

Instruction book for the UltiMulti Cartridge

For all cartridges released by Bally or Astrocade (Astrovision) for the Arcade unit, and some that have not been released, the instructions are posted on the website www.ballyalley.com

The current list of those found there is as follows:

280 ZZZAP / Dodgem	Basic Checkers
Artillery Duel	Old Bent Nose
Astro Battle (Space Invaders)	Quadra
Bally Pin	
Biorhythm	
Blackjack / Acey Deucey / Poker	Bowling
Brickyard / Clowns	Chicken
Cosmic Raiders	Blast Droids
Dog Patch	Coloring Book
Football	Conan
Galactic Invasion	Mazeman
Grand Prix / Demolition Derby	Music Maker
Incredible Wizard	Soccer / Shoot Out
LetterMatch/SPELL&Score/Crosswords	Songs
Life (Degler)	Tests
Ms. Candyman	Video Story Book
Panzer Attack / Red Baron	
Pirate's Chase	
Sea Devil	Astro BASIC
Sea Wolf / Missile	Bally BASIC
Solar Conqueror	Blue Ram BASIC 1.1
Space Fortress	Color BASIC
Speed Math / Bingo Math	
Star Battle	
Tornado Baseball/Tennis/Handball/Hockey	
Treasure Cove	

Hot Rod Bally BASIC and ViperSoft BASIC languages are mostly the same as Blue Ram BASIC 1.1 (BRB). The exceptions are the input / output instructions. The RPL function and OP command are only in BRB 1.1. The other main

differences between the BASICS is where the program area resides and where the variables are stored.

There are several multi-load programs in the cartridge. They are:

C-8 Line Art (by Stanley Kendall)
Pro Golf (by Henry Sopko)
And Quadra (by Mike White)

These all have to be loaded with either 2 or 3 loads
To load C-8 Line art set the switches to O1O1OOO1
Then press RESET
Next set the switches to OO11OOO1
Then press the Number 2 (using the BASIC overlay – 5 without)
Then set the switches to Blue Ram BASIC 1.1
OOO11111 and press 1 (4 without BASIC overlay)

The switch settings are always shown from left to right when the cartridge is mounted in your Bally / Astrocade Arcade unit.
The O is the OFF position and the 1 is for the ON position.
Check your switch bank to see which way is ON.

The same technique applies to Pro Golf (see switch allocation sheet) and Quadra. The exception is that there is one more load for Quadra.

Most other Astro BASIC games have their instructions included in the listing of the program in the Arcadian articles listed on www.ballyalley.com Example of these is the listing and instructions for Backgammon and Obstacle Course Tournament. They are listed under type in games ... Wavemakers.

The Blue Ram BASIC 1.1 programs created by WaveMakers have their instructions there also.

Remember these 3 settings because you will use them for ALL BASIC programs:

ASTRO BASIC ALL SWITCHES **ON**
Blue Ram BASIC 1.1... Switches 1,2 & 3 are **OFF** and all the rest are **ON**
ViperSoft BASIC Switches 1 & 2 are **OFF** and all the rest are **ON**

Remember that ViperSoft Basic must be loaded before you load its programs. Astro and Blue Ram Basics get loaded AFTER the program.

Blue Ram Super-Extended Basic (1.0 & 1.1)

Introduction: BLUE RAM Super Extended Basic is the direct result of the efforts of Jay Fenton and Perkins Engineering. It contains virtually all of the features of Bally BASIC, plus many, many more. These instructions are not intended to teach programming or cover the Bally BASIC features. Rather, it is brief description of the additional features provided by this language.

