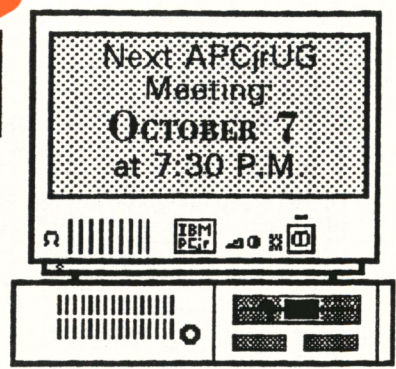


The B

NOTICE
The pages in this issue are badly out of order due to an error by the printer we didn't catch in time. Sorry. DGW

THE BENTON COUNTY PCjr USERS GROUP
Volume 1 Price: \$1.50 October, 1991



THE ASIC BASIC COMPILER

by John M. King

Do you write computer programs? Although most people don't need to learn how to program beyond writing batch files, it can be a fun experience. Writing a program can be a way to customize computing to your personal preferences. Of course, if it's just for fun, the programming must be easy.



IN THE OCTOBER, 1991 ORPHAN PEANUT:
The ASIC BASIC Compiler..... 1
Wide World of PCjr 3
A:> DIR_ 4



set up your PCjr with a ramdisk to fill the first 128K. All you get is a locked

BASIC is the obvious choice. Why? Because it is an easy language. You can write simple programs almost immediately, and you can also write complex applications. Every PCjr came with cassette BASIC built in, and many people bought Cartridge BASIC as well.

I like to write small programs. There is a sense of power and control when I write a program and watch my computer do what I have instructed it to do. I have Cartridge BASIC for my PCjr. Do I use it? No! Here's why.

BASIC PROBLEMS

There are problems with using Cartridge BASIC which make it unacceptable to me. First, it won't run if you set up your PCjr with a ramdisk to fill the first 128K. All you get is a locked
(Please turn to Page 5)



The Orphan Peanut

is the official newsletter of the *Atlanta PCjr Users Group*, a distressingly non-profit organization dedicated to the health and well-being of the first Orphan and first "clone" from IBM - the "Peanut". The Group's single purpose is as stated in Article 2 of our Constitution:

"... to provide a forum for members to share information, experiences, and techniques of use that will help other members derive maximum benefit and enjoyment from their PCjr."

The ORPHAN PEANUT is designed, laid out, and entirely created on a PCjr with 736 KB of RAM or 8 MHz of clock speed (but not both together), a Microsoft Mouse, a second floppy drive (3.5") from PC Enterprises, and two 20-megabyte hard disks from RIM via Paul Rau Consulting. There's other stuff as well, but we forget.

Software used includes *TEXIRA* word processing (ver. 6.0) and *QEdit* (2.1) text editor, Power Up! Software's *EXPRESS PUBLISHER* 2.0 desktop publishing, and sundry other programs of varying usefulness.

Permission is hereby granted for the abstraction and republication of any and all original articles in this publication, with the dual provisos that the sense of the article not be changed and that proper credit be given the author and **The ORPHAN PEANUT**.

~ OFFICERS, 1990-1991 ~

President David Blagg 475-3017

*Vice-President: Beth Geiger 872-0242

Secretary: Carol Burns 396-5675

Treasurer: Dick Anthony 992-18680

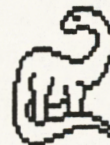
Disk Librarian: Leonard Brown 822-0113

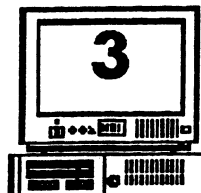
SysOp: Terry Markert 664-5056

*Newsletter Editor: David Wilson 255-2731

Landlord Stan Mislow 636-2179

* Designates Membership Committee





WIDE WORLD OF PCJR

By David Wilson

Last month, I mentioned that I had received several calls and letters from people outside the state of Georgia as a result of having been listed in PC WORLD magazine as a source of information on PCjrs and PCjr products. Well, it hasn't let up - I've received at least thirty-five calls from seventeen states, including one all the way from Hawaii. Almost all want to know the same thing: 'How can I use my PCjr to run a -----?' (fill in the blank). There are variations, but basically that's what they want to know.

