JOHN D. PERKINS
Data Systems Consultant

5504 Oceanfront
Virginia Beach, Va. 23451
(804) ~~428-9092~~
464-3745

BOB -

THIS NOTEBOOK IS INTENDED TO PROVIDE YOU WITH A GLIMPSE OF THE POTENTIAL OF OUR "BLUE BOX" ACCESSORY. THE FULL POTENTIAL IS TOO VAST TO BE ADEQUATELY TREATED IN A BOOK TEN TIMES THIS SIZE. THIS IS BECAUSE THE "BLUE BOX" PROVIDES THE BALLY ARCADE WITH MANY OF THE FEATURES OF NORMAL "PERSONAL COMPUTERS" ABOUT WHICH VOLUMES HAVE ALREADY BEEN WRITTEN. WHAT I HOPE TO PROVIDE HERE IS A SKETCH OF WHAT THE "BLUE BOX" CAN DO AS WELL AS SOME SAMPLE "EXPERIMENTAL" APPLICATIONS.

CONTENTS

1. WHAT IT IS

THE "BLUE BOX" IS AN ADD-ON ACCESSORY TO THE BALLY ARCADE TO BE USED IN CONJUNCTION WITH BALLY BASIC. IN SIMPLE TERMS IT PROVIDES AN ADDITIONAL 4.2K BYTES OF RAM/ROM (THE PROTOTYPE CONTAINS 2.2K RAM) AND TWO 8 BIT BI-DIRECTIONAL I/O PORTS. THE MEMORY (RAM/ROM) ALLOWS ADDITIONAL STRINGS, MACHINE-CODE (Z-80) PROGRAM/SUB-ROUTINES AND ALTERNATE BASIC PROGRAMS. BASIC PROGRAMS CANNOT BE DIRECTLY EXECUTED FROM THE EXTENDED RAM BECAUSE THE BASIC CARTRIDGE WILL NOT ACCESS IT FOR PROGRAMS.

THE TWO 8 BIT I/O PORTS ALLOW THE CONNECTION OF "OUTSIDE WORLD" DEVICES FOR BOTH SENSING AND ACTUATING. ANY BIT OF EITHER I/O PORT MAY, UNDER PROGRAM CONTROL BE AN INPUT OR OUTPUT BIT. OUTPUTS WILL SINK 2 MA. OF CURRENT. INPUTS DRAW LESS THAN 10 MICROAMPS.

THE RAM IS 2114L3 TYPE AND CAN BE REPLACED BY N82S137 PROMS IF FIXED ROUTINES OR DATA ARE DESIRED. IT IS ANTICIPATED, HOWEVER, THAT RAM (2114L3's) WILL BE USED EXCLUSIVE FOR SOME TIME YET.

THE PROTOTYPE HAS SWITCHES ON TOP WHICH

CONTROL ADDRESS ALLOCATION AND MODE. IT IS ANTICIPATED THAT THESE SWITCHES WILL BE REPLACED BY SOFTWARE FEATURES IN THE PRODUCTION MODEL. TWO ADDRESS RANGES ARE SELECTABLE: 5K (NORMAL EXTENDED "RAM" ALLOCATION) AND 2K (CASSETE MODE). THE MEMORY (EXCEPT FOR 128 BYTES) CAN BE CONFIGURED AS READ/WRITE OR READ-ONLY MEMORY. IN THE READ-ONLY MODE, PROGRAMS/DATA WRITTEN IN PREVIOUSLY ARE PROTECTED FROM INADVERTENT CLOBBER FROM A RESET OR PROGRAM ERROR.

THE I/O PORTS ARE ACCESSIBLE VIA A 24 PIN SOCKET ON THE BACK OF THE ACCESSORY.

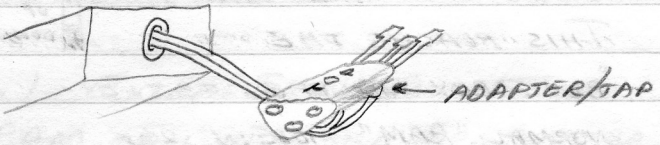
2. INSTALLATION

THE "BLUE BOX" ATTACHES TO THE BALLY ARCADE AT TWO POINTS: AT THE LIGHTPEN PLUG FOR POWER AND THE 50 PIN CONNECTOR FOR DATA EXCHANGE. THE FOLLOWING STEPS WILL INSTALL THE "BLUE BOX" ACCESSORY:

a. REMOVE THE CASE KNOCK-OUTS FOR THE 50 PIN CONNECTOR. THE LIGHT PEN KNOCK-OUT SHOULD HAVE ALREADY BEEN REMOVED.

b. CONNECT THE LIGHT PEN ADAPTER/TAP ON THE ACCESSORY TO THE LIGHT PEN SOCKET. NOTE THE BALLY MAY BE ON OR OFF AS YOU PLEASE FOR CONNECTION

AND DISCONNECTION AS LONG AS THE LIGHT PEN CONNECTION IS THE FIRST MADE AND LAST BROKEN.

