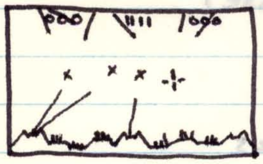
Missle command NOTES

Keep in mind items *
version for Bally handle?

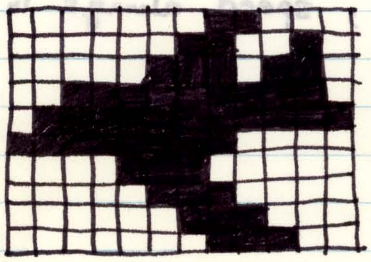
Summary of actual game



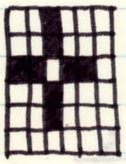
RED - score, missle trail, Planes + satellites
BLUE - cities, outgoing missles, cursor
Black - Background
Yellow - ground (the extra color)
WHITE - missle heads, X left by cursor,
Clouds (Black where overlap patterns!)
(control color - clouds active, heads + X erased before checking for intercept - then replaced)

PATTERNS

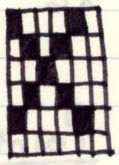
Plane



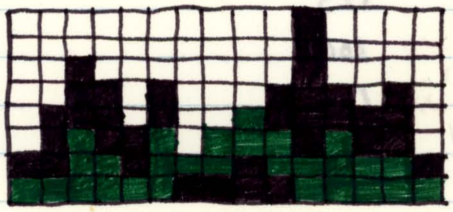
floating cursor



missle marker (Blinks)

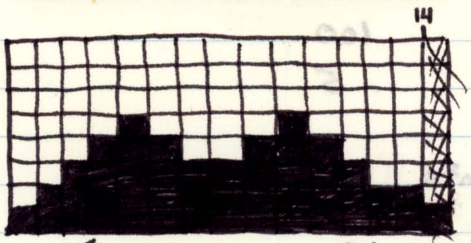


City



Smart missles and satellites to be redesigned

Base



Bottom 10 pixels will show # of Missles

WRITING TO SCREEN

Score, missle count, other? does not have to be in interrupt

Board Sequences (OF ARCADE GAME)

x 1	- 12 missles , 1 wave , No Planes
x 1	- 1wave , 3 planes (Poss.)
x 2	- 2 waves , 4 planes (poss.)
x 2	- 1 wave , 2 planes , faster (visible increase)
x 3	- 2wave , 3 planes
x 3	- 2 waves , 3 planes , 1 smart ass
x 4	- 2 waves + trailers , 1 smart ass , 3 planes
x 4	- 1 wave + trailers , 3 planes , 2 smart asses
x 5	- 2 waves , 3 planes , 3 smart asses
x 5	- 2 waves , 3 planes , 4 smart asses
x 6	- } 2 waves max (sometimes with trailers), up to 8 or 10 smart asses, 3 or 4 planes max , speed always increases
x 6	-
x 6	↓