Almost without exception, these folks have the same situation. They have a PCjr, they don't know how to use it, and there's no one around

them they can ask and hope to receive a sensible answer. In case you've forgotten, nobody takes the Junior seriously any more, and it's one of the most frustrating experiences imaginable to try to ask a salesperson about compatibility. If you don't get a 'cute' answer, you get a wrong one - if you get one at all. That's why, when I get one of these pleas, I've tried to answer them as honestly and completely as I can, but it takes a lot of time to compose an individual answer for each caller, so I've come up with a basic "one size fits all" answer note. I just call the canned letter up on TEXTRA, then fill in names, addresses and such for the individual, then add any specific comments I can think of. The text of the basic letter is included in this issue (see the next column for the text), and I'd welcome constructive criticism on how to improve it.



...the most important thing about a PCjr in determining just what will run on it and what won't: the amount of Random Access Memory (RAM) you have available, so I'm unable to make any specific recommendations right now. If you aren't familiar with just what RAM is and how it differs from disk drive capacity, let me just say that RAM is the electronic temporary, or working, storage area contained within any computer. Normally, the larger the RAM, the more advanced or sophisticated a program will be able to run on that computer. You can determine the amount of RAM available by noting the

numbers that appear next to the color bar when you start, or "boot", your PCjr. If your Junior hasn't been expanded, you'll most likely see "128 KB" as the last set of numbers before it goes on to other things - meaning that you have 128 kilobytes of RAM available to run your programs. This is the most severe limitation on what you can load and run, and if you don't have the RAM, you can't run the program. You can normally tell from the box that software comes in how much RAM is required to operate that program.

When the PCjr was introduced in 1984, 128 KB of RAM was considered a good bit, but today's newest computers can have one hundred times that amount without being considered outrageously "overpopulated", and they run at speeds that are almost as greatly improved. To me, the most astounding thing about this is that they still use much of the same software that will run on your PCjr. In fact, I saw somewhere



The ASIC BASIC Compiler

(Continued from Page 5)
it's own just like COM and EXE files.

What about those five strikes? How do compiled BASIC programs stack up? First, they run with or without a ram-disk. Problem number one is eliminated. No interpreter is needed to run a compiled program, so disk space and loading time are saved.

Furthermore, you can run the program whether you have BASIC or not.

Finally, compiled programs run much faster than interpreted BASIC programs.

No strikes here. It looks like compiled BASIC is a real winner.

Are there any negatives for compiled BASIC? Well, for one, it's not free with DOS. The Microsoft BASIC compiler costs three or four hundred dollars.

Microsoft Quick Basic, which is also a compiler, is more reasonable. The discount price is around \$100, but that's still quite a bit of money just to try programming for fun.

THE ASIC COMPILER

All of this leads to ASIC. ASIC is a shareware BASIC compiler. It's still new, and it doesn't have the power of

the other BASIC programs yet, but it can do a lot. The important point is that it is a compiler. ASIC produces COM files directly.

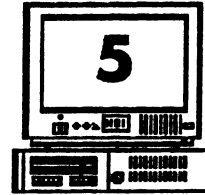
I'm using ASIC to write a small program to control my Panasonic KX-P1124 printer. Although the Panasonic has excellent front panel controls, some functions are awkward or available only through software. My personal printer program will make it easy to set the functions which I find most useful. There you have it, "Customized Computing through Programming". I also plan to use this program as the basis for another article on ASIC.

Version 1.0 of ASIC was released in March, 1990. The author of ASIC is David Visti who can be reached at this address.

David Visti
P.O. Box 2952
Raleigh NC 27602-2952

The ASIC package includes the compiler, an editor, and over fifty pages of documentation on disk. Everything fits comfortably on one 360K disk with

(Please Turn to Page 7)



The ASIC BASIC Compiler

(Continued from Page 1)

up machine. Since everything else runs significantly slower without the ramdisk, that's a big strike against Cartridge BASIC. Two boot disks aren't the answer either. I want the programs I write to work with the others I use.