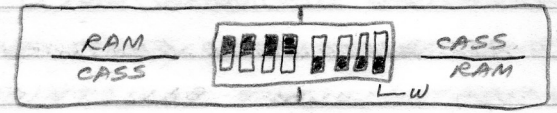


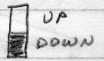
THE ADAPTER/TAP EXTENDS THE LIGHT PEN SOCKET SUCH THAT IT CAN STILL BE USED FOR THE TAPE INTERFACE.

c. SLIDE THE 50 PIN SOCKET ONTO THE 50 PIN CONNECTOR. IT SHOULD SLIDE IN ABOUT 1/4 INCH AND SEAT FIRMLY. THE SWITCHES WILL BE ACCESSIBLE FROM BEYOND THE ARCADE CASE.

d. PLACE THE SWITCHES IN

THE FOLLOWING POSITIONS:





THIS PLACES THE "BLUE BOX" IN THE NORMAL "RAM" MODE.

e. INSERT THE BASIC CARTRIDGE (IF NOT ALREADY IN) AND DEPRESS RESET. THE BLUE-BOX IS NOW READY TO USE.

f. ATTACH THE TAPE INTERFACE IN THE NORMAL MANNER TO THE LIGHTPEN ADAPTER/TAP AND PADDLE 3 CONNECTOR. NOTE THAT THE "BLUE BOX" OBSTRUCTS THE PADDLE 4 CONNECTOR MAKING PADDLE 4 UNCONNECTABLE.

3. EXPERIMENTS

THE FOLLOWING EXPERIMENTS SHOW ONLY A SMALL SAMPLE OF THE POTENTIAL "BLUE BOX" USES

3.1 EXTRA STRINGS. THE RAM CAN BE USED FOR EXTRA STRINGS BY ACCESSING THEM AS ?Ø(2Ø48Ø) THRU ?Ø(247Ø2) IN INCREMENTS OF 2 (THRU ?Ø(22652) FOR THE PROTOTYPE) → (24576)

22652
20480
2172

vs 2048

EXAMPLE:

```
1Ø FOR N=2Ø48Ø TO 2Ø5ØØ
   STEP 2
2Ø %(N) = RND(1ØØ)
3Ø NEXT N
RUN
```

22572

ELEVEN OF THE EXTENDED STRING VALUES NOW CONTAIN RANDOM NUMBERS FROM 1 TO 100. NOW ADD:

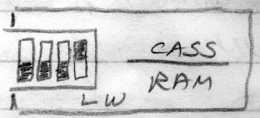
```
40  FOR N=2048 TO 2050
      STEP 2
50  PRINT N÷2-1023,%(N)
60  NEXT N
RUN
```

print 1 2 3 4 %(N) 1-2048

ANOTHER ELEVEN RANDOM NUMBERS ARE STORED IN THE EXTENDED STRING AND PRINTED.

RUN

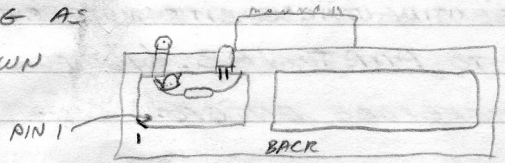
SAME AS ABOVE. NOW CHANGE THE "W" SWITCH:



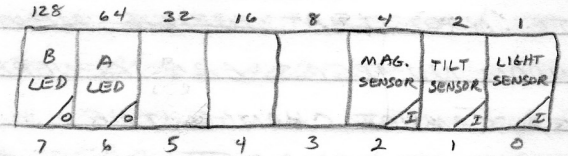
RUN

NOTE NOW THAT WHILE NEW RANDOM NUMBERS ARE GENERATED, THEY CANNOT CHANGE THE EXISTING VALUES. THE "PRINT-OUT" IS IDENTICAL TO THE PREVIOUS ONE. THE STRING IS NOW A "READ ONLY" STRING. RESET THE "W" SWITCH AND IT AGAIN BECOMES A READ/WRITE STRING.

