Instruction Book for the UltiMulti Cartridge Version 1.5.1

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
Artillery Duel
Astro Battle (Space Invaders)
Bally Pin
Blackjack / Acey Deucey / Poker
Brickyard / Clowns
Cosmic Raiders
Dog Patch
Football
Galactic Invasion
Grand Prix / Demolition Derby
Incredible Wizard
LetterMatch / Spell&Score / Crosswords
Life (Degler)
Ms. Candyman
Panzer Attack / Red Baron
Sea Devil
Sea Wolf / Missile
Solar Conqueror
Space Fortress
Speed Math / Bingo Math
Tornado Baseball/Tennis/Handball/Hockey
Basic Checkers
Old Bent Nose
Quadra
Biorhythm
Bowling
Chicken
Blast Droids
Coloring Book
Conan
Mazeman
Music Maker
Soccer / Shoot Out
Songs
Tests
Video Story Book
Pirate’s Chase
Astro BASIC
Bally BASIC
Blue Ram BASIC 1.1
Color BASIC
Star Battle
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 2 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 there 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 1.0… Switches 1,5,7 & 8 are ***OFF*** and all the rest are ***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 it's 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 programmingor 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
_ Changeable print number base
_ Faster overall execution
_ Four new Data types
_ Larger program area
_ Program “bomb” recovery
_ Two character fonts
_ Versatile program editor
_ Tape interface (2000 BAUD)
_ Parallel Printer Driver *
_ Boolean operations
_ Eight mode flags
_ Four color screen
_ Full sound effects Driver
_ More versatile math forms
_ Trace program debug aid
_ User extensibility
_ Windowed graphics and text
_ Faster multiply / divide
_ 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
_ Left Blank NEXT
_ TAB INPUT
_ ESC Halt operations
_ LINE FEED GO+10
_ RIGHT BLANK CIRCLE

In addition, the following new COMMANDS are available
_ DA. DATA
_SN. SNAP
_ DE. DEFAULT
_ SC. SCROLL
_CI. CIRCLE
_ PO. POINT
_ SH. SHOW
_ 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,… **Similar to the standard GOSUB and GOTO commands except that**

GOTO 1111,v,n1,n2,… **then v+1 with n2, etc. It could also be done with:**

DATA v,n1,n2,**…;** **(like OSUB 1111 (or 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:**

BC 239
FA 165
FB 91
FC 233
Graphic Boundaries
X Right XR 80
X Left XL -79
Y Top YT 51
Y Bottom YB -48
X Y screen location XY 0
RM 0 `(ReMainder)
X screen location CX -79
NT 3 Note Timer
Y screen location CY 51

CC 7 (Character Color)
LC 0 (Last Character)
CF LARGE (Character Font size) Large is the original BALLY font.
NB 10 (Number Base)

CR 80 (Character Right boundary)
CL -79 (Character Left boundary)
CT 51 (Character Top boundary)
CB -48 (Character Bottom boundary)

OP …. **This is a user extensibility command. Used when the BASIC language needs to run a machine language program. See Arcadian Volume 6, Page 47 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) **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; For more on the PLAY Command see: ARCADIAN Volume 5, Pages 128 & 156**

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 **Scrolls a window w,h at x,y n lines up or –n lines down.**

SHOW x,y,sm,loc **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÷4+(RM#0))**×**h) +4.
Examples:
6,4 = (6÷4 **(1)** +(RM#0)) **(1)) ×**h **(4)** + 4 **[1+1=2 × 4=8 +4] =12**
19,15 = (19÷4 **(4)** +(RM#0)) **(1)) ×**h **(15)** + 4 **[4+1=5 × 15=75 +4] =79**
20,5 = (20÷4 **(5)** +(RM#0)) **(0)) ×**h **(5)** + 4 **[4+0=4 × 5=20 +4] =24**

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

1111 (edit)(edit) ***T*he 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.**

***<u>NEW DATA TYPES</u>*:**

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×-7=-35). (This is actually a fix)**

↑ **The up-arrow is the Boolean operator AND (7_5=5)**

↓ **The down-arrow is the Boolean operator OR (9_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_2=6) (8_-3=64)**

← **The left-arrow is the Boolean operator XOR (3_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_A+5] will give you 8 XORed with the address of the A variable 27818 (!6CAA) and then it will add 5 for a total of 27823, when what you really wanted is 8 _(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 provides.

