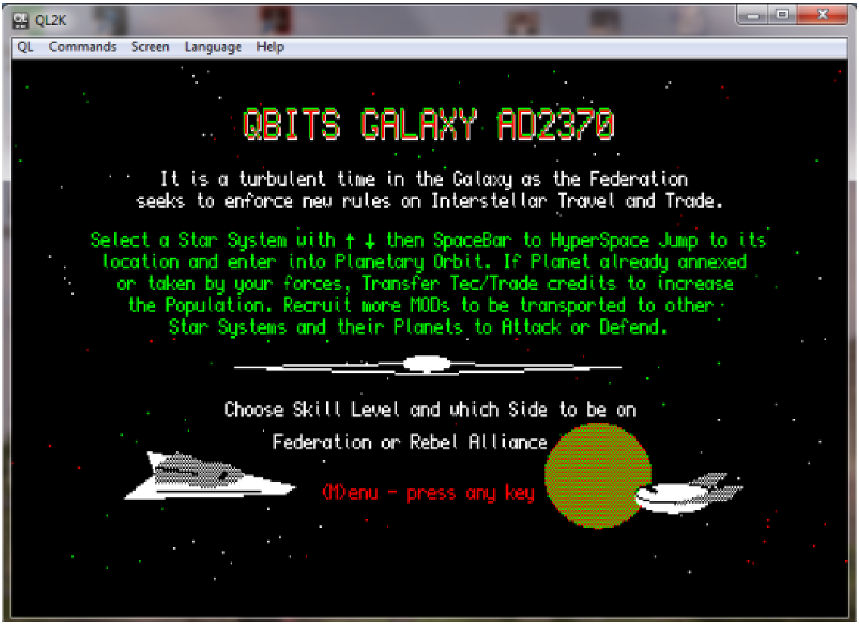
Sinclair QL retro gaming



QBITS GALAXY AD2370

It is a turbulent time in the Galaxy as the Federation
seeks to enforce new rules on Interstellar Travel and Trade.

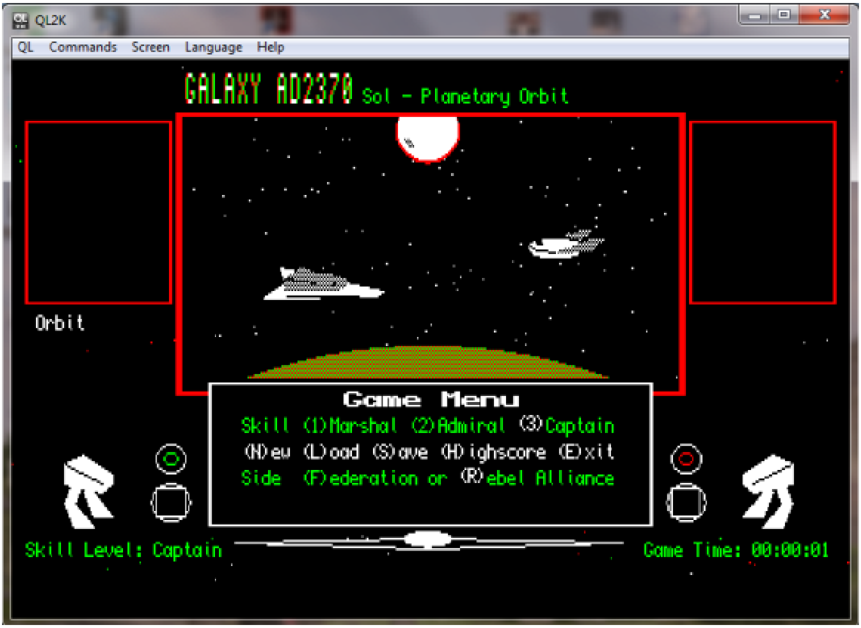
Select a Star System with ↑ ↓ then SpaceBar to HyperSpace Jump to its
location and enter into Planetary Orbit. If Planet already annexed
or taken by your forces, Transfer Tec/Trade credits to increase
the Population. Recruit more MODs to be transported to other
Star Systems and their Planets to Attack or Defend.

Choose Skill Level and which Side to be on

Federation or Rebel Alliance

(M)enu - press any key

Sinclair QL retro gaming



GALAXY AD2370 Sol - Planetary Orbit

Orbit

Game Menu

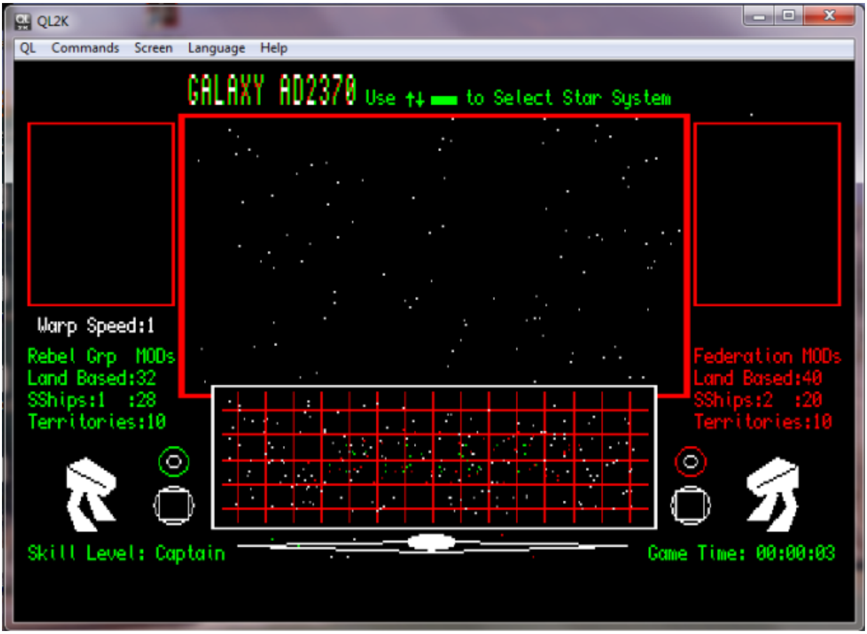
Skill (1)Marshal (2)Admiral (3)Captain

(N)ew (L)oad (S)ave (H)ighscore (E)xit

Side (F)ederation or (R)ebel Alliance

Skill Level: Captain

Game Time: 00:00:01



GALAXY AD2370 Use ↑↓ ▬ to Select Star System

Warp Speed:1

Rebel Grp MODs
Land Based:32
SShips:1 :28
Territories:10

Federation MODs
Land Based:40
SShips:2 :20
Territories:10

Skill Level: Captain

Game Time: 00:00:03

QBITS Galaxy AD2370DL

As to my thoughts behind the QBITS Galaxy AD2370, the early to mid nineteen eighties was a time when I had perhaps an overly enthusiastic interest in Science Fiction. The first of the Star Wars films was released and my family were considered by some to have an addiction to playing board games, which included variants of the game RISK. I guess all played their part in how this Game idea originated and began its development from early concept to a working program.

QBITS Galaxy AD2370 - B**asic Concept**

My to do list began:- create a QL Computer Game where Starships battle it out with laser beam weapons to capture a Galaxy's Star Systems. To expand upon this there was to be the ability to exchange Technology and Trade thereby raising Populations as a necessity to Recruiting more Troops. Multiple players were a consideration, or perhaps just the Computer to imitate the moves and actions of a human player. The winner of the game was the one who had captured all of their opponents Star Systems, leaving them unable to recover and retaliate. The stratagems deployed to achieve a WIN ultimately to depend on the Roll of the Dice, where the final result can never be certain.

Create an Intro page with basic instructions on how to play the Game before moving to a Menu providing choice of side plus the normally expected options of New, Load, Save and Exit. Levels of difficulty was added later to the list.

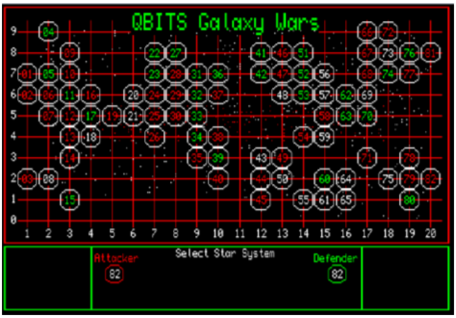
The background for the Game was as expected to be the Star Systems of a Galaxy far, far away (No surprise there). Early beginnings, which have persisted was with just two sides the Rebel Alliance and Federation of Planets. At initialisation some of the Star Systems randomly allocated to either side, others left as independent.

Each Star system to be given various levels of Population, Technological development and Interstellar Trading abilities. Other parameters covered such things as Galaxy Map coordinates, a Planet's surface colour, then thoughts on alternative directions when approaching the Planet within a Star System etc.

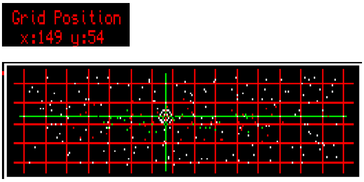
Later additions were the **End Game** sequence, a Final Battle between the Rebel Alliances defending their home Planet against the Federations Death Star. Then to follow the Final Victory a high score **League Table,** which can be amended with the name, nickname or pseudonym of the new high scorer.

Quick Note: I developed most of this code using the QL2K emulator on a six years old Desktop running Win 7. The speed is mega fast than even a compiled program running on an original QL hardware. Therefore, running QBITS Galaxy AD2370 at anything of a respectful speed you are going to need expanded memory and a later much faster version of QL Hardware or download and use an emulator such as QL2K or SMQL on a more modern computer (See page 36 for downloads).

QBITS Galaxy AD2370 - The Beginnings



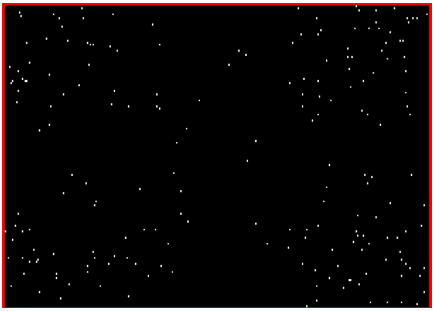
Initial thoughts began with constructing a Galaxy with Stars Systems in the form of a Territorial Map.



This changed to a Galaxy Map of random POINTs (Stars) with Grid Position of the selected Star identified by a double circle and the crossover point of horizontal & vertical Grid Lines that could be moved from within a REPeat loop by use of the cursor keys.

QBITS Development - Galaxy AD2370 Space View Screen

An action view screen I guess is an expectation with any Computer Game. For Galaxy AD2370 this had to be moving through the Stars of space. To create this illusion the view screen was subdivided into quarters (four sub-windows), then using the PAN and SCROLL commands randomly placed POINTs (Stars) are moved directionally out from the centre of the view screen towards the four corners.



After playing around with the number of generated Star POINTs and the SCROLL and PAN values the code below gave a respectable early result.

```
100 REMark QBGT01 (Test for Space View)
101 :
102 WINDOW 512,256,0,0:PAPER 0:CLS
103 DATA 304,152,98,19
104 DATA 150,75,100,20,150,75,250,20
105 DATA 150,75,100,95,150,75,250,95
106 RESTORE 103
107 :
108 FOR i=3 TO 7:OPEN#i,scr_:READ a,b,c,d:WINDOW#i,a,b,c,d
109 BORDER#3,1,2:INK#3,7
110 FOR i=1 TO 100:POINT#3,RND(10 TO 180),RND(10 TO 90)
111 REPeat lp
112   FOR i=1 TO 6:POINT#3,RND(20 TO 180),RND(10 TO 90)
113   PAN#4,-2:SCROLL#4,-2:PAN#5,2:SCROLL#5,-2
114   PAN#6,-2:SCROLL#6,2:PAN#7,2:SCROLL#7,2
115   PAUSE 5
116   IF KEYROW(1)=8:FOR i=3 TO 7:CLOSE#i:STOP
117 END REPeat lp
```

QBITS Development - Galaxy AD2370 Planetary Orbit

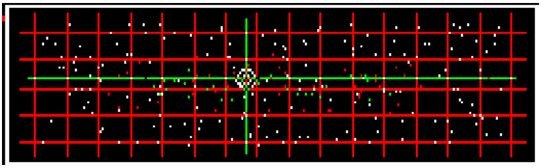
In exploring ways to emulate approaching a Star System and then coming into orbit above a Planet, both scenarios were accomplished by the redrawing an enlarging disk in the form of a SuperBASIC FILLed CIRCLE.

Galaxy Map, Space view and Planetary Orbit are all combined in the test program below.

```
100 REMark QBGT03 (Test for Galaxy Map - Space View - Orbit)

104 WINDOW 512,256,0,0:PAPER 0:CLS
106 DATA 304,152,98,19
108 DATA 150,75,100,20,150,75,250,20
110 DATA 150,75,100,95,150,75,250,95
112 DATA 304,12,98,2

116 sx=164:sy=48:RESTORE 106:AT#2,18,0:PRINT#2,'QBG03'

120 InitWin:InitGrid:Main

124 DEFine PROCedure InitWin
126 FOR i=3 TO 8:OPEN#i,scr_:READ a,b,c,d:WINDOW#i,a,b,c,d
128 BORDER#3,1,2:INK#3,7:BORDER#8,1,2:INK#8,7
130 FOR i=1 TO 100:POINT#3,RND(10 TO 160),RND(10 TO 90)
132 END DEFine
```

Creating the Galaxy Map with its Stars and Grid was not difficult, moving the Map Grid Lines the XOR function of OVER is used.

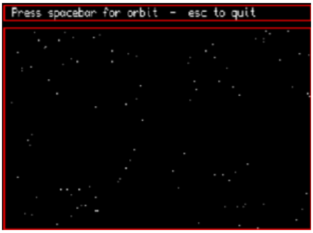
```
136 DEFine PROCedure InitGrid
138 OPEN#13, scr_:WINDOW#13,260,60,120,170
140 OPEN#14, scr_:WINDOW#14,90,22,8,6
142 CLS#13:BORDER#13,1,7:INK#13,2:SCALE#13,100,0,0
144 FOR h=12 TO 96 STEP 18
146  LINE#13,6,h TO 326,h
148 END FOR h
150 FOR i=16 TO 330 STEP 20
152  LINE#13,i,3 TO i,98
154 END FOR i
156 INK#13,2:FOR i=1 TO 60 :POINT#13,RND(60 TO 260),RND(30 TO 70)
158 INK#13,4:FOR i=1 TO 30 :POINT#13,RND(80 TO 240),RND(40 TO 60)
160 INK#13,7:FOR i=1 TO 180:POINT#13,RND(20 TO 310),RND(10 TO 90)
162 OVER#13,1:PRINT#13,'   use the cursor keys to move grid line':OVER#13,0
164 END DEFine
```



```
168 DEFine PROCedure Gridpos
170 IF sx< 10:sx= 10
172 IF sx>320:sx=320
174 IF sy<  5:sy=  5
176 IF sy> 95:sy= 95
178 OVER#13,-1:INK#13,4
180 LINE#13,sx,5 TO sx,95:LINE#13,12,sy TO 320,sy:OVER#13,0
182 CLS#14:INK#14,2:PRINT#14,'Grid Position'\' x:';sx;' y:';sy
184 AT#2,18,0:PRINT#2,'QBGT03'
184 END DEFine
```

```
Grid Position
 x:149 y:54
```

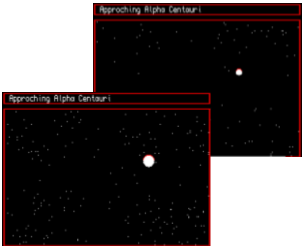
```
190 DEFine PROCedure Main
192 CLS#8:PRINT#8,' Press any key to enter orbit  -  esc to quit'
194 Gridpos
196 REPeat lp
198   StarMove
200   IF KEYROW(1)=8:FOR i=3 TO 20:CLOSE#i:STOP
202   IF KEYROW(1)=64   :StarView:Eorbit:PAUSE:Lorbit
204   IF KEYROW(1)=  2  :IF sx>5     :Gridpos: sx=sx-5    :Gridpos
206   IF KEYROW(1)=  4  :IF sy<95    :Gridpos: sy=sy+5    :Gridpos
208   IF KEYROW(1)= 16  :IF sx<320  :Gridpos: sx=sx+10  :Gridpos
210   IF KEYROW(1)=128 :IF sy>5     :Gridpos: sy=sy-10   :Gridpos
212 END REPeat lp
214 END DEFine
```



Press spacebar for orbit - esc to quit

```
218 DEFine PROCedure StarMove
220  FOR i=1 TO 10:POINT#3,RND(10 TO 160),RND(10 TO 90)
222  PAN#4,-2:PAN#5,2:PAN#6,-2:PAN#7,2
224  SCROLL#4,-2:SCROLL#5,-2:SCROLL#6,2:SCROLL#7,2
226  PAUSE 5
228 END DEFine
```

Note: This is the simple use of PAN and SCROLL although order and size of move are important to get the right balance...