Point Values

intercept points

Smart (ass) missles	125
Killer satellite	100
Bomber	100
Attack missle	25

Bonus Points

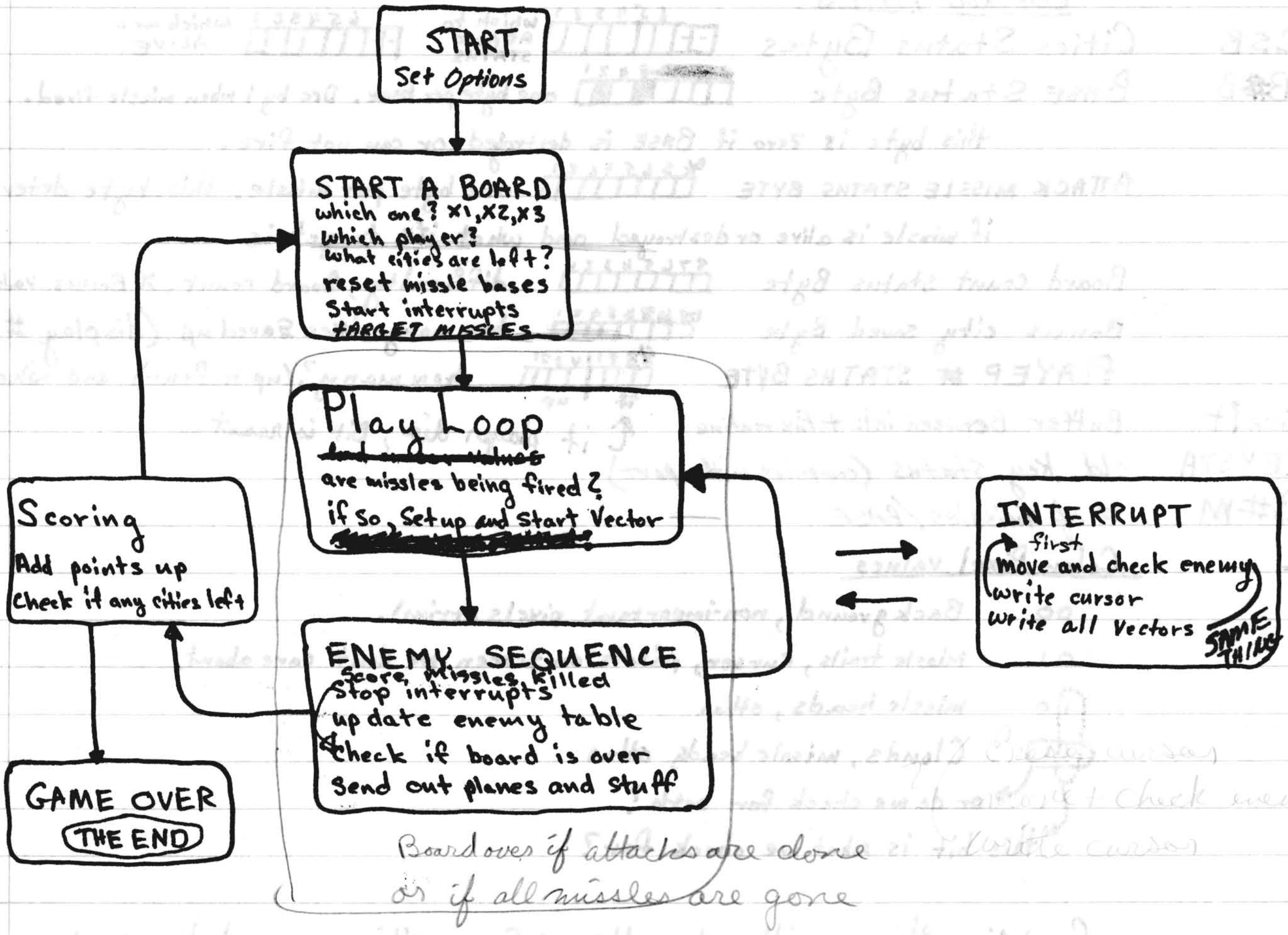
Saved cities	100
unused missles	5

Bonus cities every 10,000 points

take 3 cities max per turn

Idea? Give points for saved cities?
show extra cities saved

FLOWCHART OF GAME LOGIC



<u>Programming Ideas</u>

<u>CONTROL BYTES</u>

CSB Cities Status Bytes — 6 5 4 3 2 1 which to ATTACK STATUS — 6 5 4 3 2 1 which are ALIVE

B#B BASE Status Byte — 8 4 2 1 one byte per base. Dec by 1 when missle fired.
this byte is zero if BASE is destroyed or can not fire.

ATTACK MISSLE STATUS BYTE — one byte per missle. this byte determines if missle is alive or destroyed and what its target is

Board count status Byte — 8 7 6 5 4 3 2 1 difficulty, Board count, X Bonus value

Bonus city saved Byte — 128 64 32 16 8 4 2 1 how many cities saved up (display # left!)

PLAYEP # STATUS BYTE — 4 3 2 1 | 4 3 2 1 — # | up — how many? (up to four) and whos up?
↑ if player dies, Bit is reset

Fire it — Buffer Between int. + Fire routine

KEYSTA — old key status (compaire with port)

B#M — # of missles/BASE

<u>Color Pixel values</u>

00 Background, non-important pixels (trim).
01 Missle trails, cursor, pixels on screen we dont care about.
10 missle heads, other
11 Clouds, missle heads, other
→ or do we check for both?
→ this bit is what we check for?

Certain things will not matter - Score, cities, ground, bases, etc.
why? - they can not be hit by players fire! (any bit can be used)

BSB is BASES ALIVE?