CF **Character Font. This variable is set to LARGE (LA.) for the standard 5×7. Character Font or SMALL (SM.) for the new 3×5 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**

XR **Graphic X Right. controlled by the variables XL, XR, YT and YB.**

YT **Graphic Y Top These are set to full screen also.**

YB **Graphic Y Bottom**

FA **Foreground Color A Since there are four simultaneous colors available on the screen at once,**

FB **Foreground Color B these two new variables round out the set. (with BC and FC).**

NB **Number Base 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.**

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:**

BIT 7 **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.**

BIT 6 **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.**

BIT 5 **When this bit is set by the programmer it informs the system that another *user-defined* background processor has been established. A *call* will be made to this processor from the screen interrupt via the vector at 6DCFH.**

BIT 4 **When set, this bit disables screen printing as well as the associated sound.**

BIT 3 **Sends to the printer via MODEM interface* all text meant for the screen.**

BIT 2 **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.**

BIT 1 **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".**

BIT 0 **This bit disables the CTRL "words" and makes the keyboard* yield the actual ASCII CNTL characters.**

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

STATEMENTS

```
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 ABCDEFGHIJKLMNOPQRSTUV;. ALPHAS ARE SPACE RESERVERS FOR POKES
101 ABCDEFGHIJKLMNOPQRSTUV
102 ABCDEFGHIJKLMNOPQRSTUV
103 ABCDEFGHIJKLMNOPQRSTUV
104 ABCDEFGHIJKLMNOPQRSTUV
```

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: (These may not work the same in Blue Ram BASIC 1.0)

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.

O111111O Press RESET for the menu Select game 2, set switches to OOO11111 then press the BASIC "=" key

Whatzit?

Ken Lill

Whatzit? Is a fast paced one player game. The object is to change all items from what ever 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.

O1OOOOO1 Press RESET For the menu Select game 1, set switches to OOO11111 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 you either back over yourself or you try to cross yourself.

OOOOOOO1 Press RESET, set switches to OOO11111 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!

OOO11OO1 Press RESET, set switches to OOO11111 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.

O1O1111O Press RESET for the 1st load, set switches to OO11111O then press the BASIC "2" key, then set the switches to OOO11111 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.

O111111O Press RESET For the menu Select game 1, set switches to OOO11111 then press the BASIC "=" key

Bumble B Cruzer

Formally "Star Cruser"

Ken Lill

The object of this game is to try to avoid being shot by the "camera" looking enemy while trying to get through the walls that are coming at you. You can shoot a hole in the wall (very tiny). You'll need to hit it in several places to be able to get through. There are random portions of the wall that "re-grow" themselves.

Hitting a white wall or box removes that section. If it's yellow, it will take 2 hits (one to change it to white and the next to blow it away. Red sections take 3 hits. 1 turns it yellow, I turns it white, etc. Each round has more walls to get through, and the enemy has a more powerful explosion.

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.

O11O1111 Press RESET, set switches to OOO11111 then press the BASIC "=" key

L. W. Utility

Perkins Engineering

This program will NOT run on Blue Ram BASIC 1.0

DO NOT SELECT "Y" FOR INSTRUCTIONS!!
These instructions originally loaded from tape and they are ONLY available on www.ballyalley.com.
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 disassembler 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.

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 OO111111
Then hit RESET. Now, after you see "VIPERSOFT BASIC" on the screen, set the switches to O1O1O11O and then press RESET. The game will then start.