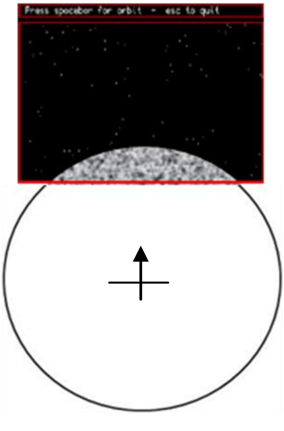


Approching Alpha Centauri

Approching Alpha Centauri

```
232 DEFine PROCedure StarView
234 CLS#8:PRINT#8,' Approaching Alpha Centauri'
236 FOR a=1 TO 8
238   INK 0:FILL 1:CIRCLE 90,70,a*.6:FILL 0
240   x=90:y=70:m=a*.6:Star
242   FOR b=1 TO 6:POINT#3,RND(10 TO 160),RND(10 TO 90)
244   PAN#4,-1:SCROLL#4,-.5
246   PAN#6,-1:SCROLL#6,.5:PAN#7,1:SCROLL#7,.5
248   PAUSE 10
250 END FOR a
252 END DEFine
```

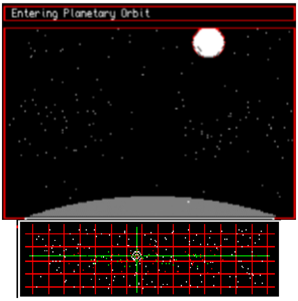
```
256 DEFine PROCedure Star
258  INK 2  :FILL 1:CIRCLE x,y,m*1  :FILL 0
260  INK 241:FILL 1:CIRCLE x,y,m*.95:FILL 0
262  INK 7  :FILL 1:CIRCLE x,y,m*.9 :FILL 0
265  INK 241:FILL 1:CIRCLE x-m*.6,y-m*.4,m*.2,.3,PI/4:FILL 0
266 END DEFine
```



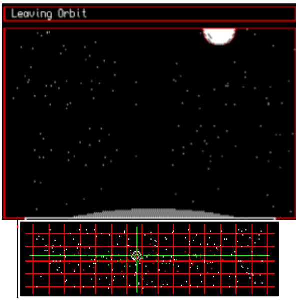
Press spacebar for orbit - esc to quit

The illusion of a rising Planet crest is achieved in SuperBASIC by drawing a large FILLed CIRCLE, where most of it graphics resides off screen. Moving the x y coordinates set off screen upwards and redrawing the circle gives the illusion of a rising crest.

```
270 DEFine PROCedure Eorbit
272 CLS#8:PRINT#8,' Entering Planetary Orbit'
274 FOR i=1 TO 20
276    SCROLL#3,-1:PAUSE 2
278    POINT#3,RND(10 TO 60), RND(5 TO 95)
280 END FOR i
282 PAUSE 5:INK#3,241
284 FOR p=-240 TO -190 STEP 2
286   FILL#3,1:CIRCLE#3,76,p,200:FILL#3,0:SCROLL#3,-1:PAUSE 2
288   POINT#3,RND(10 TO 160),RND(5 TO 10)
290 END FOR p
292 CLS#8:PRINT#8,' Press any key to leave Orbit'
294 END DEFine
```



```
298 DEFine PROCedure Lorbit
300 CLS#8:PRINT#8,' Leaving Orbit'
302 CURSOR#3,0,60
304 FOR i=1 TO 10
306   SCROLL#3,2,2 :POINT#3,RND(40 TO 120),RND(10 TO 90)
308   SCROLL#3,-1,1:PAN#4,-2:SCROLL#4,-1:PAN#5,2:SCROLL#5,-1
310   PAUSE 2
312 END FOR i
314 CLS#8:PRINT#8,' Press spacebar for orbit  -  esc to quit'
316 END DEFine
```



These early code investigations gave a starting point to develop further variations and later incorporate them as part of the full program.

Note on Code Development

I did not consciously consider before writing my Game Programs that I might work to a framework. However, something of such has evolved. It’s not that complicated and seems good sense to use in identifying areas of work.

The premise of the three-act play I use when writing fiction appears to lend itself to developing Game code. Act 1 the Welcome, covering the Intro page, Menu and Initialisation. Act 2 the Game, review of information, taking action, their result. Act 3 the Game End, the Winners recognition, and recording the results.

(1) Welcome:- an **Intro** page to describe the basic game and its function. **Initialisation** of Graphics and Parameters in line with opening choices made from a **Menu.** For **Galaxy AD2370** this is a choice of **Sides** and **Skill Levels**, plus the (**N**)ew (**L**)oad (**S**)ave (**H**)ighscore (**E**)xit much of which was covered by earlier progs.

(2) Challenge:- subdivides into multiple small acts of **Selection** derived from review of information displayed, then taking the appropriate choice of **Action** in order to progress, followed by assessment of the **Outcome** successes or failures.

(3) Game End:- when all the challenges have been met, a sequence depicting the F**inal Battle** between the opposing sides, a declared **Winner** and then some form of **League Table** showing high scores that can be updated when necessary.

QBITS Galaxy AD2370 Intro PART1

An Intro page aim is to give some insight to the Games parameters, role and goals. They can be just printed lines of text. However, there is no reason not to embellish it with a imaginative Title and background imagery to spice things up a little.

QBITS Title

QDOS SuperBASIC commands CSIZE with use of OVER printing to a CURSOR offset presents the opportunity to create 3D style character effects. Trying out different colour arrangements, can produce some pleasing even iconic results.

```
128 WINDOW#1,512,256,0,0 :PAPER#1,0:CLS#1:CSIZE#1,2,1:OVER -1
130 INK#1,2:FOR i=1 TO 3:CURSOR#1,140+i,20-i:PRINT#1,'QBITS Galaxy AD2370'
132 INK#1,7:FOR i=1 TO 2:CURSOR#1,140+i,20-i:PRINT#1,'QBITS Galaxy AD2370'
134 OVER 0:CSIZE 0,0:
```

This code was where I began, experimenting with different INK and i Offset values.

Developing the Opening Scene

Artistic licence or not, I guess this is a matter of choice and limited only by the level of understanding and usage of QL graphics. There are two, possibly more directions in which to create a background. The simple and obvious approach is a static colour pattern, making sure the text printed over it is still legible. A more complex one will involve changing and/or moving images. If the Game has more than one static image or better still a moving graphic sequences, this presents an opportunity to consider them for use as part of the Game Intro.

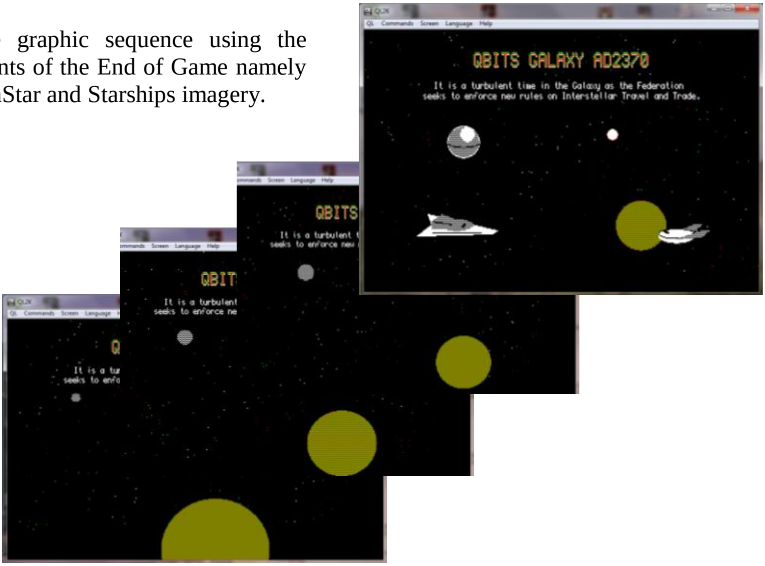
For Galaxy AD2370 I wanted to create an interesting opening. The Title to be followed by a two line statement, setting the scene and printed to screen character by character, all to give the illusion of someone typing an urgent message in real time.

To create this affect I considered the INK colour, CURSOR position and a String length (number of characters). The string variable for Game message Gm$(2,70) is first dimensioned as multiple strings then each string of characters is loaded.

```
110 DIM Gm$(2,70): RESTORE 122:FOR r=1 TO 2:READ Gm$(r)

120 DATA 7,90,50,7,76,60
122 DATA 'It is a turbulent time in the Galaxy as the Federation'
124 DATA 'seeks to enforce new rules on Interstellar Travel and Trade.'

128 RESTOR 120

130 FOR a=1 TO 2
132    READ i,c,r:INK i:CURSOR c,r:FOR b=1 TO 70:PAUSE 1:PRINT Gm$(a,b);
134 END FOR a
```

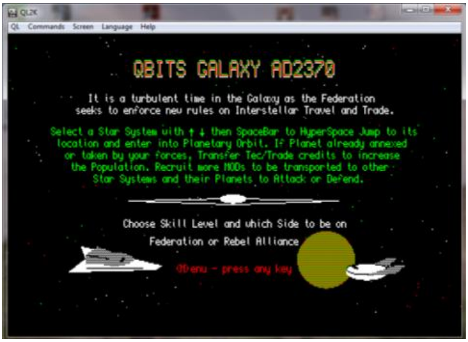
The FOR loop (a) reads the INK colour (i), CURSOR column (c) and row (r), setting the INK colour and CURSOR position. An embedded FOR loop (b) reads each string of characters and prints them one at a time to the screen. Use of the FOR loop is extended in the Galaxy AD2370 code to print further lines of the Intro.

Here the graphic sequence using the components of the End of Game namely the DeathStar and Starships imagery.


QBITS GALAXY AD2370

It is a turbulent time in the Galaxy as the Federation
seeks to enforce new rules on Interstellar Travel and Trade.

QBITS Galaxy Full Introduction
This expresses some of the Games actions and requirements.


QBITS GALAXY AD2370

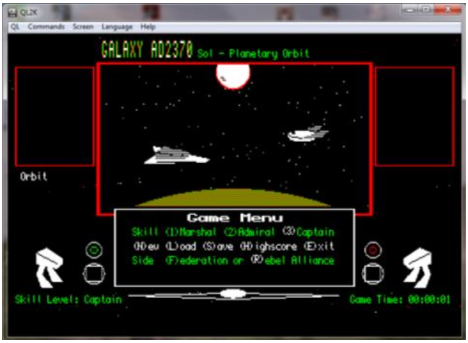
It is a turbulent time in the Galaxy as the Federation
seeks to enforce new rules on Interstellar Travel and Trade.

Select a Star System with ↑ ↓ then SpaceBar to HyperSpace Jump to its
location and enter into Planetary Orbit. If Planet already annexed
or taken by your forces, Transfer Tec/Trade credits to increase
the Population. Recruit more MODs to be transported to other
Star Systems and their Planets to Attack or Defend.

Choose Skill Level and which Side to be on
Federation or Rebel Alliance

(M)enu - press any key

QBITS Galaxy Game Menu
The next step is the player’s choice of **Skill** and the **Side** to play on.


GALAXY AD2370 Sol - Planetary Orbit

Orbit

Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation or (R)ebel Alliance

Skill Level: Captain

Game Time: 00:00:01

QBITS Galaxy Load/Save

Save and Load in Galaxy AD2370 are functions to store and retrieve Game Data so it can be continued later. In previous QBITS programs, these functions have been implemented in a variety of ways. The simplest is a single file, where the QL command PRINT writes to the file opened with OPEN_NEW for Save and the INPUT command reads from the opened file with OPEN_IN for Load.

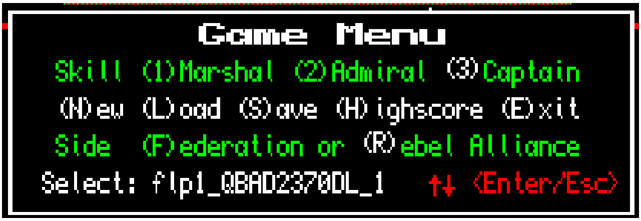
Save

```
120 DELETE device_filename$                                   Delete Old File First
121 OPEN_NEW#99,device_filename$
122 FOR a=1 TO 70
124   FOR b=1 TO 9:PRINT#99,astro(a,b):END FOR b              Galaxy Star Info
125 END FOR a
126 FOR a=1 TO 2
127   FOR b=1 TO 3:PRINT#99,side(a,b) :END FOR b              Galaxy Side Info
128 END FOR a
```

Load

```
130 OPEN_IN#99,device_filename$
131 FOR a=1 TO 70
132   FOR b=1 TO 9:INPUT#99,astro(a,b):END FOR b
133 END FOR a
134 FOR a=1 TO 2
135   FOR b=1 TO 3:INPUT#99,side(a,b) :END FOR b
136 END FOR a
```

For multiple files a little more is required. QBITS Galaxy scrolls through a short list in the form of a specific filename with an appended number QBAD2370DL_1 to 8. A more complex list with differing file names as well as the device used, mdv1_ flp1_ win1_ etc. can also be part of the selection as is the case provided with QBITS FTidy program.



```
Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side  (F)ederation or (R)ebel Alliance
Select: flp1_QBAD2370DL_1   ↑↓ <Enter/Esc>
```

The QBITS default device is flp1_ < this can be changed to users choice > see line 104 of QBITS Galaxy AD 2370 program

While working on the Skill levels and their time delays the idea of a League Table arose. Keeping track of Game Time then became of paramount importance. Therefore it needed to be part of the Game info Saved and when Loaded become the starting point of the continued **Game Time Session (GTS).**

QBITS Galaxy AD2370 Challenge PART 2

What enhances a Game is to be able to access information and take decisions, RISK's to your outcome. Developing a stratagem to achieve success is the player's prerogative. For this to be valid, my first objective was one of displaying the status information of each Side, initialised at the start and then updated as the Game progressed.

Star Systems

The arrays **astro$(70,20)** stores a Star Systems name and **astro(70,9)** holds the details of each. This begins with Technological advance (Tec 1 to 12), Trading abilities (Trade 1 to 20) and Machines of Destruction MODs (1 to 20) for defence. These entries are used in the exchange process of the Transfer Phase of Play.

The next entry is for Population (0.50 to 10 Billion), a random number that is used in calculating the initial Tec/Trade credits and MODs. The fifth entry is ownership, again randomly set for each new game:- Independent=0, Rebel Alliance=1, Federation=2. The values of these entries change as the Game moves forward.

The next four are the Galaxy Map x, y coordinates, Planet surface colour and direction of Orbital approach (These do not change during the Game).

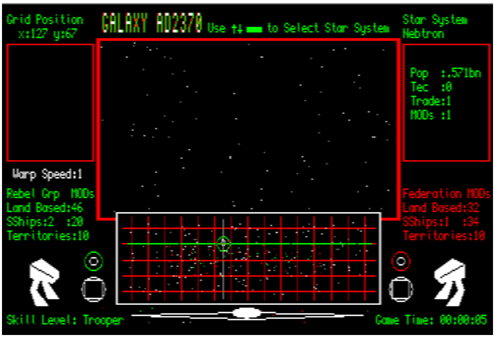
Opponents Status

The array **side(2,3)** holds Tec/Trade and MOD entries for each side. For Attack each player needs a Starship (1 to 5) calculated from the total number of Starship based MODs (1 to 100) and for the Transfer Phase Tec (1 to 24) and Trade (1 to 40) credits. The updated screen displays for this also show the number of annexed Territories (1 to 70).

The Game begins with 10 Territories (Star Systems) allocated to each side, the Rebel Alliance or Federation. Land based MODs are tallied up and any differences compared. If equal, 20 MODs are allocated to the inventory of each opponent's Starships. If not the opponent with the least number of Land based MODs has the difference added to their 20. Therefore, both sides start the Game with the same number of MODs.