3.2 I/O OPERATIONS. THE I/O CAPABILITIES OF THE "BLUE BOX" ARE EXTREMELY FLEXIBLE. THIS EXPERIMENT ONLY SCRATCHES THE SURFACE. PLUG IN THE DEMO PLUG AS SHOWN



RESET. THE DEMO PLUG ASSIGNS PORT B AS FOLLOWS

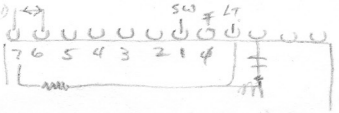


PORT A IS FUNCTIONALLY EQUIVALENT BUT IS NOT CONNECTED IN THE DEMO PLUG.

⇒ OUTPUT BIT

⇒ INAUT BIT

3.2.1 OUTPUTS. RESET SETS ALL I/O BITS TO THE INAUT MODE. TO PERFORM OUTPUT WE MUST ASSIGN THE OUTPUT BITS. (ENSURE THAT THE "W" SWITCH IS IN THE ON (▮) POSITION.) TO ASSIGN BITS AS OUTPUTS, A NUMBER REPRESENTING THOSE BITS MUST BE SENT TO PORT 163. (162 FOR PORT A) THEREFORE ENTER:

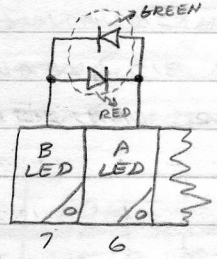


OUT(163) = 128+64

-OR-

OUT(163) = 192

RESET ALSO SETS ALL OUTPUT BITS TO LOGICAL "0" (0V.) THEREFORE, BITS 7 & 6 OF PORT B ARE NOW LOGICAL ZEROS. THE DEMO PLUG IS WIRED AS FOLLOWS:



BY SETTING BIT 7 TO "1" (+5V) AND LEAVING BIT 6 LOW (0V) THE RED PORTION OF THE LIGHT WILL LIGHT. THERE TWO METHODS OF DOING THIS

OUT(161) = 128

-OR-

$(159) = 1$

$(161) SETS ALL PORT B BITS SIMULTANEOUSLY. EQUATING IT TO 128 SETS BIT 7 AND CLEARS BIT 6. OTHER BITS ARE STILL IN THE INPUT MODE AND ARE THUS UNAFFECTED.

$(152+BIT) SETS THE RELEVANT BIT INDIVIDUALLY. EQUATING $(159) TO ANYTHING SETS BIT 7, LEAVING BIT 6 AND ALL OTHERS UNAFFECTED.

NOW ENTER:

$(161) = 64

THIS ACTION CLEARS BIT 7 AND SETS BIT 6 WHICH LIGHTS THE GREEN LIGHT INSTEAD OF THE RED. TO TO PERFORM THIS SAME ACTION USING THE INDIVIDUAL BIT MODE

ENTER:

$(161) = 128 (TO TURN RED BACK ON)

-FOLLOWED BY-

$(158) = 1 (TURNS BIT 6 ON)

$(143) = 1 (TURNS BIT 7 OFF)

BOTH METHODS ARE AVAILABLE AND GENERALLY ONE WILL BE MORE EFFICIENT BASED ON THE SPECIFIC APPLICATION. THE FOLLOWING TABLE SHOWS WHICH $'S CONTROL WHICH PORT ACTIONS

$( )	PORT	R/W	ACTION	
128/136	A/B	R	READ	BIT 0
		W	CLEAR	
129/137	A/B	R	READ	BIT 1
		W	CLEAR	
130/138	A/B	R	READ	BIT 2
		W	CLEAR	
131/139	A/B	R	READ	BIT 3
		W	CLEAR	
132/140	A/B	R	READ	BIT 4
		W	CLEAR	
133/141	A/B	R	READ	BIT 5
		W	CLEAR	
134/142	A/B	R	READ	BIT 6
		W	CLEAR	