General Improvements: These are the many improvements which do not directly reflect in the language, such as, faster program execution. Others take the form of new commands, new syntax, or new variations on old commands. A list of general improvements follows:

- Additional graphics commands
- Boolean operations
- Changeable print number base
- Eight mode flags
- Faster overall execution
- Four color screen
- Four new Data types
- Full sound effects Driver
- Larger program area
- More versatile math forms
- Program "bomb" recovery
- Trace program debug aid
- Two character fonts
- User extensibility
- Versatile program editor
- Windowed graphics and text
- Tape interface (2000 BAUD)
- Faster multiply / divide
- Parallel Printer Driver *
- Built-in keyboard driver *

* Requires extra products no longer available.

General Operations: To operate this cartridge, the Blue Ram must be connected and the switches placed in the 6K Range and the RAM or AUTO mode. The Blue Ram keyboard* will operate if connected. In either case, the Bally keypad will work. When using the keyboard the keys have the following effect:

- BREAK Halt operations
- ESC Halt operations
- Left Blank NEXT

- LINE FEED GO+10
- TAB INPUT
- RIGHT BLANK CIRCLE

In addition, the following new COMMANDS are available

- DA. DATA
- CI. CIRCLE
- SN. SNAP
- PO. POINT
- DE. DEFAULT
- SH. SHOW
- SC. SCROLL
- ZE. ZERO

➤ All key words may be entered using the shifted letter, the control key, it may be spelled out (e.g. S C R O L L), or it may be abbreviated (e.g. S C).

A PROGRAM "Bomb" recovery procedure has been implemented. If the program should bomb (garbage on the screen and / or keyboard / keypad lockup) for some reason, you may recover by pressing and holding RESET, pressing and holding the + key, releasing the RESET key, then releasing the + key.

A total of 3100 bytes are available for programs, strings, machine language routines, etc., beginning at %(24576). For more information see ARCADIAN Vol. 5, #12, Page 179.

By holding down the up arrow (↑) (LIST) key on the keypad, a running program will be traced, line by line, on the screen.

New variations on old commands : Several commands from the Bally Basic now have different parameters associated with them.

2000 Baud commands.

Requires cable(s) connected from Blue Ram unit to a cassette recorder.

:PRINT_	Dumps program and string memory to tape.
:PRINT %(aaaaa),nnnn	Dumps nnnn words (2x nnnn bytes) beginning at address aaaaa to tape.
:INPUT	Loads program and string memory from tape.
:INPUT %(aaaaa)	Loads a block of data from tape into memory beginning at the address aaaaa.
:LIST	Check-reads a 2000 BAUD tape.

300 Baud commands.

Requires cable(s) connected from a Bally cassette interface (separate) unit to a cassette recorder.

:PRINT 300	Arms the Bally BASIC serial port to dump to tape at 300 BAUD
:INPUT 300	Arms the Bally BASIC serial port to load from tape at 300 BAUD
:LIST 300	Reads the Bally BASIC serial from tape to screen for visual verification.

NOTE: Blue Ram BASIC 1.1 is expanded to accommodate both the 16 K and the 32K models.
The 4K unit does NOT have the 2000 BAUD interfaced built in.

***** *COMMANDS new and old* *****

GOSUB 1111,v,n1,n2,...
GOTO 1111,v,n1,n2,...

Similar to the standard GOSUB and GOTO commands except that then v+1 with n2, etc. It could also be done with:
DATA v,n1,n2,...;GOSUB 1111 (o GOTO 1111)

BOX x,y,w,h,m

The value for the mode m have increased to the following values:

0 – Nothing	4 – Overlay BC
1 – XOR FA	5 – Overlay FA
2 – XOR FB	6 – Overlay FB
3 – XOR FC	7 – Overlay FC

XOR are reversible type modes
Where: BC = Background Color
FA = Foreground Color A
FB = Foreground Color B
FC = Foreground Color C

PX (x,y)

This operand now determines the color of the pixel at x,y. It's responses can be: 0 = BC 1 = FA 2 = FB 3 = FC

NEW COMMANDS:

The following new commands have been implemented.

CIRCLE x,y,r,m

Draws a circle of radius r at x,y using mode m.

DATA v,n1,n2,...