<u>NOTATION</u>

CSB	City Status Byte (# ALive + which ones)
B#B	Set to 3 each new Board, checked when missle is tried to Be fired
CCOTAB	city coor. table (locations to place cities)
BCOTAB	Base coor table (same for Base)
city	city Pattern
BASE	Base Pattern
BCSB	BOARD COUNT STATUS BYTE (what #? Bonus)
Firepos	where to fire to
Curpos	cursor position
Coltab	color table

Detailed logic flows

Checking if button is pushed

From Cursor movement

```
OK
Check from first base, fire? can it fire?
        YES - load base + cursor values, leave marker x, DEC missle count
No ↓          Start vector (limits, deltas, etc.)
Check from third base, fire? Can it fire?
        Yes - load base + cursor values, leave marker x, DEC missle count
No ↓          Start vector (limits, deltas, etc.)
Check from Second base, fire? can it fire?
        Yes - load base + cursor values, leave marker x, Dec missle count
No ↓          Start faster vector (limits, deltas, etc.)
FALL INTO CHECK ENEMY TABLE
```

ENEMY TABLE

update all missles (vectors), did any hit an deadly pixel?
Plane start?
Plane movement
Smart bombs
has player used all missles? YES - DJ and finish enemy table

Scoring (two parts - 1. enemy killed durring play 2. whats left at end of board)

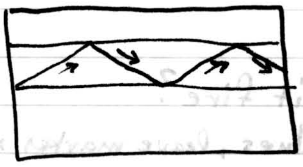
1. durring play
   inc score when a missle is stopped, Plane is Stopped, Smart bomb, etc

2. load reg. with ~~[illegible]~~ city value
   make it times bonus
   Check if cities left YES - Add score No - None left? end game

(SAME IDEA FOR UNUSED MISSLES)

IDEAS

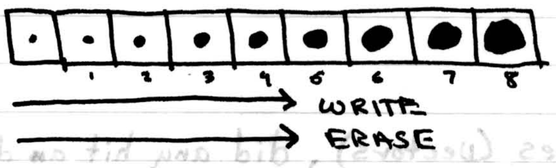
USE ΔY with Reverse delta and Narrow Boundarys

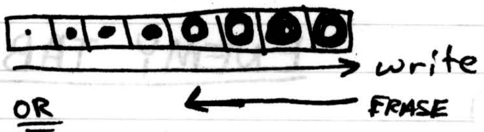


MAX ERASE SPEED?
use VBlank?
what ΔX + ΔY and Time Base will erase fully?

BYTE controlled difficulty level

8 levels of difficulty? Have a table of explosion patterns, and difficulty will be determined by how far you enter the explosion table.





Scale X,Y Joystick inputs

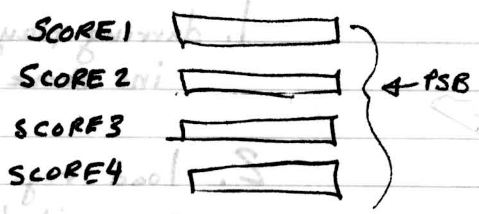
Bit set will provide limits for screen movement

Normal BALLY HANDLE

MAKE A VERSION TO WORK with Normal Bally handle $29.95 ?

More Scoring ideas

Continuously write scores to screen. (AFTER ALL OTHER WRITES)
Player Status BYTE will point to current player.
Check to see if SCORE >10000 if yes
add 10000 to check and give ~~Bonus~~ City
from playloop?



Screen write format

	ERASE		write	
	MOVE	OR	ERASE	INTERRUPT
enter here →	WRITE		move	
			WRITE	
			ERASE	

Tape Load Formats

1. Basic Program   Attract mode with instructions
   falls into :RUN   (tape must be stopped)
   Basic Program will set color ports, lines to display, etc.
2. Just :RUN

Music, Explosions, Sound

format :   EMusic
            BMusic
            Music Stack (where)
            Voice Byte
            Music Score (where)

Interrupts

if ACTINT is used, set timeout counter to 255

Stack

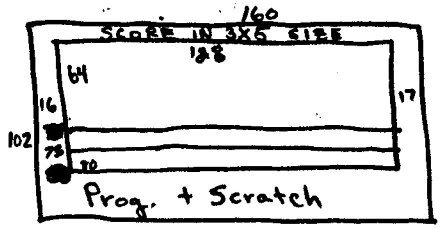
where will it go, where will it go?

Need RAM?

Spare RAM is in UNUSED SYS RAM ROUTINE AREA

Screen Size

Joy stick range	64 * 128
total screen	102 * 160



IDEAS

indicate lower missles by changing color of missle base (Rewrite with different pattern)

STEAL the interrupt handler from Football

INTERRUPTS

we can do 2 (two) SCREEN MOVEMENTS PER INTERRUPT

MAYBE 3 IF WE SEPARATE INTS.

