perhaps the best way to summarize the system is by comparing it with some of its competition. In terms of the TRS-80, I would place it somewhere between a Level I and a Level II system. Its great advantage is color graphics and simplicity—even with a disk system—but like most systems designed for use with a color set, the color graphics are coarse and the alphanumeric display has short lines.

There is, unfortunately, no easy way to couple the Imagination Machine to a color monitor, so its graphics quality greatly depends on the TV set you use with it.

The Imagination Machine is, of course, less expensive than an Apple II. Its BASIC is somewhat better than Apple's Integer BASIC, but not as good as the optional Applesoft BASIC. In terms of future expansion, there is a much greater variety of add-ons for the Apple. When a good color monitor is used with the Apple, the graphics output looks much better.

In overall capability, the Imagination Machine compares well with the Atari 400 (which, with its optional cassette recorder, lists for about $100 more). The APF keyboard is better, but nothing can compare with the speed of Atari graphics!

Ultimately, the question arises—is the Imagination Machine a game or is it a serious computer? When you really think about it, no system which relies on the coarse detail and definition of a standard color TV can really ever be a serious computer. (Even the twice-as-expensive Apple II requires an extra black-and-white video board for serious use with programs such as Visicalc.) But taken as a game, an educational machine or a vehicle for home finance applications, the Imagination Machine stacks up well against the competition.■