Loads a succession of variables beginning with v with the trailing operands n1,n2,...(IE, DATA A,20,5,4,33 sets A=20, B=5,C=4 and D=33)

DEFAULT

Sets ALL two-letter variables to their preset values as follows:

Graphic Boundaries

X Right
X Left
Y Top
Y Bottom
X Y screen location

X
NT
Y

BC 239	CC 7 (Character Color)
FA 165	LC 0 (Last Character)
FB 91	CF LARGE (Character Font size)
FC 233	NB 10 (Number Base)
XR 80	CR 80 (Character Right boundary)
XL -79	CL -79 (Character Left boundary)
YT 51	CT 51 (Character Top boundary)
YB -48	CB -48 (Character Bottom boundary)
XY 0	
RM 0	(ReMainer)
screen location	CX -79
	3 Note Timer
screen location	CY 51

OP

Volume 6, Page 47

This is a user extensibility command. When the BASIC language For more on the OP encounters this command for execution, a machine language branch Command see: ARCADIAN. (jump) is taken to a user provided interpreter routine via a jump vector at address 6DCCH (!6DCC)

PLAY %(aaaaa)

For more on the PLAY Command see: ARCADIAN Volume 5, Pages 128 & 156

Plays a string in the background mode while the program continues. With the proper sound string (at %(aaaaa)) this can play three part harmony, explosions, or any other sound effects that the Bally can make

POINT x,y,m

Same as BOX,x,y,1,1,m

RPL 1111/oldtext/newtext

Replaces the first instance of oldtext with newtext in line 1111. (I.E.:
1111 GOTO 455;GOTO 444
RPL 1111/GOTO 4/GOSUB 5
The new line 1111 now looks like this:
1111 GOSUB 555;GOTO 444

RPL 1111/nnnn

Renumbers line 1111 to nnnn and re-sequences it to it's proper position as line nnnn, and line 1111 will no longer exist.

SCROLL x,y,w,h,n
SHOW x,y,sm,loc

Scrolls a window w,h at x,y n lines up or -n lines down.
Displays a previously SNAPPED field at x,y from memory loc using the showmode sm:
0=Overlay, 1=OR, 2=XOR, 3=blank.

SNAP x,y,w,h,loc

Copies a multi-color field w,h at x,y into memory at loc for a subsequent SHOW. The amount of memory needed is: $((w \div 4 + (RM\#0)) \times h) + 4$.

Examples:

$6,4 = (6 \div 4 (1) + (RM\#0)) (1) \times h (4) + 4 [1+1=2 \times 4=8 +4] =12$

$19,15 = (19 \div 4 (4) + (RM\#0)) (1) \times h (15) + 4 [4+1=5 \times 15=75 +4] =79$

$20,5 = (20 \div 4 (5) + (RM\#0)) (0) \times h (5) + 4 [4+0=4 \times 5=20 +4] =24$

ZERO

Sets all ONE-letter variables (A to Z) to 0.

1111 (edit)(edit)

the edit key (PAUSE on the keypad, CNTL E on the keyboard) is used to step through an existing program line 1111 one character at a time. Characters may be deleted or new ones inserted as you go. When the end of the line occurs, the line will have been changed to reflect what remains on the screen. This works exactly the same as the ASTROBASIC line editor.

NEW DATA TYPES:

Four new data types have been added.

BYTE (v,b)

Accesses a single byte of a variable v. b is 0 for the lower byte and 1 for the upper byte.

!

When an exclamation point precedes a number, that number is taken as hexadecimal. I.E. !6000 = 6000 Hex or 24576 decimal.

>

Provides the address of the line number immediately following the greater than symbol.

←

Provides the address of the variable immediately following the left arrow.

NEW OPERATORS:

There are five new operators that have been provided as follows:

-

The negative sign negates the value immediately following it: $(5 \times -7 = -35)$.