Selecting Star Systems

Investigating several arrangements the final method chosen was to simply Scroll through the star list using the Up/Down cursor keys and making a selection with the Spacebar. As the Star Systems are scrolled through, the Galaxy Map position is highlighted by cross wires and a double circle.



```
Grid Position        GALAXY AD2370 Use ↑↓ ▬ to Select Star System   Star System
 x:127 y:67                                                         Nebtron

                                                                     Pop  :.571bn
                                                                     Tec  :0
                                                                     Trade:1
                                                                     MODs :1

 Warp Speed:1
Rebel Grp  MODs                                                     Federation MODs
Land Based:46                                                       Land Based:32
SShips:2  :20                                                       SShips:1  :34
Territories:10                                                      Territories:10

Skill Level: Trooper                                                Game Time: 00:00:05
```

The Grid coordinates are displayed top left, the Star System Name and details on the right::- Population, Tec/Trade and MODs etc.

Space Travel

Space is vast, the average distance across a Galaxy is a 100,000 Light years. Hyperspace or Subspace travel is almost instantaneous, but needs accurate star charts and calculated navigations to plot a course avoiding any collision. Slower warp speeds can take several days, weeks or even months to reach a Star system. Travelling on Impulse (sub light speed) can still take several hours to cross a solar system.

Hyperspace Jump, Warp Speeds & Impulse

For the Galaxy AD2370 Game my premise was to depict the journeys of space travel with a graphic sequence. This became a Hyperspace Jump to the selected Star, followed by a rapid deceleration of Warp Speed on approaching the Star and then a swing into Planetary Orbit on Impulse drive.

Attack Mode

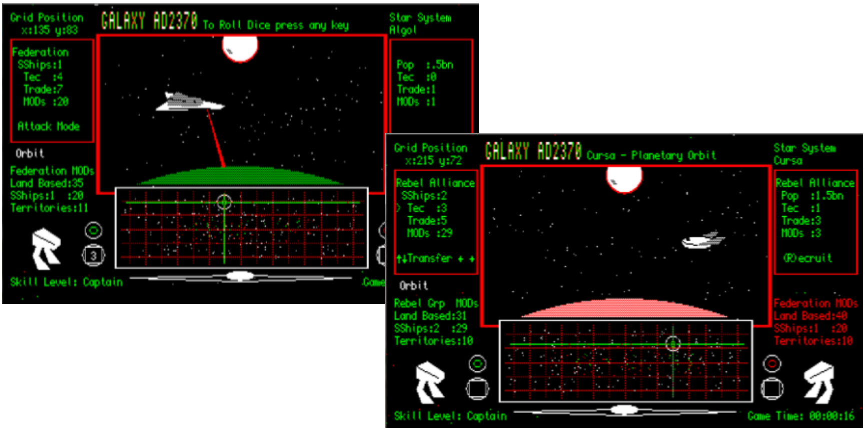
Upon arriving in orbit over a Star Systems Planet if not already annexed then Attack Mode is activated. The Roll of the Dice determines which side loses MODs, the outcome either a defeat for the defending Planet or a retreat into space by the Attackers Starship. If a **Victory** for the attacker, the Star System is annexed and calculations determined by the Planets Population lead to Tec/Trade and MODs being added to the Starship inventory.

Transfer Mode

If the Planet is already annexed or has been defeated, Transfer Mode is activated. Using the cursor keys, Transfer Tec/Trade credits to the Planet. Then use the (**R**)**ecruit** key to increase the Population and generate more MODs.

Tec/Trade and MODs can then be Transferred back to the Starship to strengthen forces for future Attacks. Alternatively, leaving higher numbers of MODs in defence of your annexed Star System makes them less vulnerable to opponent attacks.

Warning:- Transfer Mode and (**R**)**ecuit** might be your only means of raising **MODs** following heavy attack losses.



Grid Position
x:135 y:83
GALAXY AD2370 To Roll Dice press any key
Star System
Algol

Federation
SShips:1
Tec :4
Trade:7
MODs :20

Attack Mode

Pop :.5bn
Tec :0
Trade:1
MODs :1

Orbit
Federation MODs
Land Based:35
SShips:1 :20
Territories:11

Skill Level: Captain

Grid Position
x:215 y:72
GALAXY AD2370 Cursa - Planetary Orbit
Star System
Cursa

Rebel Alliance
SShips:2
> Tec :3
Trade:5
MODs :29

↑↓Transfer ← →

Rebel Alliance
Pop :1.5bn
Tec :1
Trade:3
MODs :3

(R)ecruit

Orbit
Rebel Grp MODs
Land Based:31
SShips:2 :29
Territories:10

Federation MODs
Land Based:40
SShips:1 :20
Territories:10

Skill Level: Captain
Game Time: 00:00:16

QBITS Galaxy AD2370 CPlay

For the computer to act as a worthy opponent, combining stratagems and logic to create a workable code required a good amount of forethought. My programming skills being limited, no way was this going to approach anything like Artificial Intelligence.

The obvious route to follow was the actions dictated by the Game, namely seek out a Star System, Attack and/or Transfer Tec/Trade credits, recalculate the Population and thereby generate and Recruit more MODs. However, if this is to be truly representative, then the same fortunes and misfortunes when Rolling the Dice must apply as to a human player.

The Pro's and Con's of Conflict

At Game start if the computer is the one awarded the extra MODs to their Starship fleet, this is a straight advantage when sallying forth to Attack Star Systems. Yet the fickle hand of fate when Rolling the Dice can be both fortuitous and equally as quick lead to a pending disaster.

To guard against the worst outcomes, avoidance mechanisms needed to be part of the Computer player's code. For example before an Attack, the number of Starship MODs should be maybe three or four times larger than the Star Systems defence MODs. As the Game progresses if the Computers Starships MODs fall low in value, the computer player should avoid attacking Star Systems and seek to recruit more MODs from Transfers with its own Star/Planet Systems. To a Human player these actions can be a relatively easy decision to make. For the Computer player however, what criteria does it use, does it involve analysing and weighing the Pro's and Con's of each and every Star System against some master plan of action or keep it as a set of simple random choices.

Computer Interaction

How and when should the Computer interact? I considered a number of options. The first and obvious is a straight flip-flop between Human and Computer. When working with long time delays, the Computer player reliant on random number choices, placed it at a distinct disadvantage. Potentially lesser actions of value would be accomplished leading to a relatively easy win by most human players.

Alternatives

Here my thoughts again were on letting the computer compete for access in the main Game loop by control of a random number. A short experimentation with this form of access depending on choice and range of random numbers, proved either ineffectual or at times allowed the computer to block any human opponents interaction leading to an impossible task of a human player ever winning.

Variable Level

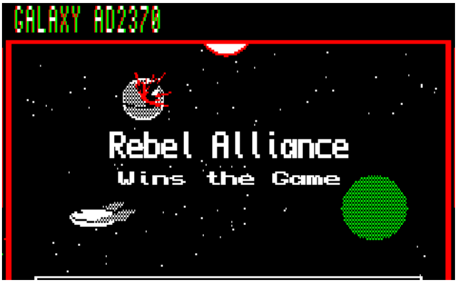
In the end, I decided on a time delay that locked the Computer out of play rather than a straight flip-flop or reliance on a random period. By increasing or decreasing the period of delay, the time for the human player to complete their actions gave the guise of a variable Skill level.

QBITS Galaxy AD2370 Game End PART 3

A few thoughts on winning, essentially the winner's opponent has to have lost all of their Star System Territories and Land based MODs. Although they may still have Starship based MODs and the slimmest of chances of a comeback, I decided to base a WIN on the losing opponent having had their Territories reduced to zero.

Final Battle

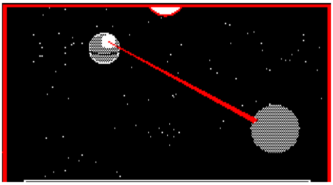
Creating an End of Game sequence, the Final Battle between Rebel Alliance and Federation had to involve the Death Star. Then with the chance of either side winning, I had to consider two possible outcomes. The Death Star blown up or the Rebels home Planet and last refuge totally destroyed.



GALAXY AD2370

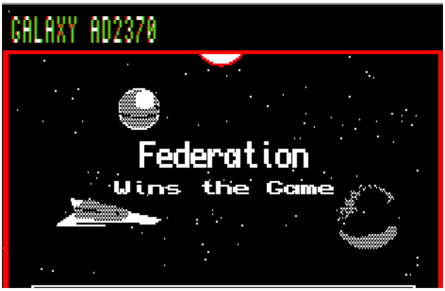
Rebel Alliance

Wins the Game



Declared Winner

To accomplish this is just a printed declaration blazed across the screen. '**Rebel Alliance or Federation Wins the Game**' and displaying the relative Starship.



GALAXY AD2370

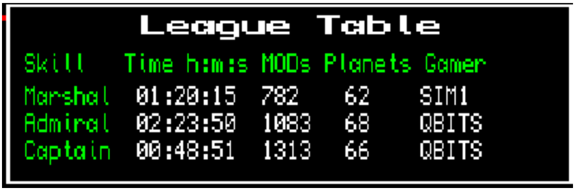
Federation

Wins the Game

League Table

Having met the challenge and acquired a final victory, it's not unexpected to want to match you results against others. I therefore added a **League Table** to satisfy this Gamer prerogative. **QBAD2370DLT** Loads at initialisation and with a new high scorer the program allows updates to be made and then Saved automatically.

.



League Table

Skill	Time h:m:s	MODs	Planets	Gamer
Marshal	01:20:15	782	62	SIM1
Admiral	02:23:50	1083	68	QBITS
Captain	00:48:51	1313	66	QBITS

QBITS Galaxy AD2370 Testing

Some Galaxy AD2370 checks can be made with the test code built into the program.

(**F1**) checks the screen sequences, a Highperspace Jump, Star Approach and Planetary Orbit, followed with a test firing of the selected forces Starships laser weapon. Then after a short pause, the Starship leaves orbit. The final battle sequence with the approach of the DeathStar to the Rebels Planet now takes place.

Depending on which force is chosen from the menu Federation or Rebel Alliance, either the DeathStar discharges its weapon and destroys the planet or the DeathStar itself is destroyed. At this point, the Galaxy map shows the status of all the annexed star systems. Then after another delay, switches to show the high score **League Table**.

(**F2**) activates the **Simulator Mode** this disables the human player access and assumes their role. The simulation mode now plays against the actions of the **Computer player**.

The League Table

As the high score results are held on file, this required some means of resetting to zero and checking its function with regarded to updates. To this end a test program is provided that can be merged with the existing Galaxy program.

LOAD QBAD2370v4DL01 and **MERGE** Test Program **QBAD2370DLTP.** RUN the program and select (**N**) from **Game Menu.** Following initialisation press **CTRL-Spacebar** to halt the program. Type **TP** at QL Command line and press ENTER this should reset the **League Table** entries to all zeros and clear any names. To check run Galaxy AD2370 again and select (**H**) from Menu.

Further checks - as supplied **QBAD2370v4DLTP** has a number of lines REMarked out. By removing the REMark on certain lines, you can set new values and force an entry on any of the different Skill levels of the **League Table**.

LName	Checks skill level and if new high score allows name entry.
LSave and **LLoad**	These **Save / Load** the **QBAD2370DLT** high score file.
LScore	Displays the **League Table** and current entries.
GTS	**G**ame **T**ime **S**ession
GSk=	**Skill** Level (**1**) Marshal (**2**) Admiral (**3**) Captain
GEnd	Runs **End Game** sequence.

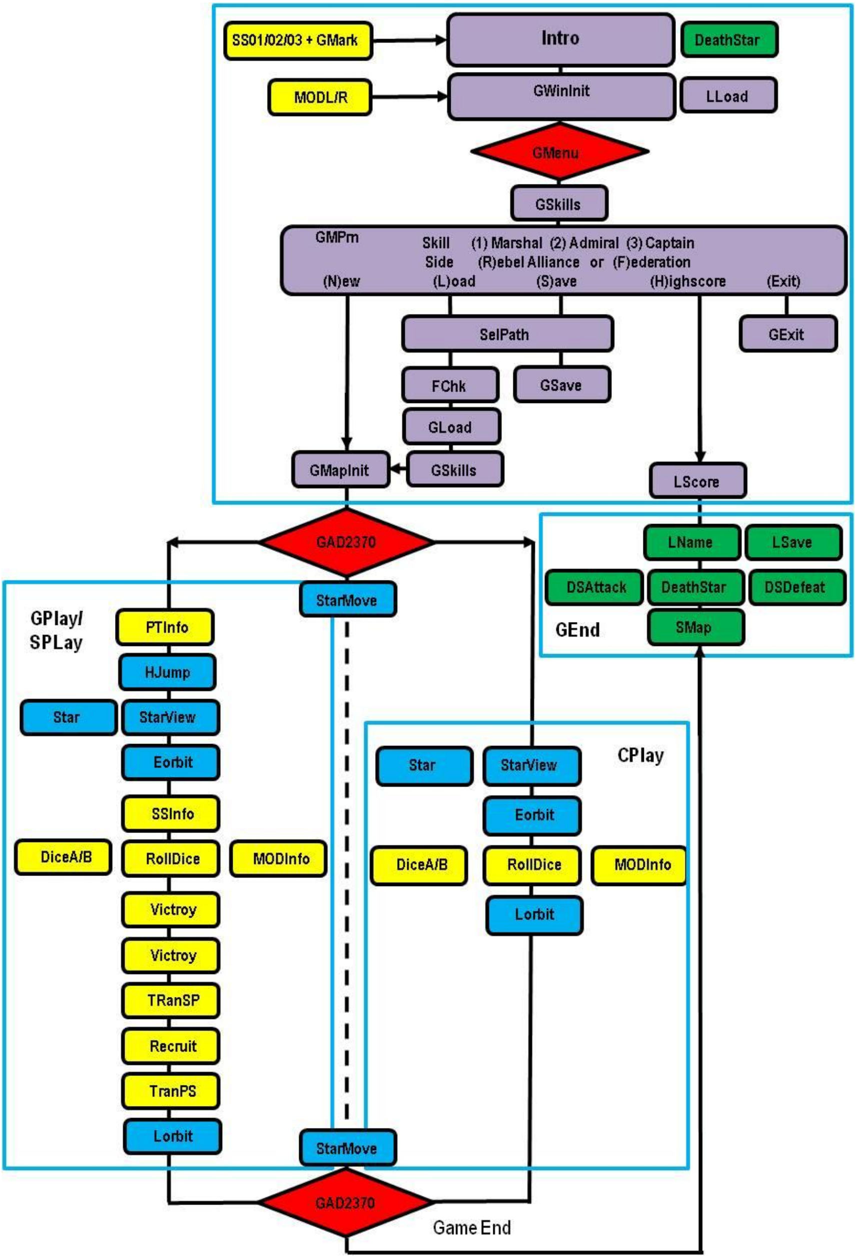
Note: Skill Level, Game Time Session, MODs, Territories and Gamer Name are contained within file **QBAD2370v4DLT**. Saved after any update and Loaded before Initialisation of Game.