135/143	A/B	R	READ	BIT 7
		W	CLEAR	
144/152	A/B	R	READ	BIT 0
		W	SET	
145/153	A/B	R	READ	BIT 1
		W	SET	
146/154	A/B	R	READ	BIT 2
		W	SET	
147/155	A/B	R	READ	BIT 3
		W	SET	
148/156	A/B	R	READ	BIT 4
		W	SET	
149/157	A/B	R	READ	BIT 5
		W	SET	
150/158	A/B	R	READ	BIT 6
		W	SET	
151/159	A/B	R	READ	BIT 7
		W	SET	
160	A	R	READ	ALL BITS
		W	SET/CLR	
161	B	R	READ	ALL BITS
		W	SET/CLR	
162	A	W	DEFINE OUT BITS	
163	B	W	DEFINE OUT BITS	
164	MODE	W	DEFINE MODE	

NOTE: READ BITS SHOW UP IN BIT 7 AS 128 DECIMAL.

OUTPUTS WILL SUPPLY UP TO 2 MA. WHERE HIGHER CURRENTS ARE REQUIRED USE TRANSISTOR DRIVERS.

3.2.2 INPUTS.

DEPRESS RESET TO CLEAR ALL OUTPUTS. ENTER:

```
10 PRINT &(161); GOTO 10
RUN
```

THE SCREEN SHOULD PRINT A COLUMN OF ZEROS. WE ARE READING THE THREE DEMO SENSORS. IF A "1" IS PRINTING, THERE IS TOO MUCH AMBIENT LIGHT. IF A "2" PRINTS, THE BALLY IS NOT ON AN EVEN SURFACE. IF A "4" PRINTS, A SUPER STRONG MAGNETIC FIELD EXISTS. A NUMBER FROM 1 TO 7 INDICATES A COMBINATION AS SHOWN IN THE FOLLOWING TABLE. USE A FLASH LIGHT AND A MAGNET TO CAUSE A RESPONSE. TILT THE BACK OF THE BALLY UP TO CAUSE A TILT INDICATION.

DEMO RESPONSE

PRINT OUT	VISUAL SENSOR	TILT SENSOR	MAGNETIC SENSOR
0			
1	X		
2		X	
3	X	X	
4			X
5	X		X
6		X	X
7	X	X	X
	2^0	2^1	2^2

NOTE THAT INDIVIDUAL BIT READS CAN ALSO BE PERFORMED. ENTER:

```
10 PRINT 8(152); GOTO 10
RUN
```

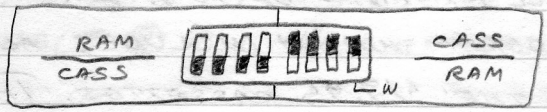
THIS EXERCIZES THE VISUAL SENSOR BY ITSELF. 0 ⇒ NO LIGHT; 128 ⇒ LIGHT PRESENT.


```
A=20480; FOR N=A TO 22527; %(N) =
%(N) ÷256^256+KP; NEXT N
```

24576 -- 28671 - in prod version - 4096

3.2.3 CASSETTE EMULATION.

DEPRESS RESET AND LOAD THE FIRST PROGRAM FROM THE DEMO TAPE USING BALLY BASIC IN THE NORMAL WAY. NOTE! DO NOT STOP TAPE UNTIL PROMPT APPEARS!! THIS SHOULD TAKE ABOUT 5 OR 6 MINUTES. NOW CHANGE THE "W" SWITCH TO OFF (▮) TO PROTECT THE MEMORY. NEXT SWITCH THE REMAINING THREE RIGHT-HAND SWITCHES TO OFF AND THE LEFT-HAND SWITCHES TO ON. DEPRESS



RESET. NOTE THAT THE BALLY THINKS THAT A 280 ZZZAP CASSETTE IS INSTALLED. YOU MAY PLAY EITHER GAME JUST AS IF

THE REAL CARTRIDGE WAS IN PLACE. AND IT IS UNAFFECTED BY RESET OR ANY OTHER ENTRY. WHAT I HAVE DONE IS TO COPY BALLY'S 280 ZZZAP CASSETTE ONTO TAPE AND YOU HAVE RELOADED IT INTO THE "BLUE BOX" WHERE YOU CAN EXECUTE IT! THIS MEANS THAT WE CAN WRITE HIGH-SPEED FAST-ACTION PROGRAMS AND DISTRIBUTE THEM ON TAPE INSTEAD OF ROM CARTRIDGE. THE MEMORY IN THE PROTOTYPE WILL EXECUTE UP TO A 2K PROGRAM WHICH INCLUDES ALL OF THE $19.95 CASSETTES. THE $24.95 CASSETTES REQUIRE 4K WHICH THE PRODUCTION MODEL "BLUE BOX" WILL BE ABLE TO HANDLE.

3.2.4 COLORS. IF YOU HAVE CAREFULLY OBSERVED SOME OF THE GAMES PREPARED BY BALLY YOU WILL NOTICE THAT THEY SHOW MORE THAN 4 COLORS AT A TIME. SPACE INVADERS IS A GOOD EXAMPLE. RETURN THE SWITCHES TO THE "RAM" POSITION AND RESET FOR BALLY BASIC. LOAD THE NEXT SEGMENT FOR A COLOR DEMO SHOWING WHAT IS POSSIBLE WITH THE "BLUE BOX" OPERATING IN THE "BACKGROUND" MODE. STOP THE DEMO BY DEPRESSING THE HALT (H) KEY. NOW ENTER:

```
A=16;C=12;&(9)=5Φ
```

NOTICE THAT EACH LINE IS HIGHLIGHTED IN A DIFFERENT COLOR. WHAT IS HAPPENING

IS THAT THE COLOR REGISTERS ARE BEING CHANGED "ON THE FLY" AS THE RASTER SCANS FROM TOP TO BOTTOM. BASIC OPERATES IN A NORMAL MANNER AS THE PRIMARY (FOREGROUND) PROGRAM WHILE THE COLOR CHANGE ROUTINE OPERATES AS A SECONDARY (BACKGROUND) PROGRAM IN THE BLUE BOX. ENTER:

```
CLEAR ; LIST 10, 5
```

THIS IS THE BEGINNING OF THE COLOR DEMO. THE BACKGROUND PROGRAM IS ACTIVATED BY THE "CALL (21504)." ENTER:

```
:RETURN
```

THE COLORS ARE GONE! ENTER:


```
CALL (21504)
```

AND THEY'RE BACK! WHAT YOU ARE DOING IS DEACTIVATING AND REACTIVATING THE BACKGROUND PROGRAM AS I HAVE WRITTEN IT. THIS WILL NOT WORK WITHOUT THE PROPER PROGRAM IN THE BLUE BOX.

THIS BACKGROUND PROGRAM COMMUNICATES WITH THE FOREGROUND (AS WRITTEN) THROUGH VARIABLES A, B, AND C. A IS THE COLOR BAND WIDTH. TRY:

```
A=32
```

-AND-

```
A = 8
```

-AND BACK TO

```
A=16
```

B CONTROLS THE BOTTOM LINE OF COLOR CHANGE. TRY:

B=1Ø

-AND-

B=8Ø

-AND BACK TO-

B=23Ø

C CONTROLS THE TOP LINE OF COLOR CHANGE. TRY:

C=5Ø

-AND-

C=9Ø

-AND BACK TO-

C=12

BY MOVING THE HORIZONTAL BOUNDARY OVER, MORE COLORS SHOW UP. - ENTER:

$S(9)=2Ø$

THE ACTUAL COLORS ARE STORED IN BLUE BOX RAM. THE DEMO GENERATED RANDOM COLORS. TRY:

%(22562)=Ø

THIS SHOULD BLACKEN THE 7TH COLOR BAND ON THE LEFT SIDE.

%(22566)=1799

WILL WHITEN THE CORRESPONDING RIGHT SIDE. YOU CAN BEGIN TO SEE THE POSSIBILITIES. THE BACKGROUND MODE CAN ALSO BE USED FOR OTHER THINGS. WE ARE COMPLETING A KEYBOARD INTERFACE WHICH SCANS A SIMPLE KEYBOARD VIA THE I/O PORTS.

THE ONLY EXTERNAL COMPONENTS WHICH ARE REQUIRED ARE THE KEYS (SWITCHES) AND SOME RESISTORS AND DIODES. THE BACKGROUND PROGRAM WILL INJECT THE ASCII CHARACTERS INTO THE FOREGROUND BUFFER (BASIC) WITH NO ADVERSE EFFECT ON NORMAL OPERATIONS. THE BACKGROUND MODE CAN BE USED FOR A WIDE VARIETY OF APPLICATIONS SUCH AS:

- TIMING (TO A 60TH OF A SEC.)
- MONITORING GRAPHICS INTERACTIONS
- PROVIDING HIGH SPEED ANIMATION
- PLAYING COMPLEX MUSIC
- INTERPRETING LIGHT-PEN INPUTS