↑

The up-arrow is the Boolean operator AND ($7 \uparrow 5 = 5$)

↓

The down-arrow is the Boolean operator OR ($9 \downarrow 7 = 15$)

→

The right-arrow causes the *preceding* value to be shifted *right (sign extending)* the number of places in the *following* term or *left (circularly)* that many places if the following term is negative. ($24 \rightarrow 2 = 6$) ($8 \rightarrow -3 = 64$)

←

The left-arrow is the Boolean operator XOR ($3 \leftarrow 5 = 6$). BE CAREFUL

HERE to follow with a number. Following the left arrow with a letter (like in a calculated number [e.g: $8 \leftarrow A + 5$] will give you 8 XORed with the address of the A variable 27818 (I6CAA) and then it will add 5 for a total of 27823, when what you really wanted is $8 \leftarrow (A + 5)$ which will give you the XOR of 8 and what is in the variable A with 5 added to the A sum.

NEW VARIABLES:

Fourteen new two-letter variables have been provided.

CF

Character Font. This variable is set to LARGE (LA.) for the standard 5x7 Character Font or SMALL (SM.) for the new 3x5 set.

CC

Character Color. This variable sets the mode of character screen printing. It's values are the same as the mode values of the BOX command.

LC

Last Character. Contains the ASCII value of the last character printed.

CL

Character Left.

The text printed on the screen is constrained to a window controlled by the variables CL, CR, CT and CB. These are preset to full screen

CR

Character Right.

CT

Character Top.

XL

Graphic X Left.

Similarly, the graphics symbols are also constrained to a window as controlled by the variables XL, XR, YT and YB.

XR

Graphic X Right.

YT

Graphic Y Top

These are set to full screen also.

<p>YB FA FB</p>	<p>Graphic Y Bottom Foreground Color A Foreground Color B</p>	<p>Since there are four simultaneous colors available on the screen at once, these two new variables round out the set. (with BC and FC).</p>
<p>NB</p>	<p>Number Base</p>	<p>This variable controls the number base in which numeric values are printed. It is normally set to 10 (decimal) but it can be set to 2 for binary, 8 for octal, 16 for hexadecimal, etc.</p>

MODE FLAGS

The upper 8 bits of the note timer (NT) have been implemented as mode flags since only the lower byte is used as the note interval time. Each bit, when set, has it's own meaning, as follows:

<p><i>BIT 7</i></p>	<p>This bit is used by the program to indicate when a full keyboard* scan is to be done as opposed to only a scan of the BREAK and ESC keys. You can generally ignore this bit.</p>
<p><i>BIT 6</i></p>	<p>When this bit is set the regular Arcade background processor will operate off of the screen interrupt. The PLAY makes use of this bit. Also, there are some counters and timers supported by this processor.</p>
<p><i>BIT 5</i></p>	<p>When this bit is set by the programmer it informs the system that another <i>user-defined</i> background processor has been established. A <i>call</i> will be made to this processor from the screen interrupt via the vector at 6DCFH.</p>
<p><i>BIT 4</i></p>	<p>When set, this bit disables screen printing as well as the associated sound.</p>
<p><i>BIT 3</i></p>	<p>Sends to the printer via MODEM interface* all text meant for the screen.</p>
<p><i>BIT 2</i></p>	<p>This bit set, along with some additional software linked to vector 6DD8H, will call that software with the ASCII character code in A for all characters meant for the screen.</p>
<p><i>BIT 1</i></p>	<p>With this bit set, the printer will print lower case characters as opposed to token "words" such as the one character commands. Example "t" instead of "PRINT".</p>
<p><i>BIT 0</i></p>	<p>This bit disables the CTRL "words" and makes the keyboard* yield the actual ASCII CNTL characters.</p>