You can get around the ramdisk problem. If you have the cartridge, you can rename BASICA.COM from your DOS disk and run it. Otherwise, you'll have to patch BASICA.COM, so that it will run without the cartridge. However, you must now load BASICA.COM from disk before you can run a program you have written.

The DOS 2.1 BASICA.COM is 26,112 bytes. That's a lot bigger than most of the programs I'd want to write. If you use GW-BASIC, the MS DOS 3.3 version is 80,592 bytes. That's even worse. The large BASIC program takes up too much disk space and loads too slowly, but you can't run a program you have written without loading it first. Two more strikes against BASIC.

If three strikes aren't enough, here's a fourth. If you share your programs, the other users must have BASIC. Furthermore, there are enough

differences between BASICA and GW-BASIC and between the DOS 2.1 and DOS 3.3 versions that your program may not run correctly if they don't have the same BASIC you have.

Finally, if you do use Cartridge BASIC, BASICA, or GW-BASIC your programs will run slowly. I mean really slowly. To do anything will take three to four times longer than you are used to with other programs.

This slowness is a characteristic of "interpreted" BASIC. Cartridge BASIC, BASICA, and GW-BASIC are interpreters. They must be loaded to interpret the BASIC program you have written. The interpreting process is slow, and that's strike five.

COMPILED BASIC

There is an alternative to "interpreted" BASIC, it's called "compiled" BASIC. Microsoft Quick Basic is a compiler. You write your program in BASIC, but the compiler generates a standard COM or EXE file. You don't have to load the compiler to run the program. It runs on

(Please turn to Page 6)
(Continued from Page 5)



Volume 4, Number 4
October, 1991

The Orphan Peanut

TEXT OF REPLY TO INQUIRIES BY
PCJR OWNERS

recently that the PCjr will actually run more different programs than any other computer made by IBM; that's because it'll still run all the stuff that was written for a PCjr (which most other computers won't run), as well as most programs that have been written for other IBM PC, PC/XT, and PC/AT class machines. That includes almost all the "clones", but not the new PS/2-specific software from IBM and others. Depending on how much RAM you have, you should be able to find a program at a price you want to pay that'll do most anything an average person will need to do. The two things you most need to be aware of are whether or not it says on the box that it'll run on a PC, a PC/XT, or a PC AT, and how much RAM you need. If the software passes these tests, it will very likely work on your Junior. It may do it slowly compared to the new machines, but it'll get done. You just have to select it as carefully as you can.

I wouldn't be treating you fairly if I didn't mention here that, from the little I've seen of educational software these days (my youngest is 22, so I haven't been exposed to it lately), I don't think you'll find much in a computer store that'll work on a Junior with less than 256KB, and very little of that. Games and sophisticated business software will also use a good bit of RAM, so if that's a problem, you'll probably want to consider "upgrading" your RAM capacity to 384KB or more - my personal recommendation is that you add 512KB to get you to a total of 640KB. Then you won't have to worry about whether most things will work - they will.

That brings us to another subject: upgrading your hardware. I don't know, of course, just

how much you're interested in spending on an old computer, but I'm going to assume that, like most Junior owners, you want to keep it down to a minimum. In that case, I suggest that you order a memory expansion called the Jr HotShot from a company whose address I'll give later. The last price I saw for them was around \$140.00 plus shipping, for an upgrade of 512KB. You can get a 256KB upgrade for around \$40-\$50 less, but I'd strongly urge you to go ahead and get the full amount if you can. This unit, like almost all PCjr expansions, is installable by you with only a screwdriver and an ability to read directions. I did it, and if I can, anyone who isn't seriously handicapped can, believe me. This single improvement will enhance your computer's ability to perform more than any other one thing. I can't emphasize it too strongly.

There are a number of things you can do that will help. If you want to use a printer, you're going to need a Parallel Port sidecar and you'll need a printer; you may want a second floppy disk drive - a huge convenience, but not always necessary. You may even want a hard disk, but that'll be something to decide later. I tell people to get the memory - that's essential - then take your time and see what you really need to do what you really want to do. I have a very sophisticated and fancy setup, but I've been building it for more than five years and really wouldn't recommend that anyone try to duplicate it today unless they get a kick out of "tweaking" a PCjr to make it do things no one thought it could do a few years ago. Taking Junior to all the heights it can reach these days is much like restoring a classic car - you do it for the love of it, not because of its performance relative to a new one.