Ultimulti Cartridge   Version 1.5.1

There are 28 more programs on this cart. Some are repeat AstroBASIC programs that are in BASICcart format. All BASICcart programs were originally written in AstroBASIC and have been made to "autorun" as soon as they are selected and you press RESET. Most were not on the 1.0 version.

There are some different programs

Fawn Dungeon **This is a program that was started by Barry McCleave who has since passed on. This can be used to see how to move something using machine language. *Want to finish it?***

Menu Demo This extremely short program shows the versatility of the MENU part of the on board ROM. Written by Mike White, he has a special piece of artwork on the menu screen. Once selected it, IT MOVES!

PseuDoKu (Beta) **This rendition of the popular Japanese puzzle is the only one of it's kind for the Bally. This plays like a cartridge because it is all written in machine code.**

Here's the BETA version of my Su-Do-Ku clone for the Bally/Astrocade, it's in need of a good NAME since the Working Title inplied vaporware, how's:

BALL
Y-KU
2008

instead? Kind of limited to that arrangement by the control panel - at least no-one has laid claim to that since the game is so recent.

Known omissions are: only two puzzles (the reset one and the NEW one which is a 6-star for sure - good luck with that!) IF I can hold the size to under 4K (doubtful) there's room for up to 50 more puzzles so if there were a way to switch out the upper 4K it would be unlimited!

Also the puzzles were supposed to be scrambled (keeping their radial symetry, of course) to make it appear there were even more puzzles. Also to that end, the numbers 1-9 could easily be shuffled around.

A title screen is needed not only for the (c) notice with my name, but to choose the difficulty of the puzzle from no-stars up to nine-stars.

Selecting with the knob also needs to GRAY out numbers that have been clued out already, but it should still allow you to place them anyway. Oh, yeah - does the knob direction seem backwards on a normal arcade ? I used the parameters 9-1 as they appeared on a built-in title screen.

UNDO works finally!! Pretend that the RED *test* letters are BLACK. When the text turns RED you must delete (or change) the wrong number before filling in another cell or the text turns BLACK. After that all the entries you make are not saved, so you're wasting your time.

Highlight UNDO and pull the trigger. When you release it all of the correct numbers you entered display as unchangable (8x8 wide) numbers and the first wrong number you entered has the cursor on it, for you to change. Again the UNDO text is RED as above, although since that is too much of a giveaway, in the final release it WILL be in BLACK.

HINT does NOT work yet. When it does it will show all possible cells that you can place the number that is dialed in below the HINT text.

Caution: Trigger 2 (or the A key in MESS) now resets the game - twice!

Try out both the standard and MESS keyboard layout versions if you can.
Make a overlay - just don't try to interpolate using the wrong binary.

Here's the short version of the User's Manual from the .ASM file:

```
; To use: Use clue squares to determine which number to place.
; Position red CURSOR on un-numbered square, pull trigger, then
; turn knob (1) to desired number or blank, and release trigger.
; Also enter the number directly from the keypad (1-9) and/or
; over-ride clue squares with the numeric keypad as follows:
;
; '1'  '2'  '3'  'x'  <--  toggle clues using BPA keypad
; '4'  '5'  '6'  'x'
; '7'  '8'  '9'  'x'  <--  'x' is dead key for now [or test]
;
; '1'  '2'  '3'  'x'  <--  set numbers using BPA keypad
; '4'  '5'  '6'  'x'
; '7'  '8'  '9'  '0'  <--  "=" key or "KP Enter" clears cell
;
; MESS uses the PC keys 1-9 to set a number and 0 to erase it.
; Clue toggles are on E, R, O then S, H, PgUp and C, ., PgDn.
;
; Incorrect guesses WILL be allowed, but they will prevent you
; from solving the SuDoKu.   Note that any changes you made to
; the clues will be obliterated if you change a placed number.
;
; Click on "NEW" for a new (well, a different) puzzle, and on
; "UNDO" if BLACK to restore to the original un-populated state
; or when RED to position cursor on the first incorrect entry.
; To get a "HINT" turn the Knob to select the number to isolate
; and hold the trigger in to show where missing digits might go.
;
```

It dosn't say "Only one of each number can be placed in the same row, column, or 3x3 box" although that's pretty much a given how to play.