SAMPLE PROGRAM
STATEMENTS

The following program is provided as a basis for experimentation. Try modifying the lines to see the effects.

```
10 CLEAR;BOX 0,0,15,15,6;CIRCLE -5,3,5,7; CIRCLE 5,3,5,7; CIRCLE 0,-5,5,7; CIRCLE 0,0,12,5;  
    SNAP 0,0,24,24,@(0);DRAW THREE COLOR PATTERN  
20 FOR N=0TO 50;SHOW RND (148) -74,RND (88)-44,0,@(0);NEXT N;. PUT 'EM ALL OVER  
30 FOR N=1TO 50;SCROLL 0,N-25,N,N,25-N;NEXT N;. SCROLL CENTER OF SCREEN  
40 FOR X=80TO -79STEP -1;F=(80-X)÷5;FOR D=0TO 20STEP 10;SHOW X÷D,Y,0.>(100÷RM);  
    NEXT D;FOR N=0TO 20;NEXT N;. NOTE THAT LINE REQUIRES THAT LINES 100  
    THROUGH 104 TO BE ENTERED AND "POKED" WITH THE DATA STATEMENTS  
    BELOW  
50 GOTO 10  
100 ABCDEFGHIJKLMNOPQRSTU; ALPHAS ARE SPACE RESERVERS FOR POKES  
101 ABCDEFGHIJKLMNOPQRSTU  
102 ABCDEFGHIJKLMNOPQRSTU  
103 ABCDEFGHIJKLMNOPQRSTU  
104 ABCDEFGHIJKLMNOPQRSTU
```

Now type in:

```
DATA >100,8,9,0,12291,-4096,-4096,4093,-16384,12291,3276,15408  
DATA >101,8,9,0,12291,-4096,-4096,4093,-16384,12291,3084,12348  
DATA >102,8,9,12291,-4096,-4096,4093,-16384,-16381,5363,3075,15  
DATA >103,8,9,12291,-4096,-4096,4093,-16384,-16381,-4096,-16384,-16381  
DATA >104,8,9,12291,-4096,-4096,4093,-16369,-16372,12300,12348,-4096
```

NOTE: Once these data have been poked into lines 100 – 104, these lines cannot be edited since they are no longer printable characters. They essentially represent the values stored as a result of a SNAP command. For example, with the proper picture on the screen, SNAP 0,0,8,9,>100 would have the same effect as the first DATA statement. The advantage of SNAPPING pictures into lines of the program is that it will not change when you edit the program as it would when it is stored in the @() string . SNAPs can also be "poked" (SN 0,0,8,9,%(!7000)

PRINTER COMMANDS:

NT=!1800;LIST ;NT=3 (lists the program to the printer via the parallel printer port in the MODEM interface.)*

NT=!1A00 (enables the printer for printing lower case characters instead of token "words")

* Requires components no longer available. These are here only for those who have these components

One final note, there is no *(n) string available in BLUE RAM BASIC 1.1. If you are using strings, you may want to use the PEEK & POKE method. I.E. %(nnnnn)= and =%(nnnnn) instead. You can prevent overwriting your saved POKES by setting to the beginning and end of program area variables. Then you can still PEEK & POKE to addresses outside of the program area. This way your saved data will not be overwritten as you change the length of you BASIC program.

O1111110 Press RESET for the menu Select game 2, set switches to 00011111 then press the BASIC “=” key

Whatzit?

Ken Lill

Whatzit? Is a fast paced one player game. The object is to change all items from whatever color they are to white. Watch the start of the level for the way this is done. Each level has a different pattern. Each 2500 points gives you another life. Each round gets tougher. Stay away from the Chaser. If you and hit occupy the same space, you loose a life. Just when you think you've got a pattern going along comes the Changling. Note it's color. It will change all items to its own color. You will have to change all items to white to finish any round. You can stop a Changling from changing the items by landing on it or taking a ride on the magic carpet located off the grid on the right. If you go outside of the grid anywhere but onto the magic carpet you will lose a life. Using the Magic carpet also re-starts the Chaser. You have a limited number of carpets, depending on the level / round.