QBIT Galaxy ADA2017 PROCedures

Intro	Introduction screen to the Game
InitWin	Initialise the screens (WINDOWS 3 to 18)
GMark	Galaxy Mark draws Rebel Alliance icon
SS01/2/3	StarShip image SS01 / SS02 / SS03
MODL/R	MODL - Left image /MODR - Right image
RollDice	Dice throw - random numbers 1 to 8
DiceA/B	Square image / Diamond image
GMenu	Choice of **Side** /**Skill** / (**N**)ew (**L**)oad (**S**)ave (**H**)ighscore (**E**)xit
GSkill	Get Skill Level
GMPrn	Display Menu choices
GExit	Close unused channels clear screen...
GNewInit	Initialise the Star / Planet DATA
InitMap	Initialise the Galaxy Map Grid & Star Systems
GLoad/GSave	**Save /Load** Game Data file
SelPath/FChk	Select File **QBAD2370DL_1-8**, File checks on device [flp1_]
GNotes	Galaxy AD2370 Stratagems / Notes
GameAD2370	Main Game Loop
StarMove	PAN / SCROLL of screens 3 to 10 (Space View)
PMap	Displays Game Players Star System Grid location on Galaxy Map
CMap	Displays Computer Players Star System on Galaxy Map
HJump	HyperSpace Jump [Warp 12] to Star System
StarView/ Star	Star Approach [Warp 1] - Enlarging Star
Eorbit	Takes Star Ships in to Planetary Orbit (Impulse)
Lorbit	Leaves Orbit.
GPlay	Player decisions over Attack/ Transfer – Tec/Trade/MODs etc.
MODInfo	MODs distribution of Rebel Alliance **/** Federation Starships &Planets
PTInfo	Planet :- Pop/Tec/Trade/ MODs
SSInfo	StarShip:- Tec/Trade/ MODs
Phaser	Fires Phasers down on to Planet
Victory	Updates Attacking Sides Tec/Trade Credits and MODs
TranPS	Transfer Tec/Trade/MODs Planet >Starship
TranSP	Transfer Tec/Trade/MODs Starship>Planet
Recruit	Re-Calculate Pop & MODs after change in Tec/Trade credits
GEnd	Game End Winner sequence – returns to Game Menu
DeathStar	DeathStar arrival at a Planet location
DSAttack	Deathstar destroys Planet
DSDefeat	Deathstar is itself destroyed
LMap	League Table Rebel Alliance or Federation annexed Star Systems
LScore	Displays the **League Table** of high scores
LName/ LSave/ LLoad	New Name entry, Save & Load **QBAD2370DLT**
CPlay	Computer Player Attack/ Transfer Tec/Trade/MODs
SPlay	Simulated Player Attack/ Transfer Tec/Trade/MODs

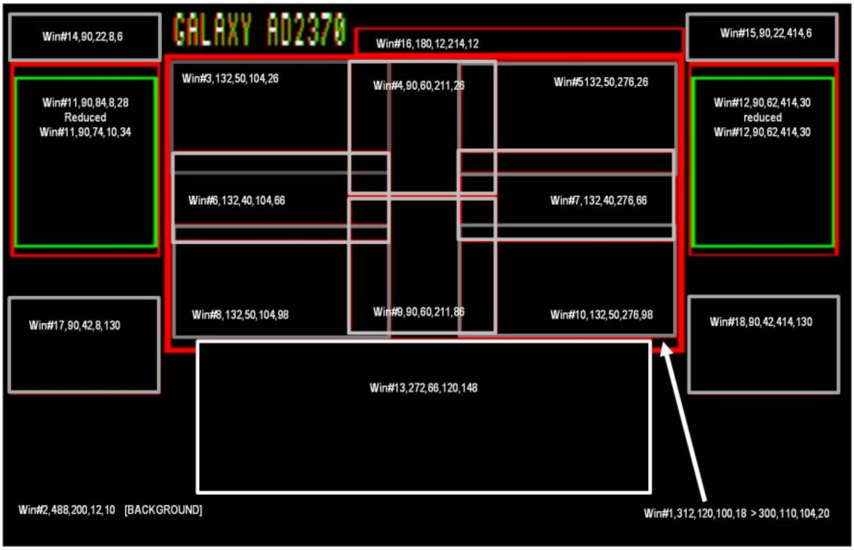
Note: SPlay and **CPlay** can play against each other to check strength of Stratagems.

QBITS Galaxy AD2370 Flow Chart



```
100 REMark QBAD2370v4DL01 - QBITS Galxay AD2370 v4.01 2017                              PART1

102 DIM Dat$(8,12)                  :REMark Game Data Files 1 to 8
103 FOR df=1 TO 8:Dat$(df)='QBAD2370DL_'&df
104 Dn$='flp1_':df=1                :REMark Devicename [Change to User preference]
105 m=0:slk=1:file=1                :REMark Game Data file New/Load/Save checks
106 LTFile$=Dn$&'QBAD2370DLT'       :REMark League Table file

108 DIM astro$(70,20),astro(70,9)    :REMark Star/Planet 1-70 DATA
109 DIM side(2,3),Sclk$(20)         :REMark Game Players 1-2 DATA : GTS h:m:s
110 DIM name$(4,10),score(3,3)      :REMark HighScore League Table
111 DIM Gm$(10,70):RESTORE 124:FOR l=1 TO 10:READ Gm$(l)

113 Gp=0:Gp1=0:Gp2=0:xsd=54:sd$='(R)'   :REMark Game Player checks
114 RMOD=0:RSSn=1:Rt=10             :REMark Rebel Alliance MODs:SShips:Territories
115 FMOD=0:FSSn=1:Ft=10             :REMark Federation    MODs:SShips:Territories
116 en=0:s=1:t=1:zr=1:Lck=0         :REMark RollDice:Attack/Transfer:(R)ecruit:LName
117 dm=3:ws=1:Mes$='          '     :REMark delete MODs:Warp Speed:Messages
118 GTS=0:Tm1=60:Tm2=20:Test=0      :REMark Game Time :Test=0/1 for checks

120 Mode 4:Intro:GWinInit:GMenu     :REMark Title - Game Menu

122 DATA 7,90,50,7,76,60,4,48,78,4,54,88,4,64,98,4,70,108,4,96,118
123 DATA 7,130,156,7,160,171,2,188,194
124 DATA 'It is a turbulent time in the Galaxy as the Federation'
125 DATA 'seeks to enforce new rules on Interstellar Travel and Trade.'
126 DATA 'Select a Star System with ¾ ¿ then SpaceBar to HyperSpace Jump to its'
127 DATA 'location and enter into Planetary Orbit. If Planet already annexed'
128 DATA 'or taken by your forces, Transfer Tec/Trade credits to increase'
129 DATA 'the Population. Recruit more MODs to be transported to other'
130 DATA 'Star Systems and their Planets to Attack or Defend.'
131 DATA 'Choose Skill Level and which Side to be on'
132 DATA 'Federation or Rebel Alliance','(M)enu - press any key'

134 DEFine PROCedure Intro
135 WINDOW#1,512,256,0,0:PAPER#1,0:CLS#1:PAPER#2,0:CLS#2:SCALE#1,150,0,-10
136 FOR i=1 TO 250:INK RND(2 TO 7):POINT RND(0 TO 230),RND(0 TO 140)
137 CSIZE#1,2,1:OVER#1,-1
138 FOR i=2 TO 4:INK#1,i*2:CURSOR#1,138+i,18+i:PRINT#1,'QBITS GALAXY AD2370'
139 CSIZE#1,0,0:OVER#1,0:RESTORE 122
140 FOR a=1 TO 2
141   READ i,c,r:INK i:CURSOR c,r:FOR b=1 TO 70:PAUSE 1:PRINT Gm$(a,b);
142 END FOR a
143 n=20:astro$(n)='Sol':astro(n,5)=1:astro(n,8)=243:Gp=1:x=140:y=80:m=3
144 Star:DeathStar:ink1=7:ink2=241:x=186:y=26:SS02:x=30:y=22:SS03
145 WINDOW#2,496,220,8,10:PAPER#2,0:INK#2,7:CSIZE#2,0,0:PAUSE 50
146 x=83:y=40:LINE#2,x-40,y TO x,y+1 TO x+40,y TO x,y-1 TO x-40,y
147 FILL#2,1:CIRCLE#2,x,y+1,5,.3,PI/2:FILL#2,0
148 FOR a=3 TO 10:READ i,c,r :INK i:CURSOR c,r :PRINT Gm$(a):PAUSE 5
149 PAUSE :CURSOR#2,0,116:FOR a=1 TO 16:SCROLL#2,-8,1:SCROLL#2,5,2:PAUSE 3
150 END DEFine
```

Galaxy Symbol



```
152 DATA 132,50,104,26,90,60,211,26,132,50,276,26      :REMark win 3-10 View screen
153 DATA 132,40,104,66,132,40,276,66
154 DATA 132,50,104,98,90,60,211,86,132,50,276,98
155 DATA 90,84,8,28,90,84,414,28,272,66,120,148         :REMark win 11-13 Starship / Planet /Galaxy Map
156 DATA 90,22,8,6,90,22,414,6,198,12,214,12            :REMark win 14-16 Star x,y / Name / Message
157 DATA 90,42,8,130,90,42,414,130                      :REMark win 17-18 Side Status

159 DEFine PROCedure GWinInit
160 file=1:Gp1=1:Gp2=2:LLoad:RESTORE 152
161 OPEN#3,scr_496x20a8x4:CLS#3:CSIZE#3,1,1:OVER#3,-1
162 FOR i=1 TO 2:INK#3,i*2:CURSOR#3,96+i,0:PRINT#3,'GALAXY AD2370'
163 OVER#3,0:CLOSE#3
164 FOR i=3 TO 18:OPEN#i,scr_:READ a,b,c,d:WINDOW#i,a,b,c,d
165 WINDOW#1,312,130,100,24:BORDER#1,2,2:PAPER#1,0:CLS#1
166 WINDOW#1,300,120,106,26:WINDOW#0,496,20,8,230
167 BORDER#11,1,2:WINDOW#11,86,74,10,34
168 BORDER#12,1,2:WINDOW#12,86,74,416,34
169 BORDER#13,1,7:PAPER#13,0:CLS#13
170 INK#2,7:CIRCLE#2, 30,21,1.25:INK#2,4:CIRCLE#2, 30,21,3
171 INK#2,7:CIRCLE#2,136,21,1.25:INK#2,2:CIRCLE#2,136,21,3
172 MODL:MODR:l=2.9:dx=30:DiceA:dx=136:DiceA
173 sn=n:StarView:Eorbit:x=30:y=30:SS03
174 END DEFine
```

Game Menu

This is where you choose the **Skill** level to play (**1**)Marshal - (**2**)Admiral - (**3**)Captain and decide which **Side** (**R**)ebel Alliance or (**F**)ederation to play on. (**S**)ave at this point is disabled but activated following a (**N**)ew Game initialisation or (**L**)oad of a previously saved one.

(**H**)ighscore – **QBAD2370DLT** the current **League Table** file is loaded at start up. The Winners **GTS** (**G**ame **Ti**me **S**ession) is shown with Total MODs, Territories annexed and the name, nickname or pseudonym (up to 9 characters) of the current holder.

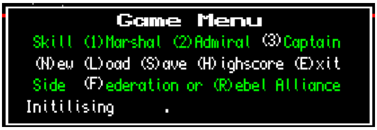
```
176 DEFine PROCedure GSkill
177 IF Gp1=1:xsd=54:sd$='(R)':ELSE xsd=174:sd$='(F)'
178 IF Tm1=20:Tm$='Marshal' :GSk=1:xsk= 54:sk$='(1)'
179 IF Tm1=40:Tm$='Admiral' :GSk=2:xsk=120:sk$='(2)'
180 IF Tm1=60:Tm$='Captain' :GSk=3:xsk=186:sk$='(3)'
181 INK#2,4:AT#2,21,0:PRINT#2,'Skill Level: ';Tm$
182 AT#2,21,63:PRINT#2,'Game Time: '
183 END DEFine

185 DEFine PROCedure GMPrn
186 ch=13:CSIZE#ch,0,0:INK#ch,4
187 CURSOR#ch,18, 14:PRINT#ch,'Skill (1)Marshal (2)Admiral (3)Captain'
188 CURSOR#ch,18,38:PRINT#ch,'Side  (R)ebel Alliance or (F)ederation'
189 INK#ch,7:CURSOR#ch,18,26:PRINT#ch,'(N)ew (L)oad (S)ave (H)ighscore (E)xit'
190 CURSOR#ch,xsk,13:PRINT#ch,sk$:CURSOR#ch,xsd,37:PRINT#ch,sd$
191 END DEFine

193 DEFine PROCedure GMenu
194 ch=13:CLS#13:CSIZE#ch,2,0:INK#ch,7:OVER#ch,1
195 FOR i=1 TO 2:CURSOR#ch,78+i,2:PRINT#ch,'Game Menu'
196 OVER#ch,0:GSkill:GMPrn
197 REPeat lp
198  k=CODE(INKEY$(-1))
199  SELect ON k
200   =49:Tm1=20:GSkill:GMPrn                :REMark Marshal
201   =50:Tm1=40:GSkill:GMPrn                :REMark Admiral
202   =51:Tm1=60:GSkill:GMPrn                :REMark Captain
203   =70,102:Gp1=2:Gp2=1:GSkill:GMPrn       :REMark Federation
204   =82,114:Gp1=1:Gp2=2:GSkill:GMPrn       :REMark Rebel Alliance
205   =78,110:slk=0:m=1:GNewInit             :REMark (N)ew
206   =76,108:SelPath  :GLoad                :REMark (L)oad
207   =83,115:IF slk=0:SelPath:GSave         :REMark (S)ave
208   =72,104:LScore:PAUSE:GMenu             :REMark (H)ighscore
209   =69,101:GExit                          :REMark (E)xit
210   END SELect
211 END REPeat lp
212 END DEFine

214 DEFine PROCedure GExit
215 CLOSE#99:FOR w=3 TO 18:CLOSE#w
216 INK#2,7:INK#1,7:WINDOW#1,512,256,0,0:PAPER#1,0:CLS#1
217 INK#0,7:WINDOW#1,490,220,16,8:PRINT#0,'bye...':STOP
218 END DEFine
```

A New Game requires seeding the Galaxy Map with Star Systems and their details: Population, Tec/Trade credits and MODs to defend themselves.

Creating a name list of stars required poring over several star charts and then taking a quite liberal use of names, altering a few and making up one or two others. The DATA distribution is deliberate so as not to end up in alphabetic order or anything like.

```
220 DATA 'Vega','Bellatrix','Zibal','Castor','Naos','Artos','Ilad'
221 DATA 'Tabit','Diphda','Elnath','Acrux','Errai','Cellus','Detrus'
222 DATA 'Polaris','Homan','Rigel','Sabik','Sarin','Westron','Zenda'
223 DATA 'Furud','Gienah','Hadar','Propus','Izar','Centra','Nebtron'
224 DATA 'Enif','Algol','Mirzarm','Libeta','Atria','Panus','Ulan'
225 DATA 'Keid','Kochab','Lesath','Ankaa','Marsic','Nebus','Alean'
226 DATA 'Meissa','Subra','Menkib','Altair','Muscida','Pusan','Curson'
227 DATA 'Cursa','Nashira','Nunki','Ogma','Pollux','Xilon','Scarb'
228 DATA 'Sirius','Merack','Serena','Denab','Tureis','Noran','Theron'
229 DATA 'Beid','Wesen','Yildun','Caph','Zooma','Jarron','Cetrus'
```

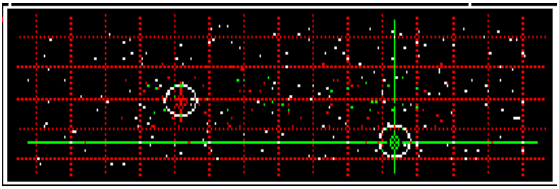
(**N**)ew Initialising Game



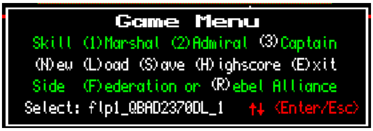
Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation or (R)ebel Alliance
Initilising .

```
231 DEFine PROCedure GNewInit
232 RESTORE 219:ch=13:CURSOR#ch,12,50:PRINT#ch,'Initilising';:CLS#ch,4
233 FOR n=1 TO 70
234    READ a$:astro$(n)=a$:CURSOR#ch,80+n*2,50:PRINT#ch,'.';:PAUSE 1
235    pop=RND(2 TO 10)/RND(2 TO 8):IF pop<.5:pop=.5
236    astro(n,1)=INT(pop*1.2)         :REMark Tec
237    astro(n,2)=INT(pop*2)           :REMark Trade
238    astro(n,3)=INT(pop*2)           :REMark MODs
239    astro(n,4)=pop                  :REMark Pop
240    astro(n,5)=0                    :REMark Side
241    astro(n,6)=n*4+15               :REMark x coordinate
242    astro(n,7)=RND(10 TO 90)        :REMark y coordinate
243    astro(n,8)=RND(220 TO 254)  :REMark Planet Colour
244    astro(n,9)=RND(1 TO 3)          :REMark Orbit Approach
245 END FOR n
246 REMark astro(n,5) 0=Independent 1=Rebel Alliance 2=Federation
247 FOR i=1 TO 10
248    n=RND(15 TO 55):IF astro(n,5)=0:astro(n,5)=2:ELSE GO TO 247
249 END FOR i
250 FOR i=1 TO  5
251    n=RND( 5 TO 20):IF astro(n,5)=0:astro(n,5)=1:ELSE GO TO 250
252    n=RND(50 TO 65):IF astro(n,5)=0:astro(n,5)=1:ELSE GO TO 251
253 END FOR i
254 FOR n=1 TO 70:IF astro(n,5)>0 AND astro(n,3)<3:astro(n,3)=3
255 FOR Gp=1 TO 2 :side(Gp,1)=3:side(Gp,2)=5:side(Gp,3)=0
256 GMapInit:GTS=0:GAD2370
257 END DEFine
```

```
259 DEFine PROCedure GMapInit
260 ch=13:CLS#ch:BORDER#ch,1,7:INK#ch,0,2,3:SCALE#ch,100,0,0
261 FOR h=12 TO 96 STEP 18
262  LINE#ch,6,h TO 310,h
263 END FOR h
264 FOR v=16 TO 330 STEP 20
265  LINE#ch,v,3 TO v,98
266 END FOR v
267 INK#ch,2:FOR i=1 TO 60 :POINT#ch,RND(60 TO 260),RND(30 TO 70)
268 INK#ch,4:FOR i=1 TO 30 :POINT#ch,RND(80 TO 240),RND(40 TO 60)
269 INK#ch,7:FOR i=1 TO 180:POINT#ch,RND(10 TO 300),RND(10 TO 90)
270 ch=16:CLS#ch:INK#ch,4
271 PRINT#ch,'Use ¾¿    to Select Star System':BLOCK#ch,16,4,40,4,4
272 ch=1:CLS#ch:INK#ch,7:FOR i=1 TO 120:POINT#ch,RND(10 TO 200),RND( 5 TO 95)
273 END DEFine
```