(Com

RE

PR
YR

LO
GO
GO

RO
RE
IF P
RE

EN

This
520
CO
is 3
spac
the
on c
ASI
mor
gran
80K

Som
and
of la



The ASIC BASIC Compiler

(continued from Page 8)

Print your name

```
PRINT "Your name"; INPUT  
YRNAME$
```

```
PRINT YRNAME$  
SUB ROWCHECK:  
TO LOOP:
```

```
ROWCHECK: ROW = CRSLIN  
Note CRSLIN not CSRLIN  
IF ROW > 20 THEN END:  
RETURN
```

```
END
```

A short ASIC program compiles to a small byte COM file. (The minimum COM file which can be created by ASIC is 15 bytes consisting of code and data reserved by the compiler.) Since the minimum disk space to store any file is 1K (2K on a hard disk), an ASIC COM file may not take up any more storage space than a BASIC program, and you won't need the 26K to store the BASIC interpreter on the disk.

One of the differences between ASIC and BASICA code are striking. The use of labels (names that end in a colon), in-

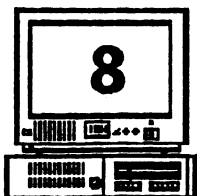
dentation, and blank lines makes the ASIC code much easier to read. For example, the GOSUB and GOTO commands specify a label as the destination. This provides an immediate idea of the purpose without REM statements.

A less obvious difference is that in ASIC INPUT cannot send out its own prompt text. A separate PRINT statement must precede INPUT if you want a prompt. This is certainly done to make compiling easier. In general, ASIC functions are less complex than those of BASICA or GW-BASIC. I'm sure David will add the bells and whistles later if enough of us support him with registrations.

There are other subtle differences between ASIC and other BASICs that can cause problems until you get used to them. Note that CRSLIN is spelled differently although it does the same thing as CSRLIN. Also, the IF ROW > 20 must be followed by THEN and a label, not a command.

Of course, what would version 1.0 of any program be without a bug or two. Although I've found none that are serious, here's one. The PRINT statement doesn't execute the carriage

(Please turn to Page 10)



Volume 4, Number 4,
October, 1991

The Orphan Peanut

The ASIC BASIC Compiler

(Continued from Page 7)
languages. We'll see more of what this looks like in an example.

WHAT'S MISSING

This is version 1.0 of ASIC and there is a lot in BASICA that's not here yet. Only integer (whole number) arithmetic is available. That's no problem for writing printer control programs, but you won't want to do your tax program in ASIC yet. There also are no higher math functions such as trigonometric functions and exponentials.



Although you can do a lot with the screen, there are no advanced commands such as CIRCLE, DRAW, LINE, and PAINT. If you want to do these functions, you must do them stepwise using the simpler statements.

Hard disk subdirectory commands for your programs are not yet implemented.

There are no CHDIR, MKDIR, RMDIR, or FILES statements at present.

Finally, only disk file errors can be tested directly. Printer errors, such as

Out of Paper, simply result in an exit to DOS.

AN ASIC PROGRAM

Just about everybody's first BASIC program is to make the computer print the name on the screen. Here's one in BASICA that's a little more advanced. This won't scroll your name forever, only until it reaches row 20 on the screen. There is a subroutine which uses CSRLIN to find out what row the cursor is on. If it is below 20, the program ends. If you start on line 19 or below, your name will be printed only once.

```
10 REM Print your name
20 INPUT "Your name"; YRNAME$
30 PRINT YRNAME$
40 GOSUB 100
50 GOTO 30
60 END 100 ROW = CSRLIN
110 IF ROW > 20 GOTO 60
120 RETURN
```

When saved to disk, this BASIC program is 133 bytes long. The same program looks like this in ASIC:

(Please turn to page 9)

The ASIC BASIC Compiler



(Continued from Page 6)

or 100K free for programs you are writing. The compiler, ASICC.EXE compiles code written in ASCII by an editor or word processor. The editor allows you to move directly from the editor to the compiler and

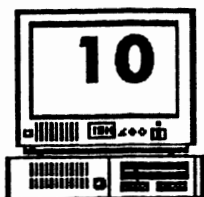
combined editor-compiler environment requires 400K of free RAM and can compile a program of up to 809 lines. When run separately, the editor and compiler each run in 320K of free memory. When run as a stand-alone compiler, ASICC has no line limit and can generate up to a 64K COM file. That might represent compiling over 3,000 lines of code, or more than 50 pages. Enough to handle my projects many times over.

ASIC is available on many BBSs, including the **PCJR WORKSHOP** as well as from most shareware disk vendors. Registration is only \$10 which I will only send to David after confirming that ASIC will do enough of what I need.

The 41 BASIC statements which the ASIC compiler handles are sufficient for many useful programs. You can control visible and audible output with commands to set colors, move the cursor or determine its position, print to the screen or printer, and control sound. In graphics mode, you can turn pixels on and off. You can obtain input from the keyboard or disk files and write to disk. Within a program you can do nested FOR/NEXT loops, IF/THEN/ELSE tests (for equal to, greater than or less than), and GOTO or GOSUB. For the more advanced there are PEEK, POKE, VARPTR, MID\$, LEN, ABS, random numbers, and arrays. You can move between string (text) and integer (numerical) variables with ASC, CHR\$, STR\$, and VAL. String variables can be up to 80 characters, and integers can range from -32,768 to +32,767.

Most of the statements are very similar to BASICA and GW-BASIC although there are minor differences in syntax. One big difference is that ASIC is a modern version of BASIC and uses labels instead of line numbers. This is a major improvement and brings ASIC a step closer to the "power" programming

(Please turn to Page 8)



Volume 4, Number 4
October, 1991

The Orphan Peanut

A > DIR

by David Blagg, President, APCjrUG

There are some concerns I have which I want to share with all of you. They are important enough, I think, to ask that you read this to the end, then begin your reflections.

Our user group is approaching an opportunity for renewal of its spirit, its reason for being, and the activities which give expression to that spirit. I am referring to the election of officers. This is nothing new, for it is now the fourth year in which we have, or will, elect officers as provided in our constitution. And that leads to my major concern.

The group of people who presently serve as officers is little changed from the group first installed in 1988. To be sure, we split the Secretary-Treasurer office into two offices and established appointed offices, a couple people have come and gone, and a couple people have gone from one office to another (myself included). But for the most part the group has not changed much. And we are tired.

In an organization such as this one, member involvement is a key element to continued success. We come together for one reason: to enjoy the PCjr and to learn more about its many capabilities. Hence, the search for meeting topics is constant, and it is reasonable to expect that those topics should address member needs. But how do we find out what those needs are? And who will

take the time and effort to ferret out a response to those needs? How do we as an organization maintain ourselves and the resources we share? Who will keep track of what we have done, and who will take the time to collect our dues and pay our bills? And what about the lifeblood of any organization, inquisitive folks and new members? Who will sell us to them? The list is not endless, and it is not even very long. But someone has to do these things, or we will not amount to much as a user group.

Indeed, as a user group the APCjrUG is fortunate to have both a newsletter and a telecom bulletin board system. We get national attention for these activities. But these, too, require sustained time and effort. And who will make the necessary commitments which, if not forthcoming, means there simply will not be a BBS or a newsletter or both?

The officers met, as usual, before the last regular meeting. We discussed membership involvement in the APCjrUG. Among other things, we decided to approach you, the members, in a concentrated effort to assess your needs and willingness to get involved, especially in officer roles.

Simply put, the question is this: *What do you want to do as a member of the APCjrUG?* Please let us know.

TEXT OF REPLY TO INQUIRIES FROM PCJR OWNERS...

(Continued from Page 4)

Software, now, is a whole 'nother ballgame. The Atlanta PCjr Users Group doesn't maintain a "compatibility list" of programs that will run on a PCjr, but there's an excellent publication that does. It's called jr Newsletter, and it's the last remaining of several national PCjr publications. It's also probably the best of the bunch, which most likely explains its survival. It's a monthly newsletter with articles on every aspect of the PCjr, from "how-to" to program reviews, and it's all written on and for PCjrs. I can't recommend it too highly - next to membership in the APCjrUG, it's the best value in the PCjr world. You can write for information or send \$23.00 for a subscription to: jr Newsletter, P.O. Box 163, Southbury, CT 06488.