INT 1	INT 2
LOOK AT KEYS	DO 3 SCREEN MOVEMENTS
MOVE + CURSOR	

FOR INT 2

```
Next    LD    HL, CURVEC          HL = CURRENT VECTOR
        DEC   HL
        JP    P, JMP              did we go to zero?
        LD    (HL), # of TOTAL VECTORS    Re load
        ~~NO IS VECTOR ACTIVE? yes~~
JMP     LD    C,(HL)
        LD    HL, VEC TAB
        LD    B, Ø
        ADD   HL,BC
        ADD   HL,BC
        LD    A,(HL)
        INC   HL
        LD    H,(HL)
        LD    L, A
        INC   HL                  Points to vector STATUS
        Bit   3,(HL)
        JP    Z, Next
        ↓ make it move   CALL ▭
        Next till 3 are done!
```

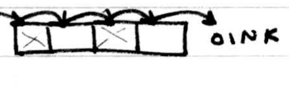
PROGRESSIVE EXPLOSIONS

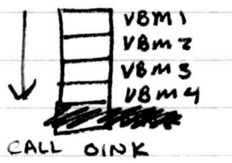
4 levels of missles/explosions for player

Reserve 4 VB's and 4 explosion areas

if player tries to fires missle see if VB is open

Yes then use it, no then OINK





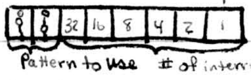
```
INC HL      → exit if find one
INC HL LO
DJNZ -LO
[illegible]
```

EXPLOSION BYTES

1 X POS.

2 Y POS.

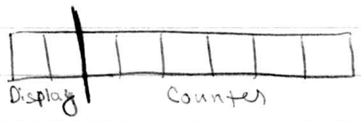
3 Current STATUS (which explosion to display)

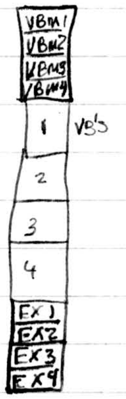
4 

0 = START 255 = time to stop

00 -Write 1
01 ~~erase 1~~ write 2
10 ~~erase 2~~ write 3
11 ~~erase 3~~ ~~write 4~~ erase all

00 WRITE 1
01 WRITE 2
10 WRITE 3
11 ERASE ALL

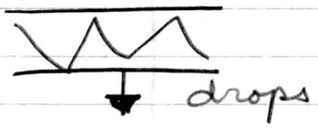




512 674 9,56

ICBM LORE

Mother Ship sends out
⊛Fighter - ship
⊛nermal - smart bombs



Sound

EXPLOSIONS	- NOISE + NOISE VOLUME
Ship	- tone A
Nermal	- tone B
Background?	- tone C
oink	- all?

Player

	Player 1	Player 2	Player 3	Player 4
BSB				
CSB				
SCORE { three Bytes? }				
Board #				

debounce cursor more?

VECTOR BLOCK EXTENSIONS

0	MAGIC
1	STATUS
2	TIME BASE
3	X Delta
4	
5	X Position
6	
7	X Limit
8	Y delta
9	
10	Y Position
11	
12	Y Limit
13	TARGET POSITION
14	
15	EXPLOSION POSITION
16	
17	ANIMATION COUNTER
18	LIMIT TABLE
19	

IDEAS

use destroied patterns?

use cartoons

USE CRADERS AND ROCKS!!!

ICBM ATTACK / MARTIAN OUTPOST / LUNAR BASE

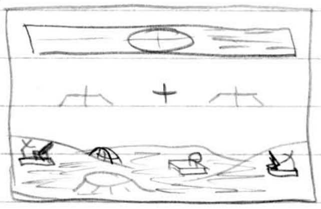
JX and JY moves cursor

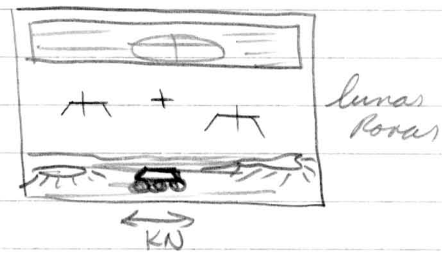
TR fires missiles

KN selects "Base"

Object: protect your space pad, base, and rover from space attacks. laser weapons fire when trigger is pulled. Rovers zip around.

Protect settlement, supplies are brought by Spaceship. Status display at top of screen.




lunar
Rover
KN