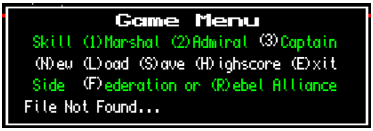


```
275 DEFine PROCedure SelPath
276 ch=13:CSIZE#ch,0,0:INK#ch,2:CURSOR#ch,0,50:CLS#ch,4
277 CURSOR#ch,180,50:PRINT#ch,'⇧⇩ <Enter/Esc>':INK#ch,7
278 REPeat Path_lp
279  CURSOR#ch,12,50:PRINT#ch,'Select: ';Dn$&Dat$(df)
280  k=CODE(INKEY$(-1))
281  SELect ON k
282   =208:df=df+1:IF df>8:df=1
283   =216:df=df-1:IF df<1:df=8
284   = 10:file=1:device_filename$=Dn$&Dat$(df):EXIT Path_lp
285   = 27:file=0:EXIT Path_lp
286  END SELect
287 END REPeat Path_lp
288 device_filename$=Dn$&Dat$(df)
289 END DEFine
```



```
Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side  (F)ederation or (R)ebel Alliance
Select: flp1_QBAD2370DL_1   ↑↓ <Enter/Esc>
```

SelPath selects filename for Save/ Load

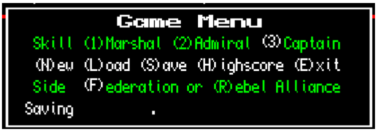


```
Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side  (F)ederation or (R)ebel Alliance
File Not Found...
```

FChck returns **Not Found** if not previously saved.

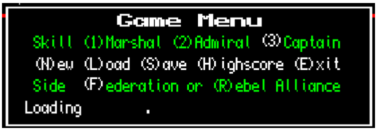
```
291 DEFine PROCedure FChck
292 ch=13:CURSOR#ch,12,50:PRINT#ch,'Searching...':CLS#ch,4
293 DELETE Dn$&'FList'
294 ch=99:OPEN_NEW#ch,Dn$&'FList':DIR#ch,Dn$:CLOSE#ch
295 OPEN_IN#ch,Dn$&'FList'
296 REPeat Dir_lp
297   IF EOF(#ch)
298      CLOSE#ch:ch=13:CURSOR#ch,12,50:PRINT#ch,'File Not Found...'
299      PAUSE 50:CURSOR#ch,0,50:CLS#ch,4:file=0:slk=1:RETurn
300   END IF
301   INPUT#ch,Fchk$:IF Fchk$==Dat$(df):CLOSE#ch:EXIT Dir_lp
302 END REPeat Dir_lp
303 ch=13:CURSOR#ch,12,50:PRINT#ch,'Loading':CLS#ch,4
304 END DEFine

306 DEFine PROCedure GSave
307 ch=13:IF file=0:CURSOR#ch,0,50:CLS#ch,4:RETurn
308 DELETE device_filename$
309 CURSOR#ch,12,50:PRINT#ch,'Saving':CLS#ch,2:CLS#ch,4
310 OPEN_NEW#99,device_filename$
311 FOR a=1 TO 70
312   CURSOR#ch,66+a*2,50:PRINT#ch,'.';:PRINT#99,astro$(a):PAUSE 1
313   FOR b=1 TO 9:PRINT#99,astro(a,b):END FOR b
314 END FOR a
315 FOR a=1 TO 2
316   FOR b=1 TO 3:PRINT#99,side(a,b) :END FOR b
317 END FOR a
318 PRINT#99,GTS\Tm1\Gp1\Gp2:CLOSE#99:CURSOR#ch,0,40:CLS#ch,2
319 END DEFine
```

(S)ave Game Data to file


Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation or (R)ebel Alliance

Saving .

(**L**)oad Game Data File


Game Menu
Skill (1)Marshal (2)Admiral (3)Captain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation or (R)ebel Alliance

Loading .

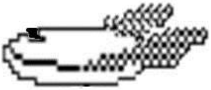
```
321 DEFine PROCedure GLoad
322 ch=13:IF file=0:CURSOR#ch,0,50:CLS#ch,4:RETurn
323 FChck:IF file=0:CURSOR#ch,0,50:CLS#ch,4:RETurn
324 OPEN_IN#99,device_filename$
325 FOR a=1 TO 70
326   CURSOR#ch,72+a*2,50:PRINT#ch,'.';:INPUT#99,astro$(a):PAUSE 1
327   FOR b=1 TO 9:INPUT#99,astro(a,b):END FOR b
328 END FOR a
329 FOR a=1 TO 2
330   FOR b=1 TO 3:INPUT#99,side(a,b) :END FOR b
331 END FOR a
332 INPUT#99,GTS\Tm1\Gp1\Gp2:CLOSE#99:CURSOR#ch,0,40:CLS#ch,2
333 slk=0:GSkill:GMapInit:GAD2370
334 END DEFine
```

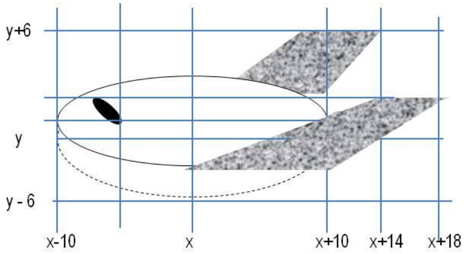
QBITS Galaxy AD2370 Starships & MODS

A number of ideas to incorporate Starship images was part of my agenda. Creating these can be done in several ways, with a pixel by pixel grid, generated from a scanned image or simply by use of SuperBASIC commands ARC, CIRCLE, LINE, POINT. That is not forgetting the accompanying, INK, FILL, OVER, PI, and use of SCALE.

Creating a Starship

The range of designs are as you would expect numerous and can get quite complex, my aim was to keep it as simple as possible. Starting with a few doodles to identify a usable image, it was then broken down into possible sections. As a child I remember being busy for hours linking up sets of points to create a picture. With a little imagination, employing this technique an image can be broken down into a number of shapes using LINE and CIRCLE commands. Then the image can be build from overlapping layers.

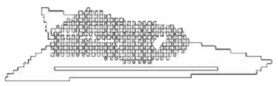




The one shown has 5 overlapping shapes, a filled quadrilateral for the far wing, a filled white circle for the main body with a black ink circle drawn over the top as an outline. A black filled circle for the flight deck. A 2nd quadrilateral for the nearside wing.

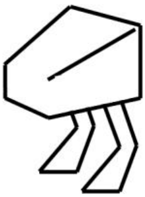
Individual elements of an image are coded separately within their own PROCedure where the elements of LINE, CIRCLE etc. are grouped together with colour settings and x, y off-sets to global coordinates.

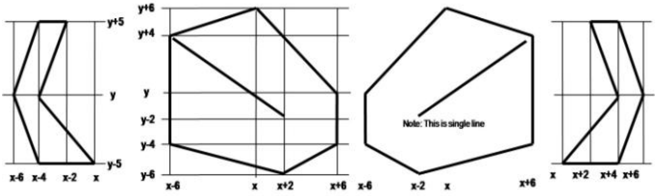
Being able to set the ink colours deployed in the image Procedure allows the useful possibility to redraw with all colours set to the background, so it can in affect be made to disappear.



Creating the MODs

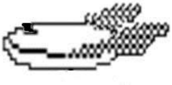
The LINE values again are straightforward to work out.



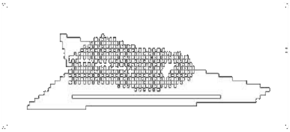


The other shapes can be generated in much the same way see opposite page -

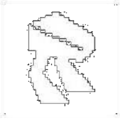
```
336 DEFine PROCedure SS01
337 ch=1:REMark DeathStar
338 INK#ch,ink2:FILL#ch,1:CIRCLE#ch,dx,dy,9:FILL#ch,0
339 INK#ch,ink1:CIRCLE#ch,dx,dy,9:INK#ch,0:ARC dx-12,dy TO dx+12,dy,PI/3
340 INK#ch,ink1:FILL#ch,1:CIRCLE#ch,dx+2,dy+4,4:FILL#ch,0
341 END DEFine
```



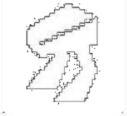
```
343 DEFine PROCedure SS02
344 ch=1:INK#ch,ink2:REMark Rebel SShip x=120:y=40
345 FILL#ch,1:LINE#ch,x-6,y TO x+10,y+6 TO x+14,y+6 TO x+6,y TO x-8,y:FILL#ch,0
346 INK#ch,ink1:FILL#ch,1:CIRCLE#ch,x,y,10,.4,PI/2:FILL#ch,0
347 INK#ch,0:CIRCLE#ch,x,y+1,10,.28,PI/2:INK#ch,ink2
348 FILL#ch,1:LINE#ch,x,y-2 TO x+14,y+3 TO x+18,y+3 TO x+14,y-1 TO x,y-2:FILL#ch,0
349 INK#ch,0:FILL#ch,1:CIRCLE#ch,x-6,y+2,2,.3,PI/3:FILL#ch,0
350 END DEFine
```



```
352 DEFine PROCedure SS03
353 ch=1:REMark Federation SShip x=50:y=40
354 INK#ch,ink1:FILL#ch,1:LINE#ch,x,y TO x+8,y+6 TO x+6,y+12 TO x+45,y+3 TO x+42,y+1 TO x,y:FILL#ch,0
355 INK#ch,ink2:FILL#ch,1:LINE#ch,x+8,y+8 TO x+14,y+12 TO x+24,y+8 TO x+20,y+4 TO x+8,y+4 TO x+8,y+8:FILL#ch,0
356 FILL#ch,1:CIRCLE#ch,x+22,y+6,9,.35,PI/2:FILL#ch,0:INK#ch,0
357 FILL#ch,1:CIRCLE#ch,x+26,y+6,.5,3,PI/4:FILL#ch,0
358 LINE#ch,x+8,y+8 TO x+20,y+6:LINE#ch,x+36,y+1.5 TO x+8,y+1.5
359 END DEFine
```



```
361 DEFine PROCedure MODL
362 ch=2:INK#ch,7:x=18:y=14
363 x=x:y=y:FILL#ch,1:LINE#ch,x,y-5 TO x-4,y-5 TO x-6,y TO x-4,y+5 TO x-2,y+5 TO x-4,y TO x,y-5:FILL#ch,0
364 x=x-4:y=y-2:FILL#ch,1:LINE#ch,x,y-5 TO x-4,y-5 TO x-6,y TO x-4,y+5 TO x-2,y+5 TO x-4,y TO x,y-5:FILL#ch,0
365 x=x-2:y=y+6:FILL#ch,1:LINE#ch,x,y+4 TO x-4,y+2 TO x-4,y-2 TO x+4,y-4 TO x+6,y-2 TO x+6,y TO x,y+4:FILL#ch,0:INK#ch,0:LINE#ch,x+4,y-2 TO x-4,y+2:INK#ch,7
366 END DEFine
```



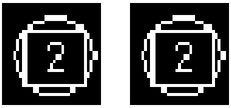
```
368 DEFine PROCedure MODR
369 ch=2:INK#ch,7:x=148:y=14
370 x=x:y=y:FILL#ch,1:LINE#ch,x,y-5 TO x+4,y-5 TO x+6,y TO x+4,y+5 TO x+2,y+5 TO x+4,y TO x,y-5:FILL#ch,0
371 x=x+4:y=y-2:FILL#ch,1:LINE#ch,x,y-5 TO x+4,y-5 TO x+6,y TO x+4,y+5 TO x+2,y+5 TO x+4,y TO x,y-5:FILL#ch,0
372 x=x+2:y=y+6:FILL#ch,1:LINE#ch,x,y+4 TO x+4,y+2 TO x+4,y-2 TO x-4,y-4 TO x-6,y-2 TO x-6,y TO x,y+4:FILL#ch,0:INK#ch,0:LINE#ch,x-4,y-2 TO x+4,y+2:INK#ch,7
373 END DEFine
```

RollDice

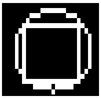
The Roll of the dice decides which side looses MODs. The Attacker looses to lower or even scores to that of the defending Planet. For a human Game Player if their Starship has less than four MODs the attack is automatically aborted.

(d**n**) holds the current Star System ID (**en**) controls the action:- a PAUSE for the human player to press any key and Roll the Dice. For the Computer's Move or the Simulator Mode this action is instant. (**Gp**) holds the selected side [Rebel Alliance or Federation] for updating a sides **MOD** results (see **Opponents Who's Who** below).

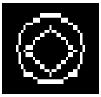
```
375 DEFine PROCedure RollDice
376 ch=2:REMark en=0 GPlay/en=1 SPlay dn=n : en=2 dn=cn CPlay
377 IF en=0 OR en=1:dn=n:ELSE dn=cn
378 IF en=0:PRINT#16,'To Roll Dice press any key':PAUSE
379 FOR a=1 TO RND(4 TO 6)
380   dx=  30:DiceA:PAUSE 1:DiceB
381   dx=136:DiceA:PAUSE 1:DiceB
382 END FOR a
383 dn1=RND(1 TO 8):dx=  30:DiceA:CURSOR#ch,dx,dy,-3,-4:PRINT#ch,dn1
384 dn2=RND(1 TO 8):dx=136:DiceA:CURSOR#ch,dx,dy,-3,-4:PRINT#ch,dn2
385 IF dn1>dn2:astro(dn,3)=astro(dn,3)-dm
386 IF dn1=dn2:side(Gp,3)=side(Gp,3)-dm
387 IF dn2>dn1:side(Gp,3)=side(Gp,3)-dm
388 IF astro(dn,3)<0:astro(dn,3)=0
389 END DEFine
```



```
391 DEFine PROCedure DiceA
392 dy=12:INK#ch,0:FILL#ch,1:CIRCLE#ch,dx,dy,4:FILL#ch,0
393 INK#ch,7:CIRCLE#ch,dx,dy,3.9
394 LINE#ch,dx-l,dy+l TO dx+l,dy+l TO dx+l,dy-l TO dx-l,dy-l TO dx-l,dy+l
395 END DEFine
```



```
397 DEFine PROCedure DiceB
398 dy=12:INK#ch,0:FILL#ch,1:CIRCLE#ch,dx,dy,4:FILL#ch,0
399 INK#ch,7:CIRCLE#ch,dx,dy,3.9
400 LINE#ch,dx,dy+l TO dx+l,dy TO dx,dy-l TO dx-l,dy TO dx,dy+l
401 END DEFine
```



Opponents Who's Who

It's a simple fact when trying to keep track of players turns it can get confusing. Giving the option of being either the Rebel Alliance or Federation doubles the problem.

The array **side(2,3)** holds the two players and their status of **Tec, Trade** & **MOD** values. Game Player **Gp1** represents the Human player and **Gp2** represents the Computer player.