O1000001 Press RESET For the menu Select game 1, set switches to 00011111 then press the BASIC “=” key

Snake Snack

Ken Lill

This game is similar to “Caterpillar” except that you have a maze that you are going through. When you chomp up a bit, you get longer. It IS possible to get all bits without dying. You die when tou either back over yourself or you try to cross yourself.

00000001 Press RESET, set switches to 00011111 then press the BASIC “=” key

Spring Thing Pt. 1

Ken Lill

Another fast paced game. It is a 4 player game with separate options. They are selected by KN(1) and TR(1) in the setup screen in the beginning. The object is to collect as many gold rings as you can. The only problem is that you are a Spring Thing. You are out on space but you can be controlled by the selected joystick. One player plays at a time. You must stay away from the laser cannon shots. They will always fire in an upward pattern, but their speed of firing changes. In this game, it is BAD to be on top of things and it is GOOD to be “under-the gun”. Every 2500 gets you a new life. Turn it over at 12,500 and YOU start all over. You'll be glad you do!

00011001 Press RESET, set switches to 00011111 then press the BASIC “=” key

Arcade Golf 1.5

Ken Lill

This game is similar to regular golf. You adjust the direction that the ball will leave it's at rest position by moving the clocklike “Hand” with your JX() motion. JY() selects the club you are using. Pay attention to the wind. The action is highly exaggerated. Once you hit the ball, and it stops, you'll need to move your man to the ball. Use your joystick. When your man's feet are very close to the ball, squeeze the trigger quickly. Your man will set up for his next shot. When you are on the green, you don't have your man to move any more. Just sink the ball in the hole. The lowest score wins.

01011110 Press RESET for the 1st load, set switches to 00111110 then press the BASIC “2” key, then set the switches to 00011111 and press the BASIC “=” key

Pro Golf

Henry Sopko

This is similar to the “Links” games. The “Swing” power / slice box is controlled by to trigger. Let go at just the right time to get the maximum shot. The “right” rime is when the moving box is in line with the line on the right side. Every thing else is pretty much easy to figure out.

01111110 Press RESET For the menu Select game 1, set switches to 00011111 then press the BASIC “=” key

Star Cruiser

Ken Lill

At this time this game is incomplete. When I get an update, you will be notified. You COULD try to fix it for yourself. However saving it to tape or another computer is quite another story. There is a way to do it, but the best is if you have a Blue Ram or Viper unit. Otherwise you have to go through AstroBASIC. Check www.ballyalley.com for any discussions on this procedure.

Most other games are pretty explanatory. Just experiment and have fun. You can't ruin a program by making a mistake, so just have fun.

01101111 Press RESET, set switches to 00011111 then press the BASIC “=” key

L. W. Utility

DO NOT SELECT “Y” FOR INSTRUCTIONS!!

These instructions originally loaded from tape and they are NOT currently available.

This utility is mainly used to make machine language programs. It has some features in it that are no longer useful. What it DOES have is an excellent hex editor and a dis-assembler program in it. This allows for testing of cartridge-like machine language programs if you have write protection on your memory expansion unit, and a way to simulate the cartridge addressing.

There are currently no instructions for this utility, but they will be coming out soon.

Dragon’s Castle

Dragon’s castle is an Adventure type game. You must first guess Merlin’s secret number to get the drawbridge to lower so that you can enter the castle. The first room you encounter has 6 cubicles. Entering the right one and going all the way to the end MAY open a doorway. Your “Sword” actual is a gun that can shoot one shot at a time. You have limited ammo. The object is to slay the dragon and get out of the castle. Your exit is timed so you must know your way out.

This program loads uniquely.

You first set the switches to ViperSoft BASIC 00111111

Then hit RESET. Now, after you see “VIPERSOFT BASIC” on the screen, set the switches to 01010110 and then press RESET. The game will then start.

I hope you enjoy your new UltiMulti Cartridge!
Sincerely, Ken Lill

If you have any questions on game play, please email me:
kenzre@yahoo.com