There are several Software Libraries scattered around the country which sell a class of software called Shareware. Basically, what this is, is "try before you buy". You get the program you're interested in, try it for a month or so, then either erase the disk or send the author his requested "donation". This is by far the least expensive way to build up a software library, but the quality of the programs is frequently not as high as what we refer to as the "commercial" product. These Software Libraries will provide you with programs you select from their catalog at a fixed price per disk (usually from \$2.00 to \$6.00), and after you try the program you are responsible for compensating the author. You can get the addresses of many of these Libraries from the advertisements in national PC publications, so I won't attempt to list them here. "Commercial" software is what you buy at retail prices in a store. Prices vary, depending on what store you buy the product in, so do some comparison shopping before you pay too much. Don't let the

salesperson talk you into getting the most expensive program, either - you should try to have an idea of what you want before you plank down your money. Also, let me advise you to be sure of what the return policy is before you buy. Many retailers won't accept a software package after it has been opened, although I've never figured out how you're supposed to know whether you like it or not without opening it. Several chains have a very helpful and liberal returns policy, so check around before you buy. My personal preference is Egghead Software, so if you have one of their outlets near you, you can try them. Egghead will also match anyone's price for the identical product, so you might watch the ads in your local newspaper. There are others as good, however, so don't get stuck in a rut - try 'em all.

The ASIC BASIC Compiler

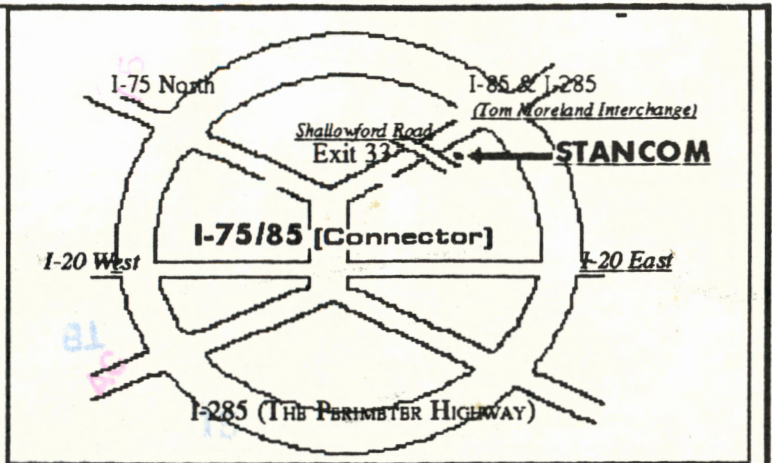
(Continued from Page 9)

return and line feed quite correctly. If you ran this example, you'd find the first letter of your name missing on the last line. A simple work around until the bug is fixed is to print a space in front of the name with `PRINT " ";`. The semicolon stops the carriage return and just moves the name over one space where it prints perfectly.

This is the first of a series of articles by the renowned PCjr authority and long-time president of the San Francisco PCjr Users Group. We'll continue next month. Ye Ed.

GETTING THERE

If you're outside the Perimeter (I-285): Just follow I-285 around Atlanta until you reach the *Tom Moreland Interchange (I-285 and I-85 North)*. Go South on I-85 toward Atlanta until you reach *Exit 33, Shallowford Road*. Exit onto Shallowford Road and turn left to cross over I-85. Immediately past the traffic signal on the *East* side of



the overpass you will see a building on the left with *STANCOM* in large letters on the front of it.

If you're inside the Perimeter (I-285): Get on I-85 heading North from the city. Watch for the *Shallowford Road exit, Number 33*, and leave the Interstate highway there. Turn right on Shallowford Road, get in the left lane and prepare to immediately turn left into the *STANCOM* parking lot.

The Orphan Peanut

The Cronicle of the Atlanta PCjr Users Group
6575 Amberglades Lane
Atlanta, GA 30328



New Frontier...
Collect Stamp