Gp1= 1 Rebels or 2 Federation or visa versa Gp2= 2 Federation 1 Rebels

Gp1= 1 or 2	**side(Gp1,1/2/3)**	Human Player
Gp2= 2 or 1	**side(Gp2,1/2/3)**	Computer Player

The SAVE and re-LOAD function retains the S**ide** played and the **Skill** information as well as the GAME Time Session **GTS** and the current inventory of **Tec/Trade** & **MODs**.

```
403 DEFine PROCedure GNotes
404 INK 7:CURSOR 54, 2:PRINT 'Galaxy AD2370 Stratagems / Notes':INK 4
405 CURSOR 16,14:PRINT 'Dependant on number of StarShip MODs: attack'
406 CURSOR 16,24:PRINT 'opponents Planets to reduce their number to '
407 CURSOR 16,34:PRINT 'zero as quickly as possible to WIN the Game.'
408 CURSOR  8,48:PRINT 'Alternatively take as many Territories with low'
409 CURSOR  8,58:PRINT 'numbers of defence MODs to help ensure survival.'
410 CURSOR 12,72:PRINT 'Use ⇦ ⇧⇩ ⇨ Transfer Tec/Trade credits to increase'
411 CURSOR 32,82:PRINT 'the Population and (R)ecruit more MODs.'
412 CURSOR 32,96:PRINT 'Select Star System with ⇧⇩ and Spacebar' :INK 2
413 CURSOR 24,108:PRINT '(M)enu (F1)Quick Test (F2)Simulation Mode'
414 INK 7:OVER 1:CURSOR 24,108:PRINT ' M      F1           F2':OVER 0
415 PAUSE:CLS:FOR i=1 TO 120:POINT RND(10 TO 200),RND(5 TO 95)
416 END DEFine
```

GNotes
Activate by pressing (**H**) between actions taken in Main Game screen. Essentially this is a help page to provide a few pointers and reminder of the basic keys to use. Pressing (**M**) takes you to the main Menu where you can (**S**)ave to continue later.

(**F1**) **Game Sequences**
This checks various screen sequences, which takes you through the Hyperspace Jump **HJump** and brings you to the Star approach **StarView** and continues into Planetary Orbit **Eorbit**. You Starship then test fires its **Phaser** weapon and after a short delay leaves Orbit **Lorbit** and enters **GEnd** the **Final Battle** sequence between the Rebel Alliance and Federation.

(F2) Simulation Mode
This locks out the human Player and continues with **SPlay** the Simulator Mode playing against the **CPlay** Computer Moves. Sit back and relax – No frustration!

Thoughts on Game Strategies
A major break at the start of play is how you are dealt your opening hand. The 10 Territories Star/Planet Systems allocated to each side are randomly sourced with Tec/Trade credits & MODs (Machines Of Destruction). However, the total number of MODs is the same for both sides the difference always being made up as extra Starship MODs given to the one with the least Land based MODs.

This presents an initial Starship MODs advantage and if fortunate enough to begin with this you might simply target your opponent’s Territories and annex then as quickly as possible. Similarly you could go for as many Territories as you can, targeting all those with low MODs (i.e. with three or less) thereby increasing your opportunity of survival.

Alternatively, you could start by visiting each of your own Territories Transferring Tec /Trade credits to increase the Population and recruit more MODs. Thereby with increased MODs available, some can be transferred to your Starships, while leaving others to better defend your Territories against future attack.

```
418 DEFine PROCedure GAD2370
419 Test=0:Sclk=DATE:oldclk1=DATE+Tm1:oldclk2=DATE
420 MODInfo:n=RND(20 TO 50):sx=-9:sy=-9:PMap
421 REPeat lp
422   IF DATE>oldclk1                 :CPlay:oldclk1=DATE+Tm1
423   IF Test=1 AND DATE>oldclk2  :SPlay:oldclk2=DATE+Tm2
424   StarMove: GTime :k=CODE(INKEY$(5))
425   SELect ON k
426    =32 :PTInfo:GPlay:PMap:sx=-9:sy=-9             :REMark [SpaceBar] Player move
427    =72,104:GNotes                                 :REMark Help
428    =77,109:GTS=(DATE-Sclk+GTS):GMenu              :REMark (N)ew(L)oad(S)ave(E)xit
429    =208:IF Test=0 AND n>  1:n=n-1:PTInfo          :REMark move⇧ up
430    =216:IF Test=0 AND n<70:n=n+1:PTInfo           :REMark move⇩ down
431    =232:sn=n:GameTest                             :REMark F1 HJump/GEnd Quick Test
432    =236:IF Test=0:Test=1:ELSE Test=0              :REMark F2 SPlay Simulator
433   END  SELect
434 END REPeat lp
435 END DEFine
```

```
437 DEFine PROCedure GTime
438 Sclk$=DATE$(DATE-Sclk+GTS):INK#2,4:AT#2,21,74:PRINT#2,Sclk$(13 TO 20)
439 END DEFine
```

GAD2370 is the main Game loop

[CPlay] **Computer Player** called on time delay **-** dependant on Skill Level selected.
[Splay] **Simulator Player** called on time delay - runs through Game play actions.
[SpaceBar] **GPlay - Game Player** Attack/Transfer Tec/Trade/MODs.
[H] **GNotes** – Help Page
[M] **GMenu -** Games main menu [(**S**)ave / (**L**)oad Game to continue]
[⇧⇩] Selects Star System using Up/Down cursor keys
[F1] **GameTest** - quick test of screen sequences from Star selection to Game End
[F2] **SPlay -** activates Simulation mode [Game Player locked out]

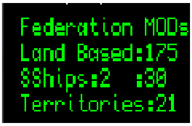
[**GTS**] **Game Time Session** - clocks the duration of the current Game and logs it as part of the (**S)**ave data. (**L)**oad adds the old **GTS** time to the new start of **GTS**.

```
441 DEFine PROCedure GameTest
442 HJump:StarView:Gp=Gp1:Eorbit:Phaser: IF Gp1=1:Ft=0:ELSE Rt=0
443 PAUSE 50:Lorbit:PMap:Lck=1:GEnd
444 END DEFine
```

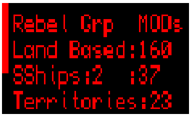
GameTest activated by (**F1**) checks basic graphics sequences, Hyperspace Jump, Star System approach and then move into Planetary orbit. The relevant sides Starship appears and fires its Phaser weapon at the planet below. After a short delay, it then leaves orbit and proceeds to Game End finishing with the high score League Table.

Game Information

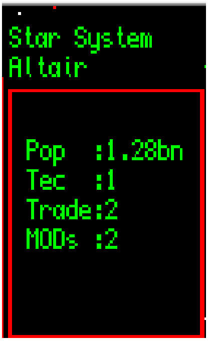
As Game Progresses, the Status of Play that take place is displayed in respective screen locations**. MODInfo** displays the current number of Land based and Starship MODs and Territories for each side. Selecting a Star **PTInfo** displays the Grid coordinates top left and Grid lines shown on the Galaxy Map. Information on Population, Tec/Trade credits and MODs are shown upper right. Upon reaching a Star, **SSInfo** displays Starships inventory of Tec/Trade & MODs plus Attack and/or Transfer action on the upper left.

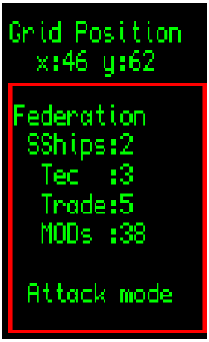

```
Federation MODs
Land Based:175
SShips:2  :30
Territories:21
```

PROCedure MODInfo
Displays the Game Status
MODs held on land bases
and in Starships together
with Annexed Territories.


```
Rebel Grp  MODs
Land Based:160
SShips:2  :37
Territories:23
```

```
446 DEFine PROCedure MODInfo
447 RMOD=0:Rt=0:RSSn=0:FMOD=0:Ft=0:FSSn=0
448 FOR i=1 TO 70:IF astro(i,5)=1:Rt=Rt+1:RMOD=RMOD+astro(i,3)
449 FOR i=1 TO 70:IF astro(i,5)=2:Ft =Ft+1:FMOD= FMOD+astro(i,3)
450 IF m=1:MODInit:m=0
451 IF Gp1=1:Rch=17:Rink=4:Fch=18:Fink=2:ELSE Rch=18:Rink=2:Fch=17:Fink=4
452 CLS#17:CLS#18:RSSn=1+INT(side(1,3)/25): FSSn=1+INT(side(2,3)/25)
453 INK#Rch,Rink:PRINT#Rch,'Rebel Grp  MODs'\'Land Based:';RMOD\'SShips:';RSSn;'
:';side(1,3)\'Territories:';Rt
454 INK#Fch,Fink:PRINT#Fch,'Federation MODs'\'Land Based:';FMOD\'SShips:';FSSn;'
:';side(2,3)\'Territories:';Ft
455 END DEFine

457 DEFine PROCedure MODInit
458 IF RMOD>FMOD:side(2,3)=20+RMOD-FMOD:side(1,3)=20
459 IF FMOD>RMOD:side(1,3)=20+FMOD-RMOD:side(2,3)=20
460 IF FMOD=RMOD:side(1,3)=20:side(2,3)=20
461 END DEFine

463 DEFine PROCedure PTInfo
464 ch=15:PMap     :side$='                    '
465 IF astro(n,5)=1:side$='Rebel Alliance'
466 IF astro(n,5)=2:side$='Federation     '
467 sx=astro(n,6):sy=astro(n,7)
468 PRINT#14,'Grid Position'\' x:';sx;' y:';sy
469 PRINT#15,'Star System'\astro$(n)
470 ch=12:CLS#ch:INK#ch,4:Pop$=astro(n,4)
471 PRINT#ch,side$
472 PRINT#ch,' Pop  :';Pop$(1 TO 4)&'bn'
473 PRINT#ch,' Tec  :';astro(n,1)
474 PRINT#ch,' Trade:';astro(n,2)
475 PRINT#ch,' MODs :';astro(n,3):PMap
476 END DEFine
```


```
Star System
Altair

 Pop  :1.28bn
 Tec  :1
 Trade:2
 MODs :2
```


```
Grid Position
  x:46 y:62
Federation
 SShips:2
  Tec  :3
  Trade:5
  MODs :38

 Attack mode
```

```
478 DEFine PROCedure SSInfo
479 MODInfo:PTInfo
480 ch=11:CLS#ch:INK#ch,4
481 IF Gp=1:Gp$='Rebel Alliance':ss=RSSn
482 IF Gp=2:Gp$='Federation     ':ss=FSSn
483 PRINT#ch,Gp$:PRINT#ch,' SShips:';ss
484 PRINT#ch,'  Tec  :';side(Gp,1)
485 PRINT#ch,'  Trade:';side(Gp,2)
486 PRINT#ch,'  MODs :';side(Gp,3)
487 AT#ch,6,0:PRINT#ch,Mes$
488 IF s=0 AND zr=1:ch=12:AT#ch,6,0:PRINT#ch,' (R)ecruit '
489 END DEFine
```

PMap shows Game Player chosen Star System on the Galaxy Map, **CMap** the opponent's location.

```
491 DEFine PROCedure PMap
492 ch=13:INK#ch,4:OVER#ch,-1
493 LINE#ch,sx,5 TO sx,95:LINE#ch,12,sy TO 306,sy
494 CIRCLE#ch,sx,sy,3:INK#ch,7:CIRCLE#ch,sx,sy,9:OVER#ch,0
495END DEFine
```

```
497 DEFine PROCedure CMap
498 ch=13:INK#ch,2:OVER#ch,-1:
499 LINE#ch,h2,5 TO h2,95:LINE#ch,12,v2 TO 306,v2
500 CIRCLE#cj,h2,v2,3:INK#ch,7:CIRCLE#ch,h2,v2,9:OVER#ch,0
501 END DEFine
```

StarMove generates new Stars and SCROLL/PAN's the Central View Screen.

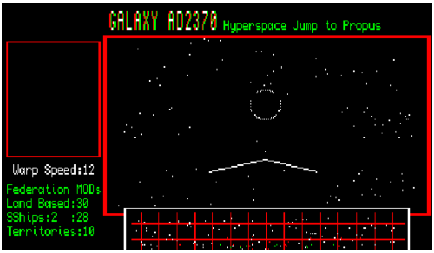
```
503 DEFine PROCedure StarMove
504 GTime:hoz=ws:ver=ws:INK#2,7:CURSOR#2,6,106:PRINT#2,'Warp Speed:'&ws&' '
505 IF ws>0:INK#1,7:FOR w=1 TO ws*2:POINT#1,RND(40 TO 160),RND(30 TO 70)
506 SCROLL#4,-ver:SCROLL#9,ver:PAN#6,-hoz*2:PAN#7,hoz*2
507 PAN#3,-hoz:SCROLL#3,-ver:PAN#5,hoz:SCROLL#5,-ver
508 PAN#8,-hoz:SCROLL#8,ver:PAN#10,hoz:SCROLL#10,ver
509 END DEFine
```

GPaly controls the action once a Star System has been selected. A Hyperspace Jump, followed by Star approach and on into planetary orbit. If already annexed to the opposition or independent Attack mode is activated. Once taken, then a Transfer of Tec/Trade and MODs can take place.

```
511 DEFine PROCedure GPlay
512 s=1:t=1:zr=1:en=0:Gp=Gp1:sn=n:HJump:StarView:Eorbit
513 REPeat act
514  IF Ft=0 OR Rt=0:Lorbit:PMap:CLS#11:CLS#12:CLS#14:CLS#15:GEnd
515  IF s=1 AND astro(n,5)=Gp:s=0:Mes$='⇧ ⇩Transfer⇦ ⇨ ':SSInfo
516  IF s=1 AND astro(n,3)=0:Victory
517  IF s=1 AND side(Gp,3)>dm+1
518     Mes$=' Attack Mode ':SSInfo:RollDice:Phaser:SSInfo:CLS#16
519  END IF
520  IF s=1 AND side(Gp,3)<dm+1
521     s=0:PRINT#16,'Not enough MODs' :PAUSE 25:EXIT act
522  END IF
523  IF s=0:ch=11:BLOCK#ch,10,30,0,20,0:AT#ch,t+1,0:PRINT#ch,'>'
524  GTime:k=CODE(INKEY$(5))
525  SELect ON k6
526    =192:TranPS
527    =200:TranSP
528    =208:t=t-1:IF t<1:t=3
529    =216:t=t+1:IF t>3:t=1
530    = 32:EXIT act
531    =82,114:IF s=0 AND zr=1:Recruit:zr=0
532  END SELect
533 END REPeat act
534 Lorbit:n=RND(1 TO 70):CLS#11:CLS#12:CLS#14:CLS#15
535 END DEFine
```

Space Travel - graphic sequence first a Hyperspace jump across the Galaxy then a slow down as you approach the sun of the selected star system.

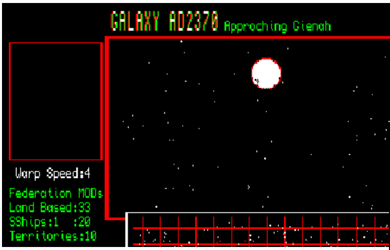
```
537 DEFine PROCedure HJump
538 ch=16:PRINT#ch,'Hyperspace Jump to ';astro$(sn)
539 w1=0:w2=0:wx=92:wy=60:ws=12
540 FOR a=9 TO 24 STEP 3
541   INK 241:CIRCLE wx,wy,a
542   INK   7:LINE 50+a,10+a TO wx,18+a TO 132-a,10+a:PAUSE 3
543   INK   0:CIRCLE wx,wy,a
544   INK   0:LINE 50+a,10+a TO wx,18+a TO 132-a,10+a:StarMove
545 END FOR a
546 FOR w1=1 TO 8
547  FOR w2=6 TO 22 STEP 4
548   INK 241:CIRCLE wx,wy,w2:PAUSE 1
549   INK   0:CIRCLE wx,wy,w2:StarMove
550  END FOR w2
551 END FOR w1
552 FOR a=1 TO 4
553   INK 241:CIRCLE wx,wy,a*6:PAUSE 1
554   INK   0:CIRCLE wx,wy,a*6:StarMove
555 END FOR a
556 END DEFine
```



```
GALAXY AD2370 Hyperspace Jump to Propus
Warp Speed:12
Federation MODs
Land Based:30
SShips:2  :28
Territories:10
```

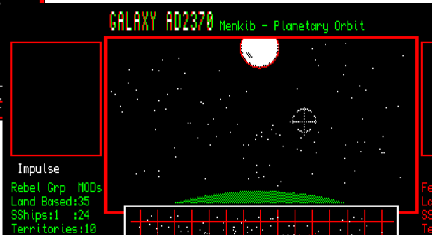
```
558 DEFine PROCedure StarView
559 ch=16:PRINT#ch,'Approching ';astro$(sn)
560  FOR i=2 TO 24 STEP 2
561    INK 0:FILL 1:CIRCLE 92,80,i*.6:FILL 0
562    ws=(26-i)/2:StarMove:x=92:y=80:m=i*.5:Star:PAUSE i/4
563  END FOR i
564 END DEFine
```

```
566 DEFine PROCedure Star
567  INK   2:FILL 1:CIRCLE x,y,m*1  :FILL 0
568  INK 241:FILL 1:CIRCLE x,y,m*.93:FILL 0
569  INK 241:FILL 1:CIRCLE x-m*.6,y-m*.4,m*.2,.3,PI/4:FILL 0
570 END DEFine
```



```
GALAXY AD2370 Approching Gienah
Warp Speed:4
Federation MODs
Land Based:33
SShips:1  :20
Territories:10
```

Star Approach



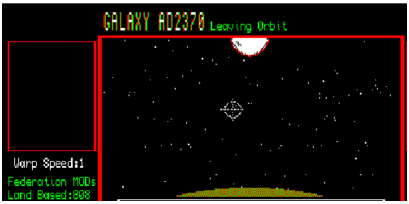
```
GALAXY AD2370 Menkib - Planetary Orbit
Impulse
Rebel Grp  MODs
Land Based:35
SShips:1  :24
Territories:10
```

Starship background signature as it comes into orbit above planet.