The clues are arranged the same as the digits under the HINT text. Not so obvious is how to use them. Here's something more on that:

Fire up the game and the layout grid appears, then the initial numbers of the first puzzle appear (always the same 2- maybe 3- star one.) After a second the clues have all been calculated and they appear in the empty cells. It used to show them as they were flipped off and then invert them all - very messy looking.

Look for the cells with only one clue in them – that's the only number that fits there. One whole column can be solved just by doing that. Totally backwards from how I normally do these, which is looking to place one number at a time in rotation, and then looking for other ways to keep going one step at a time (like finding a cell that only one number could go into.)

Down in the lower right corner is two cells (I keep saying cell instead of box since there are only nine boxes, marked by white and gray shading) that only have the clues for 8 and 9 in them. Since one has to be the 8 the other must be the 9 (it's 50-50 but don't guess yet.) you can set some of the other clues to reflect this.
Because these pairs are in the same (bottom) row, no other cell can be an 8 or 9, so move along that row and toggle the 8 & 9 clues off where ever you see them (in MESS use the page up/page down keys for this.) If you knock off the wrong clue just toggle it back so you don't give yourself the wrong info later.
Also because they are in the same box, ditto, turn those numbers off elsewhere in the same white box as well. Now there are two other cells there with two pairs in them, but they can't be of use yet since they are not in the same row or column.

In the upper right box are three cells with the same three clues marked - the 3, 5, 9 triplet. Clear out the other two cells in that box (using the O, H and page down in MESS) and you see that the 2 & 8 have just been solved. Now, two of those triplets are in the same (top) row as one that just became a match. Those three triplets now block any other cell in that row from being a 3, 5 or 9 so clear all those clues. This solves for the 1 and leaves two pairs in the top gray box so clear out the 4 & 7 from the others in that box (using the S and C in MESS, well just the C.)

The two triplets at the top of the left hand column don't have a third, so don't be tempted by them. Triplets can appear as ie: 123, 123, 123 - or can be missing digits, all the way down to: 12, 23, 13 - but never more than the same three numbers, and must appear in the same row, column or 3x3 box.
Trying to think of the numbers in ascending order makes them easier to keep track of. You'll soon recognize the clues by their positions and will no longer have to look at the hint box to know their numbers.

Once the ninth instance of a number has been placed, the corresponding digit will turn gray and if you try to select it with the knob it will turn red to let you know not to try and place a tenth. Once all nine digits have been placed you will be notified if you have solved the SoDuKu correctly or if you have to go back and try to fix your mistakes.

Keep plugging away and you'll solve the puzzle. If you do get stuck you can go back to the standard way of trying to correctly place a 1, then a 2, etc. without using the clues. Also look for rows, columns, or boxes with seven numbers placed already - if you rule out one of the last two numbers in a cell then you've solved them both. With less than seven it's harder to do but you might place one more number, which could then help you elsewhere.
Instructions as well as programming don by Richard Degler.
Used with his exclusive permission:

Sure, you can include the "PseudoKu BALLY BETA_2.bin" version

; pSeUDOKU for the BALLY/Astrocade (C) 2008 by Richard C Degler
;;

L. W. RAM Utility ***Version 1.5***

GaMBITS Engineering

This program will NOT run on Blue Ram BASIC 1.0

The noticeable differences in this version are it will not ask for instructions and there are no longer any modem commands. The SEND and RECV commands are no longer implemented. This was done to make room for the next version's improvements.

Many thanks go out to Richard Degler for his help in deciphering the code of the original Blue Ram Utility. Without his help, I would not have been able to modify the Utility.

I hope you enjoy your new UltiMulti Version 1.5.1 Cartridge!
Sincerely, Ken Lill
If you have any questions on game play, please email me: kenzre@yahoo.com