Eorbit – solar system approach to planetary orbit and presents Starship above the host planet.

```
574 DEFine PROCedure Eorbit
573 ch=16:CLS#ch:PRINT#ch,astro$(sn);' - Planetary Orbit':INK#1,7
574 CURSOR#2,6,106:PRINT#2,'Impulse      ':wx=92:CURSOR#1,0,60
575 FOR a=1 TO 18
576  POINT#1,RND(10 TO 180),RND(40 TO 80):SCROLL -1
577  IF astro(sn,9)=1:wx=18+a*3
578  IF astro(sn,9)=3:wx=182-(a*3)
579  INK astro(sn,8):wy=24-a*3
580  FILL 1:CIRCLE wx,wy,a*2:FILL 0:INK 0:PAUSE a/3
581  FILL 1:CIRCLE wx,wy,a*2:FILL 0:INK 7
582  POINT RND(40 TO 160),RND(30 TO 40)
583 END FOR a
584 FOR b=1 TO 14
585   INK 7:POINT RND(10 TO 180),RND(10 TO 20)
586   ink1=7:ink2=241:IF Gp=1:x=92+b*3:y=50:ELSE x=92-b*3:y=55
587   INK astro(sn,8):FILL 1:CIRCLE 92,-200+b,198:FILL 0
588   INK 241:CIRCLE x,y,b-2:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b:PAUSE b/2
589   INK   0:CIRCLE x,y,b-2:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b
590 END FOR b
591 ink1=7:ink2=241:IF Gp=1:x=140:y=50:SS02:ELSE x=36:y=50:SS03
592 CURSOR#2,6,106:PRINT#2,'Orbit       '
593 END DEFine
```

Lorbit - sequence of leaving planetary orbit.



```
595 DEFine PROCedure Lorbit
596 ch=16:CLS#ch:PRINT#ch,'Leaving Orbit':PAUSE 20
597 ink1=0:ink2=0:IF Gp=1:x=140:y=50:SS02:ELSE x=36:y=50:SS03
598 CURSOR 0,60:INK 7:CURSOR#2,6,106:PRINT#2,'Warp Speed:1'
599 FOR b=15 TO 1 STEP -1
600   IF Gp=1:x=140-42+(b*3):y=50:ELSE x=60+42-(b*3):y=55
601   INK 241:CIRCLE x,y,b-4:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b:PAUSE 3
602   INK   0:CIRCLE x,y,b-4:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b
603   INK 7:POINT RND(20 TO 180),RND(20 TO 70):SCROLL -1,1:PAN#6,-1
604   INK 7:POINT RND(20 TO 180),RND(20 TO 70):SCROLL  1,2:PAN#7, 1
605 END FOR b
606 CLS#16
607 END DEFine
```

Phaser- shoots laser weapon at different locations on the Planet's surface.

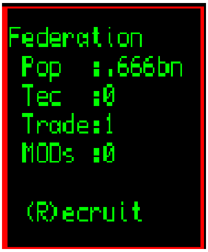
```
609 DEFine PROCedure Phaser
610 IF Gp=1:px=132:py=44:ELSE px=70:py=50
611 wx=RND(80 TO 120):wy=12
612 FOR a=1 TO 6
613  INK 2:FILL 1:LINE px,py TO wx-1,wy TO wx+1,wy TO px,py:FILL 0:PAUSE 3
614  INK 0:FILL 1:LINE px,py TO wx-1,wy TO wx+1,wy TO px,py:FILL 0:PAUSE 2
615 END FOR a
616 END DEFine
```

Victory - awards the Victor with additional Tec/Trade credits and MODs

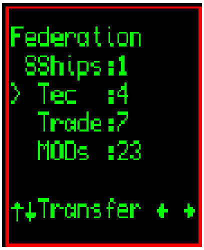
```
618 DEFine PROCedure Victory
619 CLS#16:PRINT#16,'Tec/Trade & Recruit MODs'
620 side(Gp,1)=side(Gp,1)+1+INT(astro(n,1)/4)
621 IF side(Gp,1)>24:side(Gp,1)=24
622 side(Gp,2)=side(Gp,2)+2+INT(astro(n,2)/5)
623 IF side(Gp,2)>40:side(Gp,2)=40
624 side(Gp,3)=side(Gp,3)+3+INT(astro(n,4)/2)
625 IF side(Gp,3)>100:side(Gp,3)=100
626 astro(n,5)=Gp:s=0:Mes$='¾¿Transfer ¼ ½'
627  MODInfo:SSInfo
628 END DEFine
```

Transfer Mode - move Tec/Trade credits to Planet to help (**R**)ecruit more MODs then move back Tec//Trade and MODs to Starship building up forces for future attacks.

```
630 DEFine PROCedure TranPS
631 REMark Transfer Tec/trade MOD's from Planet > Star Ship
632 IF t=1 AND astro(n,1)>1 AND side(Gp,1)<24
633    astro(n,1)=astro(n,1)-1:side(Gp,1)=side(Gp,1)+1
634 END IF
635 IF t=2 AND astro(n,2)>1 AND side(Gp,2)<40
636    astro(n,2)=astro(n,2)-1:side(Gp,2)=side(Gp,2)+1
637 END IF
638 IF t=3 AND astro(n,3)>1 AND side(Gp,3)<100
639    astro(n,3)=astro(n,3)-1:side(Gp,3)=side(Gp,3)+1
640 END IF
641 SSInfo
642 END DEFine
```



```
Federation
 Pop  :.666bn
 Tec  :0
 Trade:1
 MODs :0

 (R)ecruit
```



```
Federation
 SShips:1
> Tec  :4
 Trade:7
 MODs :23

↑↓Transfer ← →
```

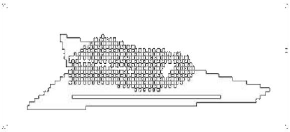
```
644 DEFine PROCedure TranSP
645 REMark Transfer Tec/trade MOD's from Star Ship > Planet
646 IF t=1 AND side(Gp,1)>1 AND astro(n,1)<12
647    side(Gp,1)=side(Gp,1)-1:astro(n,1)=astro(n,1)+1
648 END IF
649 IF t=2 AND side(Gp,2)>1 AND astro(n,2)<20
650    side(Gp,2)=side(Gp,2)-1:astro(n,2)=astro(n,2)+1
651 END IF
652 IF t=3 AND side(Gp1,3)>1 AND astro(n,3)<20
653    side(Gp,3)=side(Gp,3)-1:astro(n,3)=astro(n,3)+1
654 END IF
655 SSInfo
656 END DEFine
```

(R)ecruit- recalculates Population and MODs following Tec/Trade Transfers.

```
658 DEFine PROCedure Recruit
659 REMark Trade Tec/Trade credits increase Population & MODs
660 astro(n,4)=astro(n,1)*.5+astro(n,2)*.2
661 astro(n,3)=INT(astro(n,4)/.5)
662 ch=12:AT#ch,6,0:PRINT#ch,' Recruiting':PAUSE 20:AT#ch,6,0:CLS#ch,4
663 ch=16:CLS#ch:PRINT#ch,'To leave Orbit press':BLOCK#ch,16,4,130,4,4
664 MODInfo:PTInfo
665 END DEFine
```

PART 3

Game End - Graphic sequence showing **Final Battle** between Rebel Alliance and Federation forces.



```
667 DEFine PROCedure GEnd
668 LMap:DeathStar:CSIZE 2,1
669 IF Ft=0
670    dx=56:dy=76:wx=156:wy=28:DSDefeat:PAUSE 10:INK 7:OVER 1
671    FOR i=1 TO 2:CURSOR 64+i,44:PRINT 'Rebel Alliance'
672    OVER 0:ink1=7:x=34:y=24:SS02
673 END IF
674 IF Rt=0
675    dx=56:dy=76:wx=156:wy=28:DSAttack:PAUSE 10:INK 7:OVER 1
676    FOR i=1 TO 2:CURSOR 64+i,44:PRINT '  Federation  '
677    OVER 0:ink1=7:ink2=241:x=20:y=24:SS03
678 END IF
679 CSIZE 2,0:INK 7:OVER 1
680 FOR i=1 TO 2:CURSOR 70+i,66:PRINT#ch,'Wins the Game'
681 OVER 0:CSIZE 0,0:INK 4
682 PAUSE 100:LName:PAUSE:GMenu
683 END DEFine
```

```
685 DEFine PROCedure DeathStar
686 ws=1:dx=20:dy=96:wx=92:wy=5
687 FOR a=1 TO 18
688   INK astro(n,8):wx=92+a*4:wy=-46+a*4
689   FILL 1:CIRCLE wx,wy,50-a*2:FILL#1,0
690   INK 241:FILL 1:CIRCLE dx+a*2,dy-a,a/3:FILL 0:PAUSE 2
691   INK   0:FILL 1:CIRCLE dx+a*2,dy-a,a/3:FILL 0
692   FILL 1:CIRCLE wx,wy,50-a*2:FILL 0
693 END FOR a
694 dx=56:dy=76:wx=156:wy=28:CURSOR#2,6,106:PRINT#2,'              '
695 INK astro(n,8):FILL 1:CIRCLE wx,wy,14:FILL 0
696 ink1=7:ink2=241:SS01
697 END DEFine
```

```
699 DEFine PROCedure DSAttack
700 FOR a=1 TO 6
701   INK 2:FILL 1
702   LINE dx+2,dy+4 TO wx-10,wy+6 TO wx-12,wy+4 TO dx+2,dy+4
703   FILL 0:PAUSE 3:INK 0:FILL 1
704   LINE dx+2,dy+4 TO wx-10,wy+6 TO wx-12,wy+4 TO dx+2,dy+4
705   FILL 0:PAUSE 2
706 END FOR a
707 INK 7:FILL 1:CIRCLE dx+2,dy+4,4:FILL 0
708 FOR b=1 TO 12:INK 0 :CIRCLE 140+b,32,b:PAUSE 2
709 FOR c=1 TO 12:FILL 1:CIRCLE 150+c*2,28+RND(-4 TO 8),RND(2 TO 5):FILL 0
710 END DEFine
```

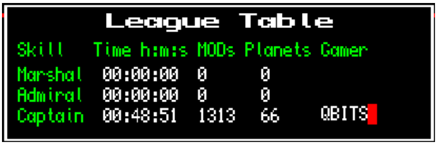
```
712 DEFine PROCedure DSDefeat
713 FOR i=1 TO 18
714    INK 2:x1=RND(-6 TO 12):y1=RND(-6 TO 12):LINE dx,dy TO dx+x1,dy+y1
715    INK 0:FILL 1:CIRCLE dx+RND(-2 TO 6),dy+RND(-2 TO 6),1:FILL 0:PAUSE 2
716 END FOR i
717 END DEFine
```

Galaxy Map - showing the Star Systems annexed by winning side Rebels or Federation.

```
719 DEFine PROCedure LMap
720 LOCal n,nx,ny
721 FOR n=1 TO 70
722    nx=astro(n,6):ny=astro(n,7):INK#13,7
723    IF astro(n,5)=Gp1:CIRCLE#13,nx,ny,3:INK#13,4:CIRCLE#13,nx,ny,6
724    IF astro(n,5)=Gp2:CIRCLE#13,nx,ny,3:INK#13,2:CIRCLE#13,nx,ny,6
725 END FOR n
726 END DEFine
```

League Table - showing the Leading Players Skill Level achievements.

```
728 DEFine PROCedure LScore
729 ch=13:CLS#ch:OVER#ch,1:CSIZE#ch,2,0:INK#ch,7
730 FOR i=1 TO 2:CURSOR#ch,60+i,2:PRINT#ch,'League Table'
731 OVER#ch,0:CSIZE#ch,0,0:INK#ch,4
732 CURSOR#ch,6,16:PRINT#ch,'Skill   Time h:m:s MODs Planets Gamer'
733 CURSOR#ch,6,28:PRINT#ch,'Marshal'\' Admiral'\' Captain':INK#ch,7
734 FOR a=1 TO 3
735   HST$=DATE$(score(a,1)):HS2=score(a,2):HS3=score(a,3)
736   CURSOR#ch,60,18+a*10:PRINT#ch,HST$(13 TO 20);'  ';HS2
737   CURSOR#ch,160,18+a*10:PRINT#ch,HS3,name$(a)
738 END FOR a
739 END DEFine
```



```
             League Table
Skill    Time h:m:s MODs Planets Gamer
Marshal   00:00:00  0       0
Admiral   00:00:00  0       0
Captain   00:48:51  1313    66      QBITS
```

(**H**)ighscore

```
741 DEFine PROCedure LName
742 GTS=DATE-Sclk+GTS:Sclk$=DATE$(GTS)
743 IF Lck=1:Lck=0:LScore:RETurn
744 IF score(GSk,1)=0:GO TO 747
745 IF score(GSk,1)<GTS:LScore:RETurn
746 IF Gp=1:HS2=RMOD+side(1,3):HS3=Rt: ELSE HS2=FMOD+side(2,3):HS3=Ft
747 AT#2,21,74:PRINT#2,'        '                          Note: Blanks current GTS display
748 score(GSk,1)=GTS:score(GSk,2)=HS2:score(GSk,3)=HS3:LScore
749 ch=20:OPEN#ch,con_60x10a0x0_10:WINDOW#ch,60,10,320,166+GSk*10
750 PAPER#ch,0:CLS#ch:INK#ch,7:INPUT#ch,Ent$:name$(GSk)=Ent$:CLOSE#ch:LSave
751 END DEFine
```

```
753 DEFine PROCedure LSave
754 DELETE HSFile$ : OPEN_NEW#99,LTFile$
755 FOR a=1 TO 4:PRINT#99,name$(a)\score(a,1)\score(a,2)\score(a,3)
756 CLOSE#99
757 END DEFine
```

```
759 DEFine PROCedure LLoad
760 OPEN_IN#99,LTFile$
761 FOR a=1 TO 4:INPUT#99,name$(a)\score(a,1)\score(a,2)\score(a,3)
762 CLOSE#99
763 END DEFine
```

QBITS Galaxy AD2370 Computer Player

CPlay follows the actions derived for the Game namely selecting a Star System, if not one of its own the order of play being to Attack. Once **Victory** has been secured or if Territory already annexed then continues with Transfer of **Tec** & **Trade** credits. The Population levels raised with **Recruit** and generation of more **MOD**s to be either left for defence, or uploaded to Starships for future Attacks on other Territories.

```
    765 DEFine PROCedure CPlay
    766 INK#2,7:CURSOR#2,408,106:PRINT#2,'Opponents Move'
    767 s=1:zr=1:en=2:Gp=Gp2:MODInfo
    768 IF Gp=1:side$='Rebels    ':ELSE side$='Federation'
    779 IF side(Gp,3)>=20:cn=RND(1 TO 70)
    770 IF side(Gp,3)<20
    771   FOR i=1 TO 10
    772     cn=RND(1 TO 70):IF astro(cn,3)>10 AND astro(cn,5)=Gp:EXIT i
    773   END FOR i
    774 END IF
    775 IF Gp=1 AND Rt>30:FOR i=1 TO 5:cn=RND(1 TO 70):IF astro(cn,5)=2:EXIT i
    776 IF Gp=2 AND Ft>30:FOR i=1 TO 5:cn=RND(1 TO 70):IF astro(cn,5)=1:EXIT i
    777 sn=cn: h2=astro(cn,6):v2=astro(cn,7):ws=1:CMap:StarView:Eorbit
A   778 IF astro(cn,5)=Gp
    779   PRINT#16,astro$(cn)&' Taken':CLS#16,4
    780   IF astro(cn,1)< 9:astro(cn,1)=astro(cn,1)+3
    781   IF astro(cn,2)<15:astro(cn,2)=astro(cn,2)+5
    782   astro(cn,4)=astro(cn,1)*.5+astro(cn,2)*.2
    783   astro(cn,3)=INT(astro(cn,4)*2)
    784   IF side(Gp,3)<20 AND astro(cn,3)>6
    785     side(Gp,3)=side(Gp,3)+3:astro(cn,3)=astro(cn,3)-3
    786   END IF
    787   IF side(Gp,3)>50 AND astro(cn,3)<10
    788     astro(cn,3)=astro(cn,3)+5:side(Gp,3)=side(Gp,3)-5
    789   END IF
    790 ELSE
    791  PRINT#16,side$&' Attack '&astro$(cn):CLS#16,4
B   792  IF astro(cn,3)>side(Gp,3)*3 OR side(Gp,3)<10:GO TO 802 [C]
    793  RollDice :Phaser: PAUSE 20
    794  IF astro(cn,3)=0
    795    ch=16:INK#ch,2:PRINT#ch,astro$(cn);' Falls'
    796    IF side(Gp,3)<=95:side(Gp,3)=side(Gp,3)+5
    797    IF side(Gp,3)<=30:side(Gp,3)=side(Gp,3)+5
    798    astro(cn,5)=Gp:GO TO 778 [A]
    799  END IF
    800  MODInfo:GO TO 792 [B]
    801 END IF
C   802 PAUSE 50
    803 MODInfo:ws=1:en=0:CLS#16:INK#16,4
    804 Lorbit: CMap:CURSOR#2,408,106:PRINT#2,'              '
    805 IF Ft=0 OR Rt=0:GEnd
    806 END DEFine
```

Note: Main disadvantage of the above code is the use of Random Numbers to correlate most actions. However, for simplicity this works reasonable well.

QBITS Galaxy AD2370 Simulation Player

Press [**F2]** while in Game Screen to activate the Simulation mode **SPlay** locking out the Human Player and plays against **CPlay**. Depending on what Skill level chosen the delay time will influence the interaction of **CPlay** and the outcome.

```
    808 DEFine PROCedure SPlay
    809 INK#2,7:CURSOR#2,408,106:PRINT#2,'Simulator Mode'
    810 s=1:zr=1:en=1:Gp=Gp1:MODInfo
    811 IF Gp=1:side$='Rebel     ':ELSE side$='Federation'
    812 IF side(Gp,3)>=20:n=RND(1 TO 70)
    813 IF side(Gp,3)<20
    814   FOR i=1 TO 10
    815     n=RND(1 TO 70):IF astro(n,3)>10 AND astro(n,5)=Gp:EXIT i
    816   END FOR i
    817 END IF
    818 IF Gp=1 AND Ft>30:FOR i=1 TO 5:n=RND(1 TO 70):IF astro(n,5)=2:EXIT i
    819 IF Gp=2 AND Rt>30:FOR i=1 TO 5:n=RND(1 TO 70):IF astro(n,5)=1:EXIT i
    820 sn=n:PTInfo:HJump:StarView:Eorbit
A   821 IF astro(n,5)=Gp
    822   Mes$='⇧⇩ Transfer ⇦ ⇨':SSInfo
    823   IF astro(n,1)<9:astro(n,1)=astro(n,1)+3
    824   IF astro(n,2)<15:astro(n,2)=astro(n,2)+5
    825   astro(n,4)=astro(n,1)*.5+astro(n,2)*.2
    826   astro(n,3)=INT(astro(n,4)*2)
    827   IF side(Gp,3)<20 AND astro(n,3)>6
    828     side(Gp,3)=side(Gp,3)+astro(n,3)-3
    829   END IF
    830   IF side(Gp,3)>50 AND astro(n,3)<10
    831     astro(n,3)=astro(n,3)+5:side(Gp,3)=side(Gp,3)-5
    832   END IF
    833 ELSE
    834   Mes$=' Attack mode '::SSInfo
B   835   IF astro(n,3)>side(Gp,3)*3 OR side(Gp,3)<10:GO TO 845 [C]
    836   RollDice:Phaser:PAUSE 20
    837   IF astro(n,3)=0
    838     ch=16:INK#ch,2:PRINT#ch,astro$(n);' Falls'
    839     IF side(Gp,3)<=95:side(Gp,3)=side(Gp,3)+5
    840     IF side(Gp,3)<=30:side(Gp,3)=side(Gp,3)+5
    841     astro(n,5)=GpGO TO 821 [A]
    842   END IF
    843   SSInfo:GO TO 835 [B]
    844 END IF
C   845 SSInfo:PAUSE 50:Lorbit:PMap:sx=-9:sy=-9
    846 ws=1:en=0:CLS#11:CLS#12:CLS#14:CLS#15:CLS#16:INK#16,4
    847 CURSOR#2,408,106 :PRINT#2,'             '
    848 IF Ft=0 OR Rt=0:GEnd
    849 END DEFine
```

Note: The **GO TO** command is a very powerful tool, yet used globally in a program it can easily lead to catastrophic effects. However, used as a simple loop command between the lines within a PROCedure it can be a very useful.

QBITS Galaxy AD2370 Test Prog

To run Galaxy AD2370 requires file **QBAD2370DLT** to be loaded. This Test Program will reset the file entries to zero and clear names in its default mode. It can also be used to test other aspects of the League Table workings. Setting new entries, updating when lower GTS times have been achieved, ignoring higher GTS times than previous one.

```
900 REMark QBAD2370DLTP Galaxy AD2370 Test Prog
901 :
902 REMark Run Galaxy - check Intro page & Menu Options working Ok
903 REMark From Menu select (N) then CTRL-SPACEBAR to halt program: RUN TP
904 REMark then Run Galaxy again select (L) check League Table entries
905 :
906 DEFine  PROCedure TP
907 REMark Update League Table - use lines 908 or 909 with 910 & 911
908 REMark Ft=0:Gp=1:RMOD=789 :side(1,3)=87:Rt=57
909 REMark Rt=0:Gp=2:FMOD=1234:side(2,3)=79:Ft=66
910 REMark GTS=2885 :REMark 48min=2885 127min=7620 252min=15060 321min=19260
911 REMark GSk=3:GEnd
912 :
913 REMark resets flp1_QBAD2370DLT League Table
914 name$(1)='        ':score(1,1)=0:score(1,2)=0:score(1,3)=0
915 name$(2)='        ':score(2,1)=0:score(2,2)=0:score(2,3)=0
916 name$(3)='        ':score(3,1)=0:score(3,2)=0:score(3,3)=0
917 LSave
918 :
919 REMark lck=0:LName:PAUSE:LScore:PAUSE
920 END DEFine
```

QL Emulators – Sinclair QL Sites

www.dilwyn.me.uk/emu/index.html www.rwapsoftware.co.uk/emulators.html

QL2K emulator

The **QLAY & QL2K emulators** use an application tool to create a QDOS directory file and append or delete files in it. See accompanying Documentation from download site.

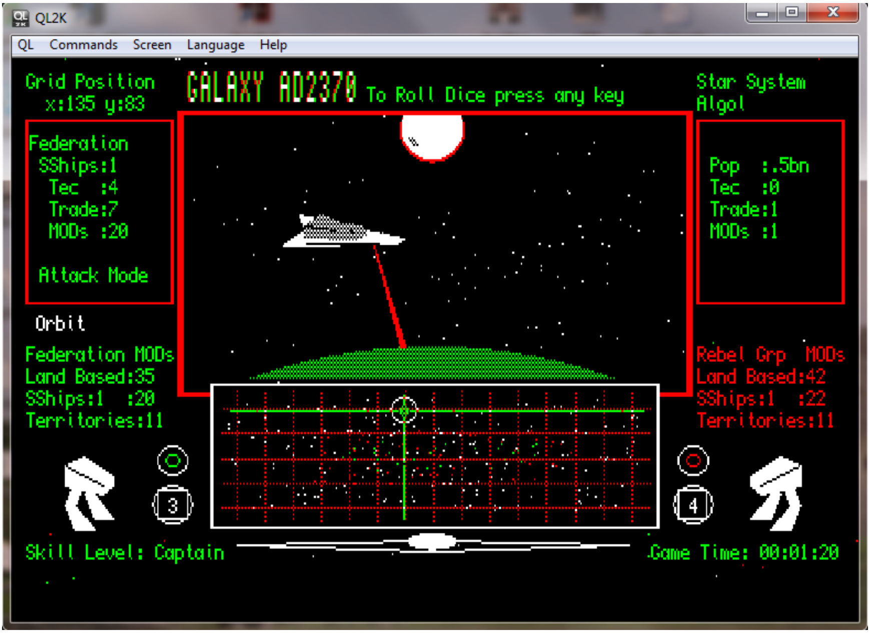
SMSQmulator

Running the emulator use **config** to set screen size to 512x256 and screen colours to QL colours then set devices for you file access.

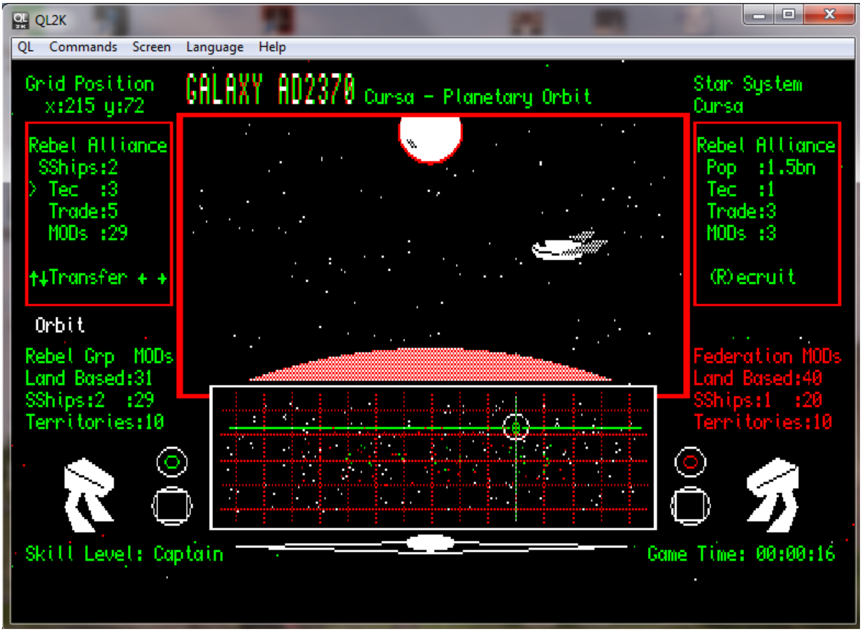
Having downloaded a file copy of **QBAD2370v3DL** and **QBAD2370v4DLT** into a recognised QL device (i.e.**flp1_**) use LRUN command to load and activate:-

LRUN flp1_QBAD2370v3DL

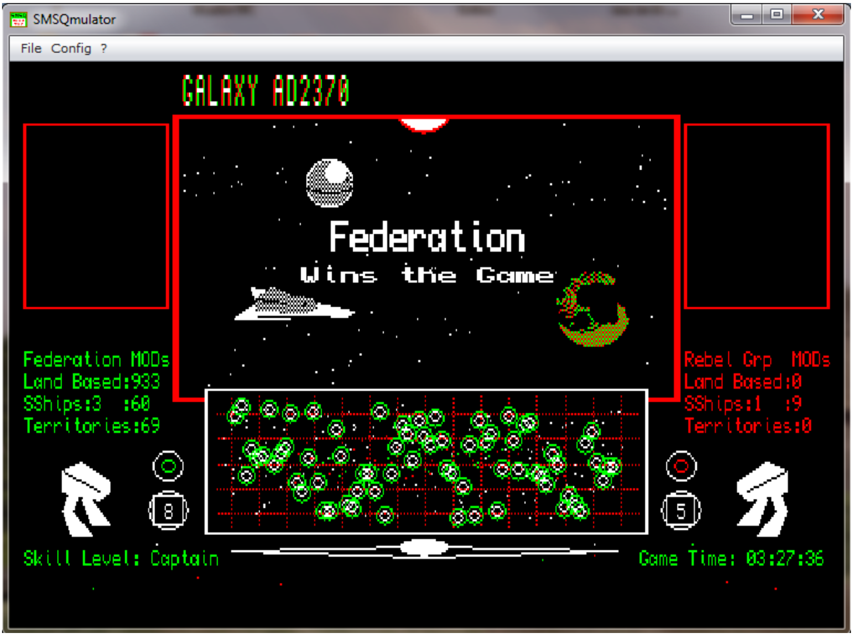
Follow the instructions on the **QBITS GALAXY AD2370** Intro screen and all being well you will soon be battling across the Galaxy...



```
Grid Position      GALAXY AD2370 To Roll Dice press any key      Star System
  x:135 y:83                                                     Algol

Federation                                                         Pop  :.5bn
 SShips:1                                                          Tec  :0
  Tec  :4                                                          Trade:1
  Trade:7                                                          MODs :1
  MODs :20

 Attack Mode

 Orbit
Federation MODs                                                  Rebel Grp  MODs
Land Based:35                                                    Land Based:42
SShips:1   :20                                                   SShips:1   :22
Territories:11                                                   Territories:11

Skill Level: Captain                                             Game Time: 00:01:20
```



```
Grid Position      GALAXY AD2370 Cursa - Planetary Orbit         Star System
  x:215 y:72                                                     Cursa

Rebel Alliance                                                   Rebel Alliance
 SShips:2                                                         Pop  :1.5bn
> Tec  :3                                                         Tec  :1
  Trade:5                                                         Trade:3
  MODs :29                                                        MODs :3

↑↓Transfer ← →                                                     (R)ecruit

 Orbit
Rebel Grp  MODs                                                  Federation MODs
Land Based:31                                                    Land Based:40
SShips:2   :29                                                   SShips:1   :20
Territories:10                                                   Territories:10

Skill Level: Captain                                             Game Time: 00:00:16
```



GALAXY AD2370

Federation

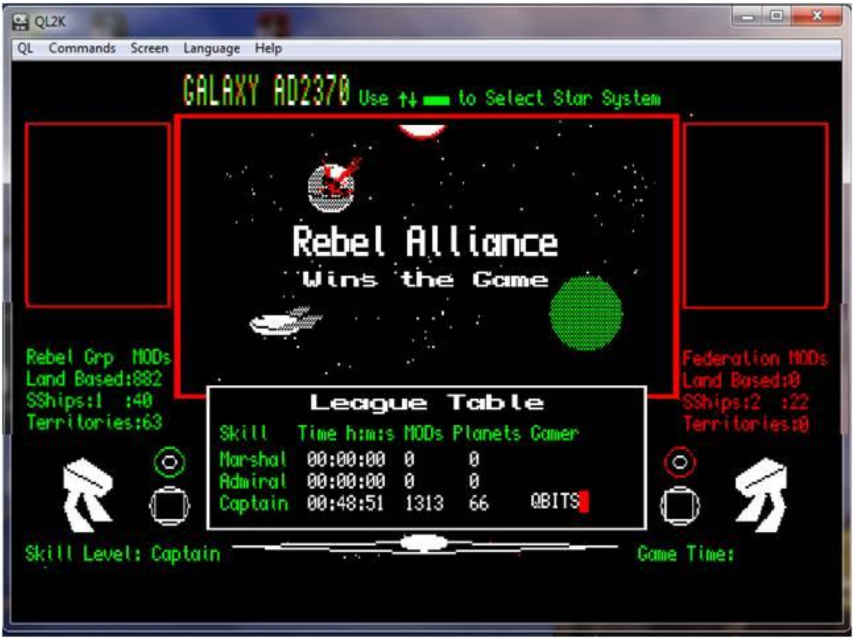
Wins the Game

Federation MODs
Land Based:933
SShips:3 :60
Territories:69

Rebel Grp MODs
Land Based:0
SShips:1 :9
Territories:0

Skill Level: Captain

Game Time: 03:27:36



GALAXY AD2370 Use ↑↓ ▬ to Select Star System

Rebel Alliance

Wins the Game

Rebel Grp MODs
Land Based:882
SShips:1 :40
Territories:63

Federation MODs
Land Based:0
SShips:2 :22
Territories:0

League Table

Skill	Time h:m:s	MODs	Planets	Gamer
Marshal	00:00:00	0	0	
Admiral	00:00:00	0	0	
Captain	00:48:51	1313	66	QBITS

Skill Level: Captain

Game Time: