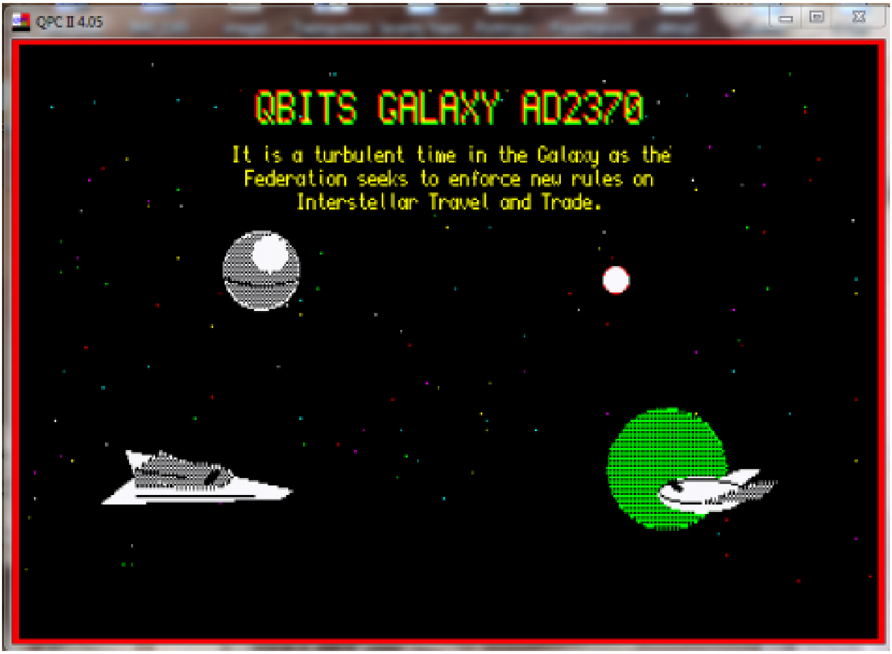
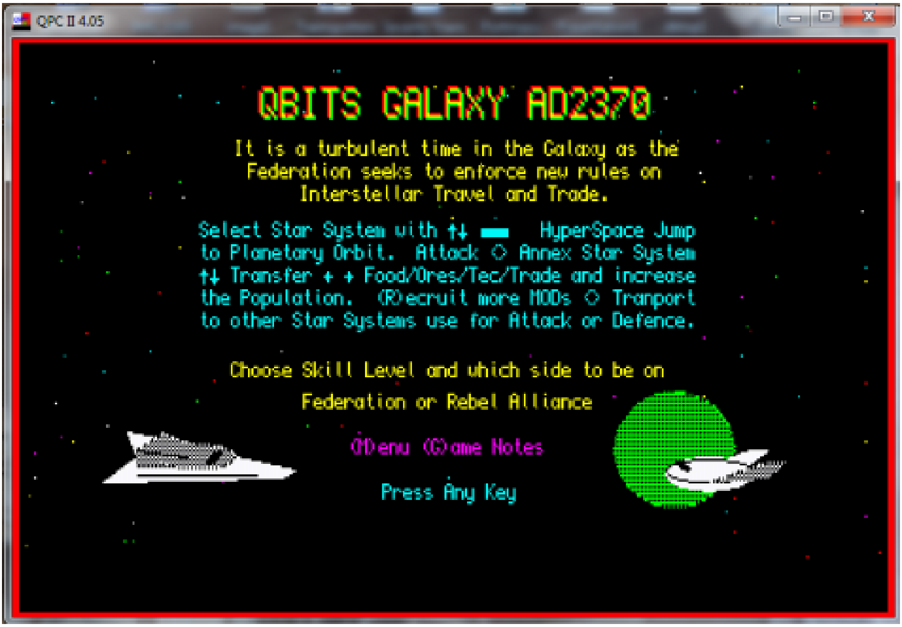
Sinclair QL Retro Gaming



QBITS GALAXY AD2370

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

Sinclair QL Retro Gaming



QPC II 4.05

QBITS GALAXY AD2370

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

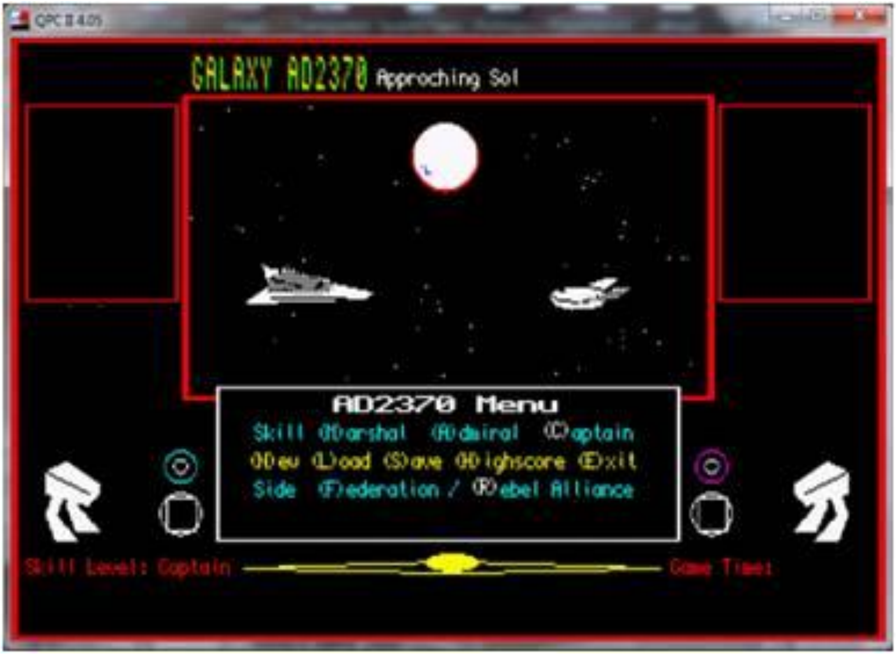
Select Star System with ↑↓ ▬ HyperSpace Jump
to Planetary Orbit. Attack ◇ Annex Star System
↑↓ Transfer ← → Food/Ores/Tec/Trade and increase
the Population. (R)ecruit more MODs ◇ Tranport
to other Star Systems use for Attack or Defence.

Choose Skill Level and which side to be on

Federation or Rebel Alliance

(M)enu (G)ame Notes

Press Any Key



QPC II 4.05

GALAXY AD2370 Approching Sol

AD2370 Menu

Skill (M)arshal (A)dmiral (C)aptain

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

Side (F)ederation / (R)ebel Alliance

Skill Level: Captain

Game Time:

QBITS Introduction

Early nineteen-eighties I bought a ZX81 with its somewhat quirky keyboard arrangement for entering key words. Depending on context each had combined functions via the SHIFT and FUNCTION keys. For example the P key gave the letter (**P**), the (**"**) quotes character and the BASIC commands **PRINT** and **TAB**. This became my introduction and first experience of computer BASIC.

In nineteen-eighty-four I gained my NHC in Electrical and Electronic Engineering. The later part of the course introduced me to the Forth Computer language. Memory was expensive back then so I was chuffed to say I managed to write a full washing machine program in just 29 lines. This was run on a microcomputer, possibly the Raspberry Pi of its day. Wired up to an electronics breadboard with discrete components, NAND and NOR gate microchips plus an Op amp or two, the on/off switched outputs were LED's. In the last few months of my time at college, they opened a computer lab laid out with BBC Micros. That was when I gained a more informative introduction to programming in BASIC.

Nineteen-eight-five I was employed on a project to network a campus sized site with 7,000 employees downsizing from Mainframes to Personal Computers. In that same year I bought my first QL at work we had just acquired our first IBM Personal Computer, but with only a monochrome screen. The QL was much less expensive had a colour output plus it was supplied with the Psion Software package. The unfortunate drawback was of course those microdrives.

QBITS Galaxy Space Adventure

I've never been much of a computer gamester, my interest lay more in trying to understand the coding producing the graphics and screen layouts. By exploring other people's programs and SuperBASIC commands I was soon writing and trying out my own code. Keen to develop a space adventure game, **QBITS Galaxy** began in the later part of the nineteen eighties. It was a time when I had perhaps an overly enthusiastic interest in Science Fiction, and I played a lot of board games such as RISK with family and friends.

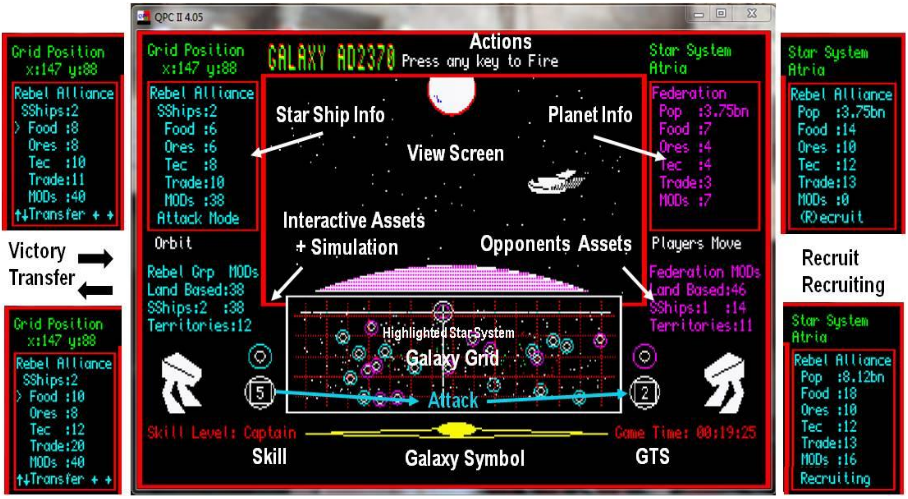
June 1987 Sinclair QL World Magazine: The Progs published **Stellaris** by David Cormona. I now realise this must have had a big impact on what I was trying to achieve and a possible reason why I didn't pursue my Game at the time. Today my thoughts are on whether there is any personal connection with that nineteen-eighties QL Game and the release of **Stellaris Galaxy** by Paradox Development Studio in 2016.

QBITS Galaxy AD2370v5

The Game having reached a level of maturity, it's time now to explore the thoughts and reasoning behind the code and its workings. However, for those lazy individuals like me you can still sit back and enjoy your coffee. Key **F1** runs the Game in **Simulation Mode**. **F2** runs through the view screen **Graphic Displays**. If you take up the Games challenge then **F3** to **F5** provides further checks and cheats to boost your survival. Have fun!

QBITS Galaxy AD2370v5 - Basic Concept

This was to be by definition a Game of strategy, where the **Antagonist** has to figure out the quickest or most sustainable route to defeat their **Opponent** and gain ultimate success. In this type of contest, a player uses **assets** to raise their **status** and bring about control of the game, taking advantage to limit and restrict their **Opponent's** opportunities or potential.



```
Grid Position
 x:147 y:88
Rebel Alliance
 SShips:2
> Food :8
  Ores :8
  Tec  :10
  Trade:11
  MODs :40
↑↓Transfer + +
```

```
QPC II 4.05

Grid Position      GALAXY AD2370 Press any key to Fire      Star System
 x:147 y:88                                                  Atria
Rebel Alliance                                               Federation
 SShips:2                                                     Pop  :3.75bn
  Food :6                                                     Food :7
  Ores :6                                                     Ores :4
  Tec  :8                                                     Tec  :4
  Trade:10                                                    Trade:3
  MODs :38                                                    MODs :7
 Attack Mode
Orbit                                                        Players Move
Rebel Grp  MODs                                              Federation MODs
Land Based:38                                                Land Based:46
SShips:2   :38                                               SShips:1   :14
Territories:12                                               Territories:11

        5                                                        2

Skill Level: Captain                                   Game Time: 00:19:25
```

```
Star System
Atria
Rebel Alliance
 Pop  :3.75bn
 Food :14
 Ores :10
 Tec  :12
 Trade:13
 MODs :0
 (R)ecruit
```

```
Grid Position
 x:147 y:88
Rebel Alliance
 SShips:2
> Food :10
  Ores :8
  Tec  :12
  Trade:20
  MODs :40
↑↓Transfer + +
```

```
Star System
Atria
Rebel Alliance
 Pop  :8.12bn
 Food :18
 Ores :10
 Tec  :12
 Trade:13
 MODs :16
 Recruiting
```

QBITS Galaxy AD2370 - Game Screen

The above screen and additional side panel's shows the layout with information on the different phases of action involved. The game begins by selecting a Star System the location highlighted by the cross-wires on the Galaxy Grid. Moving graphics depicts a Space Hyper-jump to Orbit above the Star Systems primary Planet. Actions then take place to acquire and strengthen a player's assets through **Attack**, **Victory**, **Transfer** and **Recruit**.

Top left are printed the **Galaxy Grid x,y** coordinates, in the window below a sides **Star Ship** assets **Fleet size, Food/Ores/Tec/Trade** credits and **MODs** (Machines Of Destruction). Below this are the updated **Interactive** or **Simulation** Players assets, Land based **MODs,** Star Ships in **Fleet**, Ship based **MODs,** followed by the number of **Territories** annexed. Top right the **Star** Systems **Name**, the Planets **Allegiance**, **Population** in Billions, **Food/Ores/Tec/Trade** credits and **MODs.** Below this, the updated display of the **Opponents** side assets.

The **Galaxy Grid** displays the annexed Star Systems of the two sides in different coloured circles, the numbers displayed for each side increase or decrease as the game progresses. To either side are two symbols, a double circle above a graphic representation used for the Dice Roll. The colours of the outer circles of these symbols change to show the Star Ship side and Star System playing against each other.

At the bottom left the **Skill** level and on the right **GTS, G**ame **T**ime **S**econds as they clock up to show the duration of the Game. The **Galaxy Symbol** in between is coloured differently when in **Interactive** or **Simulator Mode**.

QBITS Galaxy AD2370 - The Game

The aim is to reduce your Opponent's Territories to zero in the fastest time. The object therefore is in acquiring Territories. The strategy deployed depends on your level of assets. A major break at the start of play is how you are dealt your opening hand. Ten Territories (Star Systems) are allocated to each side, they are randomly sourced with Food/Ores/Tec/Trade credits & MODs. However, the total number of MODs is the same for both sides the difference always being made up as extra Star Ship MODs given to the side with the least Land based MODs. This can give an initial advantage and if fortunate enough in numbers you might simply target your opponent's Territories and annex then as quickly as possible.

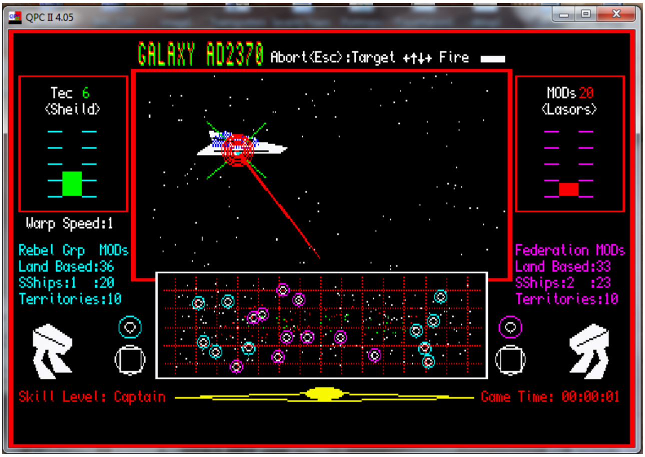
Similarly you could target as many Territories as you can with low MODs (i.e. with three MODs or less) thereby increasing the opportunity of survival. An alternatively would be to Transfer Star Ship Food/Ores/Tec/Trade credits to increase the Population and Recruit more MODs from your own Territories. Thereby with increased credits and MODs available, making a Transfer of some back to your Star Ships for further expansion, and leaving others to better defend your Territories.

QBITS Galaxy AD2370 - Skills & Hazards.

The **Skill** level is an increase in difficulty accomplished by reducing the time between completing each task. In addition, Galaxy Version 5 now has **Hazards.** Some just add delays to the game as you bypass a Galactic feature such as a **Pulsar**, a **Binary Star System**, or **Nebula.** There again watch out for the **Black Hole**, you could lose a Star Ship together with 20 MODs.

Maybe you not doing so well and on the verge of being defeated, well Good NEWS you could enter a **Timeslip** and you and your opponents assets are swapped. The Bad News is it could happen to you on the verge of winning.

Encounter is where you battle it out in space with your opponent. You lose one of your **MODs** every time you **fire.** If the enemy fires back, you lose **Shield Strength**, which is linked to your number of **Tec** credits. Good NEWS, destroy your enemy and you quickly gain more **MODs** and **Tec** credits.



GALAXY AD2370 Abort<Esc>:Target ←↑↓→ Fire

Tec 6
<Sheild>

MODs 20
<Lasors>

Warp Speed:1

Rebel Grp MODs
Land Based:36
SShips:1 :20
Territories:10

Federation MODs
Land Based:33
SShips:2 :23
Territories:10

Skill Level: Captain

Game Time: 00:00:01

QBITs Galaxy ADA2370v5 - Code Development

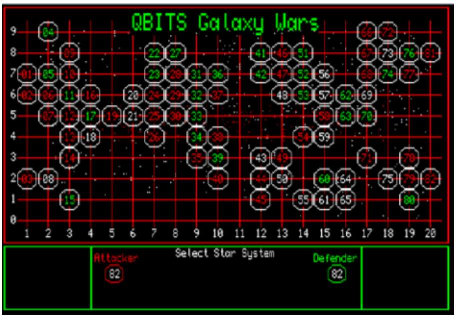
I didn't consciously consider before writing my Game Programs, that I might work to a framework. However, something of such has evolved. The three-act play scenario used in writing fiction applies just as well it appears to developing Game code : **Act 1The Welcome** : **Act 2 The Challenge : Act 3 Game End**.

[Act 1] Welcome:- **Intro** page to describe the basic game and its function. **Initialisation** of Graphic displays and Parameters in line with opening choices made from a **Menu.** For **Galaxy AD2370** they are **Skill Levels** and **Sides,** plus the expected (**N**)ew (**L**)oad (**S**)ave much of which I had developed in earlier progs. Then a (**H**)ighscore and Game (**E**)xit.

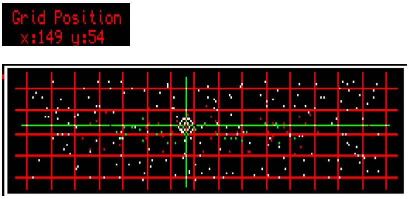
[Act 2] Challenge:- subdivides into multiple acts derived from **Review** of information displayed and taking the appropriate choice of **Action** in order to progress. Then analysis of success or failure so as to determine further moves.

[Act 3] Game End:- when the challenge has been met a sequence depicting a F**inal Battle** between the opposing sides. Then a **League Table,** showing the high scores of players coupled with the ability to update if a newly declared **Winner**.

QBITS Galaxy AD2370 - Simple Beginnings



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


Grid Position
x:149 y:54

This changed to a Galaxy Map seeded with Random POINTs (Stars) with the Grid Position of a selected Star highlighted with a circle and the meeting point of horizontal & vertical cross-wires. Within a REPeat loop these could be moved under control of the cursor keys.

I had always imagined graphical representations would be part of the actions taking place. The Galaxy Grid now reduced to a smaller screen area opened the opportunity to create a Space window with moving Stars and the other actions I had in mind. My initial display created the illusion of movement by subdividing the view screen in to four quarters with randomly placed POINTs (Stars). Then using the PAN and SCROLL commands moved them out towards the four corners.

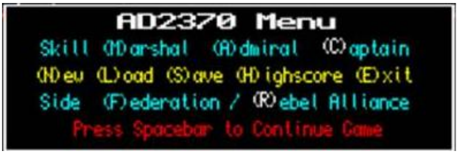
The Title and Intro blurb plus a few notes on Order of Play and this was about as far as I had progressed back in the nineteen-eighties.

QBITS Galaxy AD2370 - Windows Initialisation

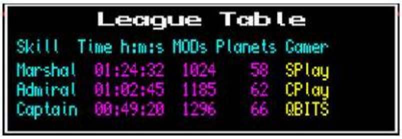
After experimenting with an expanding number of subdivided areas and varying number of Stars (POINTs), I eventually decided on eight overlapping areas that make up the main view screen (WINDOWs #3 to #10). Surrounding this are areas displaying information linked to current actions within the Game (WINDOWs #11 to #18). The **Galaxy Grid (** WINDOW #13) when playing the Game is also used for the **AD2370 Menu**, and the **Highscore League Table**.

QBITS Galaxy AD2370 - Menu

The **Highscore League Table** is accessed from the **AD2370 Menu** and includes hidden key options (**D**) & (**d**) allowing the **League Table** to be set to **Default Scores** or **Reset**. Pressing (**M**) in Game mode will bring you back to the **AD2370 Menu** with the option to take a short coffee break change your **Skill Level** and **Sides** if you so wish, before pressing Spacebar to continue your Game.

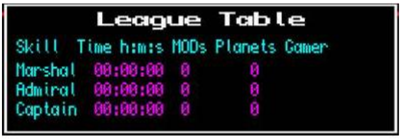

AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance

Press Spacebar to Continue Game



League Table

Skill	Time h:m:s	MODs	Planets	Gamer
Marshal	01:24:32	1024	58	SPlay
Admiral	01:02:45	1185	62	CPlay
Captain	00:49:20	1296	66	QBITS



League Table

Skill	Time h:m:s	MODs	Planets	Gamer
Marshal	00:00:00	0	0	
Admiral	00:00:00	0	0	
Captain	00:00:00	0	0	

QBITS Galaxy AD2370 – Where When Who

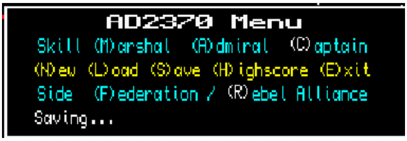
Keeping track of the actions can become complex and requires variables and arrays to carry fixed and updated information. The Star System arrays **astro$(70,20)** & **astro(70,11)** hold the **Name** and **Popu**lation/**Food/Ores/Tec/Trade//MODs** and other details. The Star Systems **Allegiance** held in **astro(n,7)** are designated as follows:- **0 Independent** colour - **INK 6**, **1** for **Rebel Alliance** - **INK 5** and **2** for the **Federation - INK 3**.

A sides status of **Food/Ores/Tec/Trade/MOD** are held by array **side(2,5).** Where variable **Gp1** is the **Interactive or Simulation Player** and **Gp2 is** the **Opponent**. Either **side(Gp1,n)** or **side(Gp2,n)** can be set for **Rebel Alliance** or **Federation.** That is **Gp1**=**1** (**Rebel Alliance**) **:Gp2**=**2** (**Federatiion**) or the reverse **Gp1**=**2** & **Gp2**=**1** etc**.**

The double circle symbols either side of **Galaxy Grid** (WINDOW#13) identifies who is playing against whom. The outer circle colour on the left shows the current Player's Move:- **Rebel Alliance** or **Federation.** On the right in Attack Mode, it shows the Planets opening allegiance, **Independent**, **Rebel Alliance** or **Federation**. A **Victory** will change the allegiance colour to the attacker's colours to show it has been annexed.

QBAD2370v5D-1 to 8 Data Files

The **SAVE** function stores inventories for both Star Systems **astro$(a) astro(a,b)** and Sides **side(a,b)**, **GTS** Game Time, **TD1** Skill chosen and side to play on **Gp1 & Gp2**.


AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance

Saving...

QBITs Galaxy ADA2370 - PROCedures Brief Overview

Part One

AD2370Intro /GTitle/GDev/LTData	Galaxy AAD2370 Title : Select Default Device : Load League Table
AD2370Info /MPrn/GIcon	Opening Scene : Print Game info messages to screen : Galaxy Icon
GWinInit	Initialising of Game Layout (WINDOWS#1 to #18)
GMenu /GSelct/GMPrn	Select **Skill** / **Side** / (**N**)ew (**L**)oad (**S**)ave (**H**)ighscore (**E**)xit **:** Print to screen
GExit	Closes and clears screen #3 to #18 and returns with 'bye...'.
GNewInit	Seeds new Galaxy Populations Food/Ores/Tec/Trade & MODs
GMapInit/LMap	Creates Galaxy Map of Star Systems : Displays owned Star Systems
SelPath / GSave /GLoad /FChck	Selection and access of Device & File

Part Two

GAD2370	Main Game Loop **-** Initiates when each side plays in which Hazards are encountered. Access to Game Information (**G**) or return to (**M**)enu. Plus **F1,F2,F3,F4,F5** see page 18.
GPlay	Controls the game moves for **Interactive, Simulation or Opponent**. Simulation and Opponent can be set to play against each other checking program Stratagems.
GTime	Displays Game Time Clock
Hazards	Initiates random choice of Stellar events
/Pulsar/Binary/Nebula	Graphic displays showing the bypass of Stellar features.
/Blackhole	Avoid, if you go to close you could lose a Star Ship, crew and 20 MODs
/Timeslip	Swaps sides If one or other sides has 20 more Territories than the other.
Encounter	Destroy to win 20 MODs and extra Tec credits lose an MOD for every shot fired.
/EncInit/EncUD	Sets Displays for Shield (Tec) and Lasers (MODs). Updates their strengths.
/Enemy/ Sights/Laser/Hit	Selects Enemy Ship /Target with Sights /Fire Laser /Destroy with direct Hit
GameTest	Tests Graphic HyperJump to Planet Conquest and on to Game End and Scores
PMap	Show cross-wires of selected Star System
MODInfo MODInit	Shows – Territories / Star Ships / MODs. Initialises MODs count.
PTInfo SSInfo	Displays Planet & Star Ship Information Population/Food/Ores/Tec/Trade/MODs
HJump /StarMove/	Space HighperJump to Star System : Background Star movement
/StarView /Star	Star approach
EOrbit /Phaser /Victory	Enter Orbit of Planet :Attack Mode fire Phaser : Gain additional credits and MODs
/TranSP /Recruit	Transfer Mode-Star Ship to Planet : Increase Population recruit etc.
/TranPS	Transfer Mode-Planet to Star Ship
LOrbit	Leave Orbit

Part Three

GEnd	Game End initialisation
/DeathStar	DeathStar enters Planetary Orbit
/DSAttack	Planet destroyed by DeathStar
/DSDefeat	Deathstar destroyed by Rebels
LTDefault /LTRest	Set League Table to Default Scores. Reset League Table.
/LScore	Displays HighScore League Table
/LName	Enter New Highscore and New Winners Name
/LSave LLoad	Saves Updated League Table. Loads League Table (see start of program code).
GNotes	Galaxy AD2370 Notes on Game control keys
RollDice /DiceA / DiceB	Graphics Simulated Roll of the dice
MODL /MODR	MODs images left & right
SS01 /SS02 /SS03	DeathStar / Rebel SShip / Federation SShip

QBITS GALAXY AD2370v5 - The Code

Most Games will use a number of Dimensioned Arrays and Global Variables that carry changes to the **Status** and **Information** associated with **Actions** within the Game. Those shown below cover most of the **Actions** within **Galaxy AD2370**. As you read through the following pages there usage will be explain in more detail. Declaring them at the beginning is usual in other BASICs and helps identify any anomalies through misuse especially as the game goes through its **Development Phase**.

```
1000 REMark QBAD2370v5QPC2  (QBITS Galaxy AD2370 v5.2 2018)

1002 MODE 4
1003 DIM Dev$(8,5),Dat$(8,13) :RESTORE 1005   :REMark Device & Data Files
1004 FOR dn=1 TO 8:READ Dev$(dn):Dat$(dn)='QBAD2370v5D_'&dn
1005 DATA 'flp1_','flp2_','win1_','win2_','dos1_','dos2_','nfa1_','nfa2_'
1006 d=1:mch=0:sch=1:file=1                   :REMark Device:MODInit: New/Load/Save checks

1008 DIM Comp(8,2):RESTORE 1010               :REMark Encounter Compass Directions
1009 FOR c=1 TO 8:READ Comp(c,1),Comp(c,2)
1010 DATA 46,10,92,10,138,10,10,50,46,90,92,90,138,90,174,50

1012 DIM astro$(70,20),astro(70,11)           :REMark Star/Planet 1-70 Info
1013 DIM side(2,5),Gclk$(20)                  :REMark Game Players Info : GTS h:m:s
1014 DIM name$(4,10),score(3,3):nch=0         :REMark HighScore League Table     Note nch - F2 Check

1016 Gp=0:Gp1=0:Gp2=0:Gs=0 :GSk=3             :REMark Game Player Side & Skill checks
1017 RMOD=0:RSSn=1:RPs=10                     :REMark Rebel Alliance MODs:SShips:Planets
1018 FMOD=0:FSSn=1:FPs=10                     :REMark Federation      MODs:SShips:Planets
1019 dn=0:ach=1:t=1:rch=1                     :REMark RollDice: Attack/Transfer:(R)ecruit

1021 DMs=3:ws=1:Mes$='         '              :REMark Delet MODs:Warp Speed:Messages
1022 GTS=0:TD1=60:TD2=20                      :REMark Game Time Seconds : Time Delays

1024 REMark Part 1 The Intro and Setup

1026 AD2370Intro:GMenu

1028 DEFine PROCedure AD2370Intro
1029 OPEN#20,scr_:WINDOW#20,512,256,0,0:BORDER#20,2,2:PAPER#20,0:CLS#20
1030 WINDOW#1,496,220,8,10:PAPER#1,0:CLS#1:SCALE#1,120,0,0
1031 ch=1:x=132:y=8:Title$='QBITS GALAXY AD2370':CSIZE#ch,2,1:GTitle:GDev
1032 END DEFine

1034 DEFine PROCedure GTitle
1035 OVER#ch,-1:FOR a=1 TO 2:INK#ch,a*2:CURSOR#ch,x+a,y+a:PRINT#ch,Title$
1036 OVER#ch,0:CSIZE#ch,0,0
1037 END DEFine
```

Note:

I wonder if it is still the norm for students learning Basic computer programming to write their first code as:- PRINT ‘**Hello world**’. For me Printing text to the screen quickly progressed from **Title** to the **Intro** blurb that conveyed a Games aims and the control keys to use.

QBITS Galaxy AD2370 - Intro

The Intro page might consist of lines of text using a single font. However, there is no reason not to embellish it with an imaginative Title to spice things up a little.

QBITS Title

CSIZE used with OVER and a CURSOR offset presents the opportunity to create a 3D Style character effect. With different colour arrangements, result can be impressive.

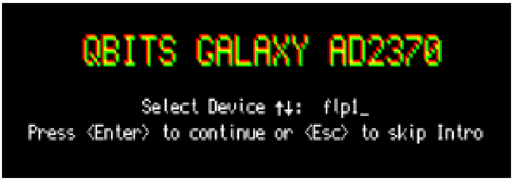
```
WINDOW#1,512,256,0,0 :PAPER#1,0:CLS#1:CSIZE#1,2,1:OVER -1
INK#1,2:FOR i=1 TO 3:CURSOR#1,140+i,20-i:PRINT#1,'QBITS Galaxy AD2370'
INK#1,7:FOR i=1 TO 2:CURSOR#1,140+i,20-i:PRINT#1,'QBITS Galaxy AD2370'
OVER 0:CSIZE 0,0:
```

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

QBITS Galaxy AD2370 - Storage Device

For **Galaxy AD2370** to run on different platforms there was the need to identify access to different storage **Devices**. The **Device** selected was for the **Save/Load** of the **Highscore League Table**, and the **Game Data files.** After starting up the Game a few times; I recognised a need to skip the **Intro**. These are dealt with by the procedure **GDev**.

```
1039 DEFine PROCedure GDev
1040 ch=1:CSIZE#ch,0,0:INK#ch,7
1041 CURSOR#ch,172,40:PRINT#ch,'Select Device ↑↓:'
1042 CURSOR#ch,100,52:PRINT#ch,'Press <Enter> to continue or <Esc> to skip Intro'
1043 REPeat lp
1044   CURSOR#ch,286,40:PRINT#ch,Dev$(d)
1045   k=CODE(INKEY$(5))
1046   SELect ON k
1047     =208:d=d+1:IF d>8:d=1
1048     =216:d=d-1:IF d<1:d=8
1049     = 10:LTData:AD2370Info:EXIT lp
1050     = 27:LTData:EXIT lp
1051   END SELect
1052 END REPeat lp
1053 LLoad:GWinInit:d=1:GMenu
1054 END DEFine
```



QBITS GALAXY AD2370

Select Device ↑↓: flp1_

Press <Enter> to continue or <Esc> to skip Intro

Note:d= Device 1 to 8 1005 **DATA** 'flp1_','flp2_','win1_','win2_','dos1_','dos2_','nfa1_','nfa2_'

QBITS Galaxy AA2370 - Game Control

For the most part access and control conforms with **Retro Games** in using the four cursors keys, **Left, Right, Up, Down,** for direction and the **Spacebar**, **Enter** and **Esc**ape keys to select and activate further actions. These were the keys and motions provided with early **Joystick controllers**. You may also have noted that they are keys in row one of the **SupeBASIC** KEYROW matrix. The use of k=CODE(INKEY$(f)), as opposed to using k=KEYROW(1) appears to give a more controlled response running emulators on today's much higher clock speed computers. To use KETROW(1) with a PAUSE value, which counts in units of 20 milliseconds, does not provide SuperBASIC REPeat with the same loop control and key access as achieved with INKEY$(f) where f is in frame counts.

Running **QBITS Galaxy AD2370** for the first time **LTData** checks for the existence of the **Highscore League Table** file on the selected device. **FChck**, part of the **Load** and **Save** Procedures checks if the file is present. If not, one is created with **LTDefault**. This saves having to supply a separate file with the Game program. There is also a Procedure **LTReset**, actioned by key (**d**) from the **AD2370** M**enu** that resets the **Highscore League Table** to zeros for a fresh start.

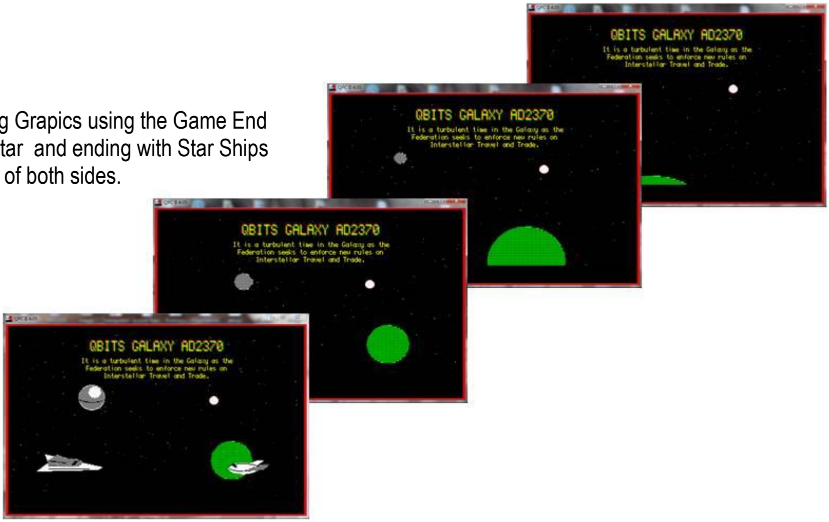
```
1056 DEFine PROCedure LTData
1057 Dv$=Dev$(d):Gf$='QBAD2370DLT':ch=1: FChck:IF file=0:LTDefault
1058 END DEFine
```

QBITS Galaxy AD2370 - Opening Scene

Artistic licence or not, I guess this is a matter of choice and limited only by the level of understanding and use of QL SuperBASIC commands. To create a background a simple approach is to use 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 the Game Intro.

For **Galaxy AD2370** I wanted an opening statement to be lines of text printed to screen giving the illusion of somebody typing words in real time. To follow this a sequence of moving Graphics, then after a short pause finish with further text giving an overview of the Games moves and actions.

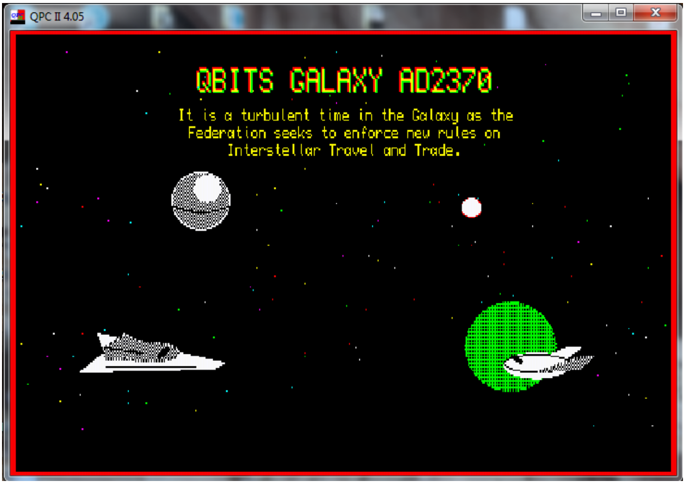
If you chose to continue as you probably would the first time you run **QBITS Galaxy AD2370**, then you are presented with a combination of text and graphics. First, a sprinkling of Stars to fill the screen, then the opening message printed to screen character by character. This is followed by graphics used in the Game. The display of a local **Star** and then the **DeathStar** moves into view from deep space as a **Planet** looms up from the bottom of the screen. **Star Ships** for each side appear and following a short pause, the rest of the introduction text to **QBITS Galaxy AD2370** is displayed.



QBITS GALAXY AD2370

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

Opening Grapics using the Game End DeathStar and ending with Star Ships Images of both sides.

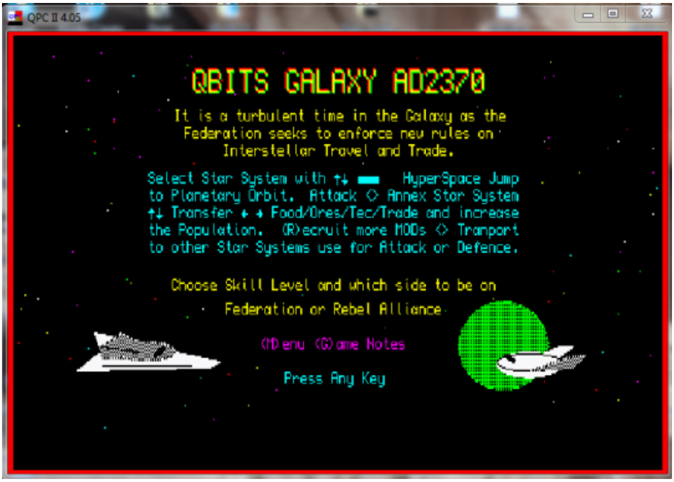


QBITS GALAXY AD2370

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

QBITS Galaxy AD2370 - Info Text
In deciding to printing character by character to screen as if it were being typed, I used **DATA** lines with variables **i, c, r, s, str$:** where **i** is **the** INK colour, **c** column, **r** row, **s** the string length and **str$** holds the string of text. Procedure **AD2370Info** contains a **FOR loop** which **READs** the **DATA** and Procedure **MPrn** uses the read Data to set INK **i**, CURSOR position **c, r** and contains a second **FOR loop b**=1 to **s** to print the **str$** character by character to screen.

```
1060 DATA 6,120, 34,43,'It is a turbulent time in the Galaxy as the'
1061 DATA 6,128, 44,40,'Federation seeks to enforce new rules on'
1062 DATA 6,158, 54,30,'Interstellar Travel and Trade.'
1063 DATA 5,100, 70,48,'Select Star System with ¾¿       HyperSpace Jump'
1064 DATA 5,100, 80,48,'to Planetary Orbit.  Attack <> Annex Star System'
1065 DATA 5,100, 90,48,'¾¿ Transfer ¼ ½ Food/Ores/Tec/Trade and increase'
1066 DATA 5,100,100,48,'the Population.  (R)ecruit more MODs <> Transport'
1067 DATA 5,100,110,48,'to other Star Systems use for Attack or Defence.'
1068 DATA 6,118,132,42,'Choose Skill Level and which side to be on'
1069 DATA 6,160,146,28,'Federation or Rebel Alliance'
1070 DATA 3,186,166,19,'(M)enu (G)ame Notes',5,206,186,13,'Press Any Key'

1072 DEFine PROcedure AD2370Info
1073 BLOCK#1,400,30,0,40,0
1074 FOR Stars=1 TO 250:INK RND(2 TO 7):POINT RND(0 TO 280),RND(0 TO 140)
1075 RESTORE 1060:FOR a=1 TO 3:READ i,c,r,s,str$:MPrn
1076 n=20:astro(n,10)=36:Gp=1:sx=140:sy=70:sr=3:Star:DeathStar
1077 ink1=7:ink2=248:x=160:y=20:SS02:x=38:SS03:PAUSE 50
1078 FOR a=4 TO 12:READ i,c,r,s,str$:MPrn:IF a=4:BLOCK 16,4,264,74,5
1079 PAUSE:CURSOR#1,9,118:FOR a=1 TO 16:SCROLL#1,-8,1:SCROLL#1,6,2:PAUSE 1
1080 END DEFine

1082 DEFine PROCedure MPrn
1083 INK i:CURSOR c,r:FOR b=1 TO s:PAUSE .5:PRINT str$(b);
1084 END DEFine
```



QBITS GALAXY AD2370

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

Select Star System with ↑↓ ▬ HyperSpace Jump
to Planetary Orbit. Attack <> Annex Star System
↑↓ Transfer ← → Food/Ores/Tec/Trade and increase
the Population. (R)ecruit more MODs <> Tranport
to other Star Systems use for Attack or Defence.

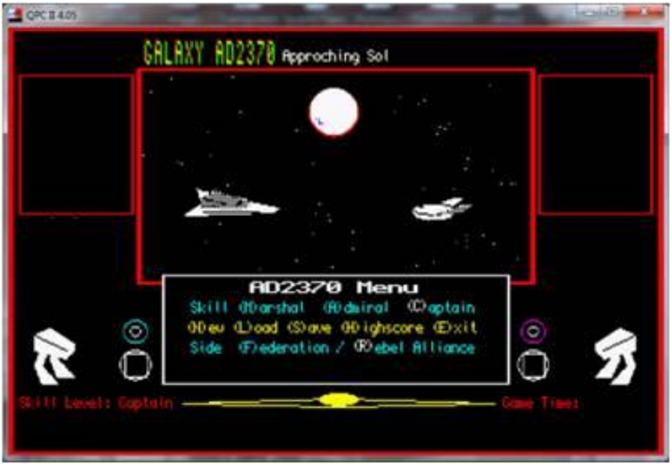
Choose Skill Level and which side to be on

Federation or Rebel Alliance

(M)enu (G)ame Notes

Press Any Key

There again you may have chosen to skip straight to the **Initialising** of the **Galaxy AD2370 Game**.



GALAXY AD2370 Approching Sol

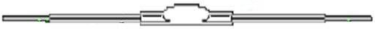
AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance

Skill Level: Captain

Game Time:

QBITS Galaxy AD2370 - Game Icon

The **Icon** is a feature I feel no **Game** should be without! This one could be described as an ellipse sitting superimposed on a flattened diamond and representing a Sunset or Star Eclipse take your pick**.** In the Game, it represents a symbol of Galactic Authority. The **G**alaxy **Icon** appears at the bottom of the Game screen coloured in **INK 6** for **Interactive** Player and **INK 2** for **Simulation** Mode.

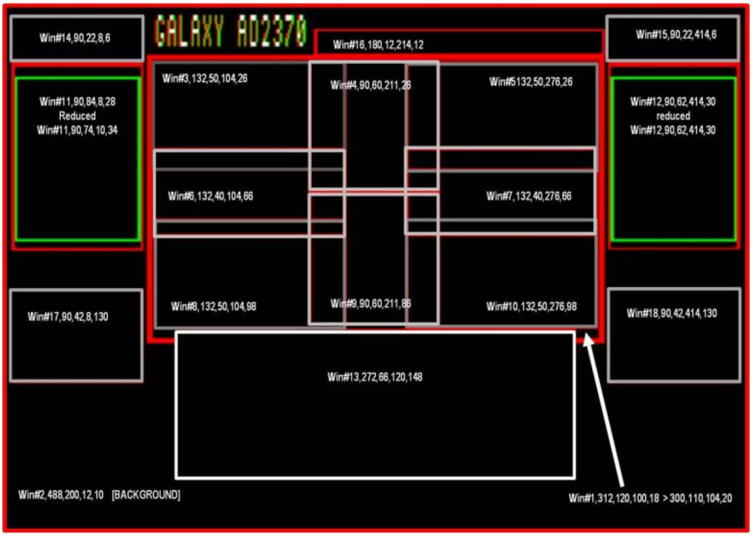


```
1086 DEFine PROCedure GIcon
1087 ch=2:x=82:y=2:IF SPch=0:INK#ch,6:ELSE INK#ch,2
1088 LINE#ch,x-40,y TO x,y+1 TO x+40,y TO x,y-1 TO x-40,y
1089 FILL#ch,1:CIRCLE#ch,x,y+1,5,.3,PI/2:FILL#ch,0
1090 END DEFine
```

QBITS Galaxy AD2370 – Game Screen

GWINInit sets up the window areas and adds a few additional graphics. In the space following the **Galaxy AD2370** Title; messages appear relevant to the current actions. The central view screen is subdivided into eight overlapping WINDOWS#3-10, used by Graphic Displays they create the illusion of moving through space. The WINDOW#13 below is used for the **AD2370 Menu** and when in Game Mode the **Galaxy Grid**. When selected, Top left #14 displays a **Stars Grid x,y** location, and on the right #15 the accessed **Star Systems Name**. On the left, i#11 **Star Ship Info**, to the right, #12 **Planet Info,** below these update of each **Side in** WINDOWS #17 & #18.

.



```
1092 DATA 132,50,104,26,90,60,211,26,132,50,276,26        :REMark win 3-10
1093 DATA 132,40,104,66,132,40,276,66
1094 DATA 132,50,104,98,90,60,211,86,132,50,276,98
1095 DATA 90,84,8,28,90,84,414,28,272,66,120,148          :REMark win 11-13
1096 DATA 90,22,8,6,90,22,414,6,198,12,214,12             :REMark win 14-16
1097 DATA 90,42,8,130,90,42,414,130                       :REMark win 17-18

1099 DEFine PROCedure GWinInit
1100 file=1:Gp1=1:Gp2=2:RESTORE 1092
1101 WINDOW#2,496,226,8,4:PAPER#2,0:INK#2,7:CLS#2
1102 ch=2:x=96:y=0:Title$='GALAXY AD2370':CSIZE#2,1,1:GTitle
1103 FOR ch=3 TO 18:OPEN#ch,scr_:READ a,b,c,d:WINDOW#ch,a,b,c,d
1104 WINDOW#1,312,130,100,24:BORDER#1,2,2:PAPER#1,0:CLS#1
1105 WINDOW#1,300,120,106,26:WINDOW#0,496,20,8,230:SCALE#1,100,0,0
1106 BORDER#11,1,2:WINDOW#11,86,80,10,30
1107 BORDER#12,1,2:WINDOW#12,86,80,416,30
1108 BORDER#13,1,7:PAPER#13,0:CLS#13:Gp=1:SPch=0:GIcon
1109 INK#2,7:CIRCLE#2,30,21,1.25:CIRCLE#2,132,21,1.25
1110 MODL:MODR:dx=30:DiceA:dx=132:DiceA
1111 sn=20:astro$(sn)='Sol ':astro(sn,10)=228:astro(sn,11)=2
1112 INK#16,7:StarView:ink1=7:ink2=248:x=40:y=30:SS03:x=140:SS02
1113 END DEFine
```

QBITS Galaxy AD2370 - Menu

The **Menu** allows selection of Game **Skill** (**M**)arshal (**A**)dmiral (**C**)aptain plus a choice of side (**F**)ederation or (**R**)ebles**.** The **Skill** choice sets different time delays and boundaries to add difficulty to playing the Game. **GSkill** sets these values, **GMprn** displays the choices as highlighted characters.

The other options are (**N**)ew (**L**)oad (**S**)ave (**H**)ighscore (**E**)xit. There are two hidden options to set the **Highscore League Table** (**D**)efault or (**d**) Reset. The save check variable **sch**=**1** to stop any use of (**S**)ave until the use of (**N**)ew or (**L**)oad has been carried out, when **sch**=**0.**

```
1115 DEFine PROCedure GMenu
1116 CURSOR#2,378,216:PRINT#2,'Game Time:        '
1117 CURSOR#2,6,110:PRINT#2,’            ‘:FOR ch=11 to 15:CLS#ch
1118 ch=13:h:CSIZE#ch,2,0:INK#ch,7:OVER#ch,1
1119 FOR i=1 TO 2:CURSOR#ch,64+i,2:PRINT#ch,'AD2370 Menu'
1120 OVER#ch,0:CSIZE#ch,0,0: SPch=0:GIcon:GSkill:GMPrn
1121 REPeat lp                Note: SPch is the Simulation Player check Off=0 / On=1.
1122   k=CODE(INKEY$(-1))
1123   SELect ON k
1124     =77,109 :TD1=20                :GSkill:GMPrn      :REMark (M)arshal
1125     =65, 97 :TD1=40                :GSkill:GMPrn      :REMark (A)dmiral
1126     =67, 99 :TD1=60                :GSkill:GMPrn      :REMark (C)aptain
1127     =70,102 :Gp1=2:Gp2=1           :GSkill:GMPrn      :REMark (F)ederation
1128     =82,114 :Gp1=1:Gp2=2           :GSkill:GMPrn      :REMark (R)ebel Alliance
1129     =78,110 :mch=1:nch=0           :GNewInit          :REMark (N)ew
1130     =76,108 :               SelPath :GLoad             :REMark (L)oad
1131     =83,115 :IF sch=0:      SelPath :GSave:GMPrn       :REMark (S)ave
1132     =72,104 :               LScore  :PAUSE:GMenu       :REMark (H)ighscore
1133     =68      :LTDefault:LScore      :PAUSE:GMenu       :REMark (D)efault Highscore [Hidden]
1134     =100     :LTReset  :LScore      :PAUSE:GMenu       :REMark (d)Reset Highscore[Hidden]
1135     =69,101 :               GExit                      :REMark (E)xit
1136     =32      :IF sch=0:     GMapInit:GAD2370           :REMark SpaceBar Continue Game
1137   END SELect
1138 END REPeat lp                                          Note variable mch=1 activates MODInit
1139 END DEFine                                             Note: nch=0 League Table update on
```

```
1141 DEFine PROCedure GSkill
1142 IF TD1=20:Tm$='Marshal':GSk= 1:xsk=  56:sk$='(M)':mx=4:my=2
1143 IF TD1=40:Tm$='Admiral' :GSk= 2:xsk=122:sk$='(A)':mx=6:my=3
1144 IF TD1=60:Tm$='Captain' :GSk= 3:xsk=188:sk$='(C)':mx=8:my=4
1145 IF Gp1=1:xsd=146:sd$='(R)':ELSE xsd=56:sd$=’(F)’
1146 INK#2,7:CIRCLE#2,30,21,3:CIRCLE#2,132,21,3
1147 INK#2,2:CURSOR#2,0,216:PRINT#2,'Skill Level: ';Tm$
1148 END DEFine
                                                            Note: GSk=1/2/3 League Table Game Skill
```

The variables **mx,my** identify boundaries for the Fire range used by Procedure **Encounter.**

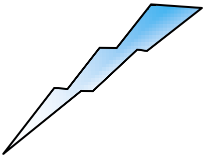
```
1150 DEFine PROCedure GMPrn
1151 INK#ch,5:CURSOR#ch,20,14:PRINT#ch,'Skill   (M)arshal   (A)dmiral   (C)aptain'
1152 INK#ch,6:CURSOR#ch,16,26:PRINT#ch,'(N)ew (L)oad (S)ave (H)ighscore (E)xit'
1153 INK#ch,5:CURSOR#ch,20,38:PRINT#ch,'Side   (F)ederation  /  (R)ebel Alliance'
1154 INK#ch,7:CURSOR#ch,xsk,13:PRINT#ch,sk$:CURSOR#ch,xsd,37:PRINT#ch,sd$
1155 IF sch=0:INK#ch,2:CURSOR#ch,40,50:PRINT#ch,'Press Spacebar to Continue Game'
1156 END DEFine
```

QBITS Galaxy AD2370 - Exit

In leaving the Game, the responsibility is to reset the environment, closing any opened channels and cleaning up the screen. When developing you need to leave the program code available for editing hence **GExit.** However, to clear a Program code, any variables and channels other than #0,#1 and #2 and return to the SuperBASIC command interpreter you only need use the NEW command.

```
1158 DEFine PROCedure GExit
1159 CLOSE#99:CLOSE#20:FOR w=3 TO 18:CLOSE#w
1160 INK#2,7:INK#1,7:WINDOW#1,512,256,0,0:PAPER#1,0:CLS#1
1161 INK#0,7:WINDOW#1,490,220,16,8:PRINT#0,'bye...':
1162 END DEFine
```

```
1164 DATA 'Vega','Bellatrix','Zibal','Castor','Naos','Artos','Ilad'
1165 DATA 'Tabit','Diphda','Elnath','Acrux','Errai','Cellus','Detrus'
1166 DATA 'Polaris','Homan','Rigel','Sabik','Sarin','Westron','Zenda'
1167 DATA 'Furud','Gienah','Hadar','Propus','Izar','Centra','Nebtron'
1168 DATA 'Enif','Algol','Mirzarm','Libeta','Atria','Panus','Ulan'
1169 DATA 'Keid','Kochab','Lesath','Ankaa','Marsic','Nebus','Alean'
1170 DATA 'Meissa','Subra','Menkib','Altair','Muscida','Pusan','Curson'
1171 DATA 'Cursa','Nashira','Nunki','Ogma','Pollux','Xilon','Scarb'
1172 DATA 'Sirius','Merack','Serena','Denab','Tureis','Noran','Theron'
1173 DATA 'Beid','Wesen','Yildun','Caph','Zooma','Jarron','Cetrus'
```

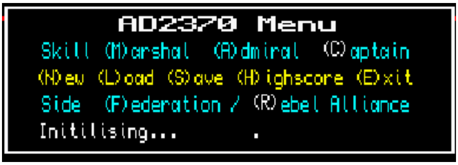


NEW GAME

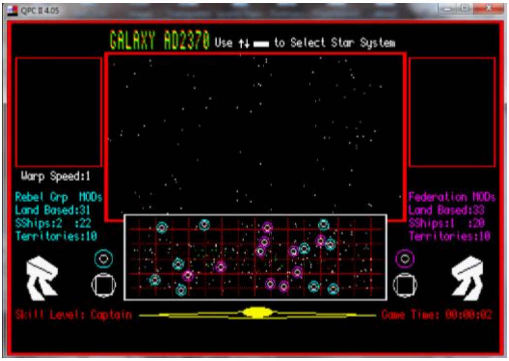
```
1175 DEFine PROCedure GNewInit
1176 RESTORE 1164:ch=13:CURSOR#ch,18,50:PRINT#ch,'Initilising';:CLS#ch,4
1177 FOR n=1 TO 70
1178   READ a$:astro$(n)=a$:CURSOR#ch,86+n*2,50:PRINT#ch,'.';
1179   Popu=RND(2 TO 16)/RND(4 TO 8)          :REMark Population 0.5-8bn
1180   astro(n,1)=INT(Popu+Popu*.9)            :REMark Food  max 20
1181   astro(n,2)=INT(1+Popu)                  :REMark Ores  max 10
1182   astro(n,3)=INT(Popu*1.12)               :REMark Tec   max 12
1183   astro(n,4)=INT((Popu+astro(n,2))/2)     :REMark Trade max 20
1184   astro(n,5)=INT((Popu+astro(n,3))/1.1)   :REMark MODs  max 20
1185   astro(n,6)=Popu                         :REMark Population max 10bn
1186   astro(n,7)=0                            :REMark Side
1187   astro(n,8)=n*4+15                       :REMark x Grid coordinate
1188   astro(n,9)=RND(10 TO 90)                :REMark y Grid coordinate
1189   astro(n,10)=RND(24 TO 46)               :REMark Planet Colour
1190   astro(n,11)=RND(1 TO 3)                 :REMark Orbital Approach
1191 END FOR n
1192 REMark Side astro(n,5) 0=Independent 1=Rebel Alliance 2=Federation
1193 FOR i=1 TO 10
1194   n=RND(15 TO 55):IF astro(n,7)=0:astro(n,7)=2:ELSE GO TO 1194
1195 END FOR i
1196 FOR i=1 TO  5
1197   n=RND(  5 TO 20):IF astro(n,7)=0:astro(n,7)=1:ELSE GO TO 1197
1198   n=RND(50 TO 65):IF astro(n,7)=0:astro(n,7)=1:ELSE GO TO 1198
1199 END FOR i
1200 FOR n=1 TO 70:IF astro(n,7)>0 AND astro(n,5)<3:astro(n,5)=3
1201 FOR Gp=1 TO 2 :side(Gp,1)=4:side(Gp,2)=2:side(Gp,3)=6:side(Gp,4)=6
1202 sch=0:GMapInit:LMap:GAD2370
1203 END DEFine
```

QBITS Galaxy AD2370 – New Game

GNewInit generates a new set of Game Data for the seventy Star Systems. The Sides at start of play are allocated 10 **Territories** (Star Systems) together with **Food/Ores/Tec/Trade** credits and **MODs**. **GMapInit** initialises the Galaxy Grid. **LMap** displays the two sides territories circled in their colours**.**



```
AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance
Initilising...        .
```



```
QPC II 4.05
GALAXY AD2370 Use ↑↓ ▬ to Select Star System
Warp Speed:1
Rebel Grp  MODs                       Federation MODs
Land Based:31                         Land Based:33
SShips:2  :22                         SShips:1  :20
Territories:10                        Territories:10
Skill Level: Captain                  Game Time: 00:00:02
```

QBITS Galaxy AD2370 – Galaxy Grid

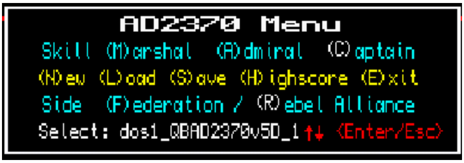
```
1205 DEFine PROCedure GMapInit
1206 ch=13:CLS#ch:BORDER#ch,1,7:INK#ch,0,2,3:SCALE#ch,100,0,0
1207 FOR h=12 TO 96 STEP 18
1208  LINE#ch,6,h TO 310,h
1209 END FOR h
1210 FOR v=16 TO 330 STEP 20
1211  LINE#ch,v,3 TO v,98
1212 END FOR v
1213 INK#ch,2:FOR i=1 TO 60 :POINT#ch,RND(60 TO 260),RND(30 TO 70)
1214 INK#ch,4:FOR i=1 TO 30 :POINT#ch,RND(80 TO 240),RND(40 TO 60)
1215 INK#ch,7:FOR i=1 TO 180:POINT#ch,RND(10 TO 300),RND(10 TO 90)
1216 ch=16:CLS#ch:INK#ch,7
1217 PRINT#ch,'Use↑↓        to Select Star System':BLOCK#ch,16,4,40,4,7
1218 ch=1:CLS#ch:INK#ch,7
1219 FOR i=1 TO 120:POINT#ch,RND(10 TO 200),RND( 5 TO 95)
1220 END DEFine

1222 DEFine PROCedure LMap
1223 LOCal n,nx,ny:ch=13
1224 FOR n=1 TO 70
1225   nx=astro(n,8):ny=astro(n,9):INK#ch,7
1226   IF astro(n,7)=1:CIRCLE#ch,nx,ny,3:INK#ch,5:CIRCLE#ch,nx,ny,6
1227   IF astro(n,7)=2:CIRCLE#ch,nx,ny,3:INK#ch,3:CIRCLE#ch,nx,ny,6
1228 END FOR n
1229 END DEFine
```

QBITS Galaxy AD2370 – Save & Load

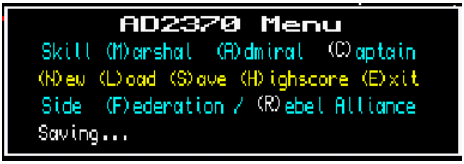
Entering directly the device and file name can lead to typo's contributing to errors halting the program. To avoid this instead of using the SuperBASIC INPUT command I decided upon eight selectable and reusable DATA files to hold the Game Information.

SelPath uses the default device selected from **GDev** and appends a data file name, selection is made using the Up and Down cursor keys. **GSave** deletes the old file and opens it as a new file to save the Games Info **astro$(a), astro(a,b), side(a,b)** arrays and Global variables **GTS** (**G**ame **T**ime **S**aved) **TD1** Current **T**ime **D**elay setting and the deployment of sides via **GP1** & **GP2**.

```
1231 DEFine PROCedure SelPath
1232 ch=13:INK#ch,2:CURSOR#ch,0,50:CLS#ch,4
1233 CURSOR#ch,176,50:PRINT#ch,' ↑↓ <Enter/Esc>':INK#ch,7
1234 REPeat Path_lp
1235  CURSOR#ch,18,50:PRINT#ch,'Select: ';Dv$&Dat$(d)          Note: Dat$(8,13) holds QBAD2370v5D_1-8
1236  k=CODE(INKEY$(-1))
1237  SELect ON k
1238   =208:d=d+1:IF d>8:d=1
1239   =216:d=d-1:IF d<1:d=8
1240   = 10:file=1:EXIT Path_lp
1241   = 27:file=0:EXIT Path_lp
1242  END SELect
1243 END REPeat Path_lp
1244 device_filename$=Dv$&Dat$(d):Gf$=Dat$(d)
1245 END DEFine
```



AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance
Select: dos1_QBAD2370v5D_1 ↑↓ <Enter/Esc>

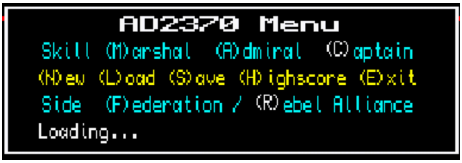


AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance
Saving...

```
1247 DEFine PROCedure GSave
1248 ch=13:IF file=0:CURSOR#ch,0,50:CLS#ch,4:RETurn
1249 DELETE device_filename$
1250 CURSOR#ch,18,50:PRINT#ch,'Saving...':CLS#ch,2:CLS#ch,4
1251 OPEN_NEW#99,device_filename$
1252 FOR a=1 TO 70
1253   CURSOR#ch,78+a*2,50:PRINT#ch,'.';:PRINT#99,astro$(a)
1254   FOR b=1 TO 11:PRINT#99,astro(a,b):END FOR b
1255 END FOR a
1256 FOR a=1 TO 2
1257   FOR b=1 TO 5:PRINT#99,side(a,b) :END FOR b
1258 END FOR a
1259 PRINT#99,GTS\TD1\Gp1\Gp2:CLOSE#99:CURSOR#ch,0,40:CLS#ch,2
1260 END DEFine
```

To Load a previously saved file; **SelPath** allows the selection of one of eight Game Files. If one is selected <**Enter**> **file**=**1** if not <**Esc**> **file**=**0**. **GLoad** will return if variable **file**=**0**. This is either because <**Esc**> was pressed in **SelPath** or **FChck** returns 'File Not Found..' If **FChck** retuns with **file**=**1** it uses the variable **device_filenmae$** to open and read the file entries into **astro$(a), astro(a,b), side(a,b)** arrays and Global variables **GTS, TD1, GP1, GP2** used by the Game.

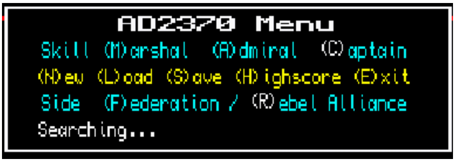
```
1262 DEFine PROCedure GLoad
1263 ch=13 :IF file=0:CURSOR#ch,0,50:CLS#ch,4:RETurn
1264 FChck:IF file=0:CURSOR#ch,0,50:CLS#ch,4:RETurn
1265 OPEN_IN#99,device_filename$
1266 ch=13:CURSOR#ch,18,50:PRINT#ch,'Loading...':CLS#ch,4
1267 FOR a=1 TO 70
1268   CURSOR#ch,78+a*2,50:PRINT#ch,'.';:INPUT#99,astro$(a)
1269   FOR b=1 TO 11:INPUT#99,astro(a,b):END FOR b
1270 END FOR a
1271 FOR a=1 TO 2
1272   FOR b=1 TO 5:INPUT#99,side(a,b) :END FOR b
1273 END FOR a
1274 INPUT#99,GTS\TD1\Gp1\Gp2:CLOSE#99:CURSOR#ch,0,40:CLS#ch,2
1275 sch=0:nch=0:GSkill:GMapInit:GAD2370
1276 END DEFine
```



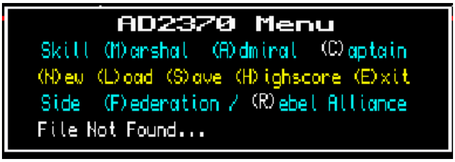
AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance
Loading...

FChck deletes any existing FList file in the default device, then generates a new FList file and saves to it the Directory of filenames. It then checks through the list of filenames reading each into **Fchk$** to find any match with the selected Game File **Gf$** from **SelPath.** If **EOF** (End Of File) is reached it returns 'File Not Found...' with variable **file**=**0**.

```
1278 DEFine PROCedure FChck
1279 CURSOR#ch,18,50:PRINT#ch,'Searching...':CLS#ch,4
1280 DELETE Dv$&'FList'
1281 OPEN_NEW#99,Dv$&'FList':DIR#99,Dv$:CLOSE#99
1282 OPEN_IN#99,Dv$&'FList'
1283 REPeat Dir_lp
1284   IF EOF(#99)
1285     CLOSE#99:CURSOR#ch,18,50:PRINT#ch,'File Not Found...'
1286     PAUSE 50:CURSOR#ch,0,50:CLS#ch,4:file=0:slk=1:RETurn
1287   END IF
1288   INPUT#99,Fchk$:IF Fchk$==Gf$:CLOSE#99:EXIT Dir_lp
1289 END REPeat Dir_lp
1290 END DEFine
```



AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance
Searching...



AD2370 Menu
Skill (M)arshal (A)dmiral (C)aptain
(N)ew (L)oad (S)ave (H)ighscore (E)xit
Side (F)ederation / (R)ebel Alliance
File Not Found...

```
1292 REMark Part 2 Playing the Game AD2370

1294 DEFine PROCedure GAD2370
1295 Gclk=DATE:clk1=DATE+10:clk2=DATE+TD1:clk3=DATE:SPch=0
1296 MODInfo:LMap:n=RND(20 TO 50):gx=-9:gy=-9:PMap:haz=0:haz2=0
1297 REPeat lp
1298  GTime:StarMove:k=CODE(INKEY$(5))
1299  IF DATE>clk1:Hazards                          :clk1=DATE+10
1300  IF DATE>clk2:              Gp=Gp2:  GPlay :clk2=DATE+TD1 :REMark Opponents  Move
1301  IF DATE>clk3 AND SPch=1  Gp=Gp1:  GPlay :clk3=DATE+TD2 :REMark Simulator Move
1302  SELect ON k
1303   =32      :IF SPch=0       : Gp=Gp1    :GPlay                  :REMark [SBar] Players Star Choice
1304   =71,103 :GNotes                                                :REMark (G)ame Notes
1305   =77,109 :GTS=(DATE-Gclk+GTS)      :GMenu                 :REMark Skill/Side/NEW/Load/Save
1306   =208     :IF SPch=0 AND n> 1:n=n- 1  :PTInfo                :REMark move↑ Up
1307   =216     :IF SPch=0 AND n<70:n=n+1  :PTInfo                :REMark move↓ Down
1308   =232     :IF SPch=0:SPch=1:GIcon:ELSE SPch=0:GIcon        :REMark F1 Simulator
1309   =236     :sn=n::GameTest                                       :REMark F2 Graphics Test
1310   =240     :TimeSlip                                             :REMark F3 Cheat-Test
1311   =244     :Encounter:CLS#16                                     :REMark F4 Cheat-Test
1312   =248     :side(Gp1,3)=12:side(Gp1,5)=90:MODInfo              :REMark F5 Cheat-Tec/MODs
1313  END SELect
1314 END REPeat lp
1315 END DEFine
```

First half of **GPlay** deals with the **Interactive** Game Player Moves...

```
1317 DEFine PROCedure GPlay
1318 ach=1:t=1:rch=1:Mes$=' ':INK#2,7:CURSOR#2,408,110
1319 IF SPch=1 AND Gp=Gp1:PRINT#2,'Simulator Move   ':GO TO 1344
1320 IF                    Gp=Gp2:PRINT#2,'Opponents Move':GO TO 1344
1321 PRINT#2,'Players Move      ':SSInfo:sn=n:HJump:StarView:EOrbit
1322 REPeat act
1323  IF FPs=0 OR RPs=0:EXIT act
1324  IF ach=1 AND astro(n,7)=Gp:ach=0:Mes$='↑↓ Transfer ←→':SSInfo
1325  IF ach=1 AND astro(n,5)=   0:ach=0:Mes$='↑↓ Transfer ←→':Victory
1326  IF ach=1 AND side(Gp,5)>DMs+1
1327    Mes$=' Attack Mode ':SSInfo:RollDice:SSInfo:CLS#16
1328  END IF
1329  IF ach=1 AND side(Gp,5)<DMs+1
1330     ach=0:PRINT#16,'Not enough MODs':PAUSE 25:EXIT act
1331  END IF
1332  IF s=0:ch=11:BLOCK#ch,10,50,0,20,0:AT#ch,t+1,0:PRINT#ch,'>'
1333  GTime:k=CODE(INKEY$(5))
1334  SELect ON k
1335   =192:TranPS                 Note Transfer phase Planet to Star Ship
1336   =200:TranSP                 Note Transfer phase Star Ship to Planet
1337   =208:t=t-1:IF t<1:t=5        Note t used to select Food/Ores/Tec/Trade MODs to Transfer
1338   =216:t=t+1:IF t>5:t=1
1339   = 32:EXIT act
1340   = 82,114:IF ach=0 AND rch=1:Recruit:rch=0           Note ach rch Attack & Recruit checks
1341  END SELect
1342 END REPeat act
1343 GO TO 1376
```

QBITS Galaxy AD2370 – Game

GAD2370 **REPeat lp** provides the main access to playing the Game. **GPlay** controls the action taken by **Interactive** (human Player), **Simulator** or **Opponent.** Select **Star System HyperJump** to **E**nter **Orbit** and if not already annexed, **Attack**. With **Victory** or if Territory already annexed, then Game continues with **Transfer** of **Food/Ores/Tec/Trade** credits to raise the Population. Press key (**R) ecruit** to generate more **Planet** credits and **MOD**s. Further **Transfer** between **Planet** and **Star Ships** can then take place planning for future **Attack**s and raising defences for other Territories. A move ends with return to space and **L**eaving **Orbit**. For **Hazards** see next page.

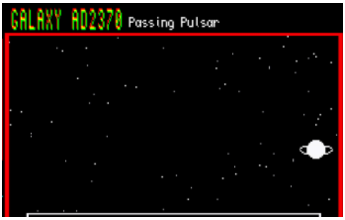
The second half of **GPlay** deals mainly with Computer **Opponent** and **Simulator** Moves...

```
1344 SSInfo:IF side(Gp,5)>=20:n=RND(1 TO 70)
1345 IF side(Gp,5)<20
1346   FOR i=1 TO 10
1347     n=RND(1 TO 70):IF astro(n,5)>10 AND astro(n,7)=Gp:EXIT i
1348   END FOR i
1349 END IF
1350 IF Gp=1 AND FPs>30:FOR i=1 TO 5:n=RND(1 TO 70):IF astro(n,7)=2:EXIT i
1351 IF Gp=2 AND RPs>30:FOR i=1 TO 5:n=RND(1 TO 70):IF astro(n,7)=1:EXIT i
1352 sn=n:PTInfo:HJump:StarView:EOrbit
1353 IF astro(n,7)=Gp
1354   Mes$='↑↓ Transfer← →':SSInfo
1355   astro(n,1)=astro(n,1)+INT(side(Gp,1)/2):IF astro(n,1)>20:astro(n,1)=20
1356   astro(n,2)=astro(n,2)+INT(side(Gp,2)/2):IF astro(n,2)>10:astro(n,2)=10
1357   astro(n,3)=astro(n,3)+INT(side(Gp,3)/2):IF astro(n,3)>12:astro(n,3)=12
1358   astro(n,4)=astro(n,4)+INT(side(Gp,4)/2):IF astro(n,4)>20:astro(n,4)=20
1359   SSInfo:PAUSE 20:Recruit:rch=0
1360   IF side(Gp,5)<20 AND astro(n,5)>6
1361      side(Gp,5)=side(Gp,5)+astro(n,5)-3
1362   END IF
1363   IF side(Gp,5)>50 AND astro(n,5)<10
1364     astro(n,5)=astro(n,5)+5:side(Gp,5)=side(Gp,5)-5
1365   END IF
1366 ELSE
1367  Mes$=' Attack Mode ':SSInfo:PAUSE 20
1368  IF astro(n,5)>side(Gp,5)*3 OR side(Gp,5)<10:GO TO 1376
1369  RollDice:PAUSE 20
1370  IF astro(n,5)=0
1371    PRINT#16,astro$(n);' Falls'
1372    Victory:PAUSE 20:astro(n,7)=Gp:LMap:GO TO 1353
1373  END IF
1374  SSInfo:PAUSE 20:GO TO 1368
1375 END IF
1376 SSInfo:PAUSE 50:LOrbit:PMap:gx=-9:gy=-9:n=RND(20 TO 50)
1377 ws=1:CLS#11:CLS#12:CLS#14:CLS#15:CLS#16
1378 CURSOR#2,408,110:PRINT#2,'            '
1379 IF FPs=0 OR RPs=0:GEnd
1380 END DEFine

1382 DEFine PROCedure GTime
1383 Gclk$=DATE$(DATE-Gclk+GTS):INK#2,2:CURSOR#2,444,216:PRINT#2,Gclk$(13 TO 20)
1384 END DEFine
```

```
Game Time: 00:00:23
```

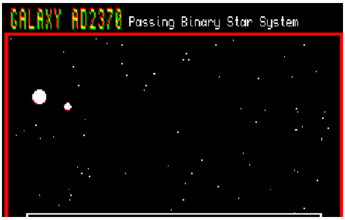
```
1386 DEFine PROCedure Hazards
1387 haz=RND(2 TO 7)
1388 IF haz=haz2 :RETurn :ELSE haz2=haz
1389 IF haz= 2 :Pulsar    :CLS#16          Note haz 2/3/4 creates time delays to Game
1390 IF haz= 3 :Binary    :CLS#16
1391 IF haz= 4 :Nebula    :CLS#16
1392 IF haz= 5 :Blackhole:CLS#16           Note haz 5 fly to near and lose 20MODs
1393 IF haz= 6 AND DATE>clk2
1394    IF Gp1=1 AND RPs<FPs+20:RETurn
1395    IF Gp1=2 AND FPS<RPs+20:RETurn
1396    TimeSlip:clk2=DATE+TD1             Note haz 6 swaps assets between players Lose or Gain
1397 END IF
1398 IF haz= 7 AND SPch=0:Encounter        Note haz 7 destroy enemy Star Ship Lose or Gain Tec & MODs
1399 END DEFine
```

```
1401 DEFine PROCedure Pulsar
1402 LOCal px,py,pr:pr=.3:ws=1:PRINT#16,'Passing Pulsar'
1403 FOR i=1 TO 32
1404    px=90+i*3:py=90-i*2:pr=i/6:GTime:StarMove
1405    INK 241:FILL 1:CIRCLE px,py,pr      :FILL 0:PAUSE 3
1406    INK    7:FILL 1:CIRCLE px,py,pr*1.1:FILL 0:PAUSE 2
1407    CIRCLE px,py,pr*2,.3,PI/2:PAUSE 2
1408    INK   0:FILL 1:CIRCLE px,py,pr*2  :FILL 0
1409 END FOR i
1410 END DEFine
```



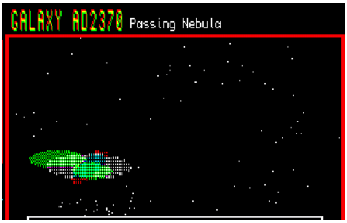
GALAXY AD2370 Passing Pulsar

```
1412 DEFine PROCedure Binary
1413 LOCal a1,a2,bx,by,rx,ry:a1=180:a2=0:rx=12:ry=4
1414 ws=1:PRINT#16,'Passing Binary Star System'
1415 FOR i=1 TO 40
1416    bx=72-i*2:by=50+i/2
1417    x1=bx-rx*SIN(RAD(a1)):y1=by-ry*COS(RAD(a1))
1418    x2=bx-rx*SIN(RAD(a2)):y2=by-ry*COS(RAD(a2))
1419    sx=x1:sy=y1:sr=i/12:Star:a1=a1+6:IF a1>360:a1=0
1420    sx=x2:sy=y2:sr=  i/6:Star:a2=a2+6:IF a2>360:a2=0
1421    PAUSE 2+i/4:INK 0:FILL 1:CIRCLE x1,y1,i/4:FILL 0
1422    FILL 1:CIRCLE x2,y2,i/2:FILL 0:GTime:StarMove
1423 END FOR i
1424 END DEFine
```



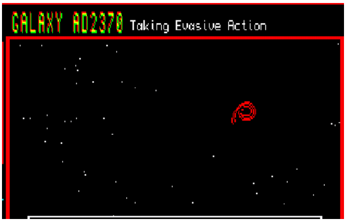
GALAXY AD2370 Passing Binary Star System

```
1426 DEFine PROCedure Nebula
1427 LOCal x1,y1,x2,y2,x3,y3,x4,y4
1428 ws=1:PRINT#16,'Passing Nebula'
1429 FOR a=1 TO 40
1430    x1=92-a*3:y1=50-a*1.5:x2=90-a*4:y2=48-a*1.1
1431    x3=94-a*3:y3=48-a*1.5:x4=92-a*3:y4=50-a*1.5:OVER -1
1432    INK 0,7,0:FILL 1:CIRCLE x1,y1,a*1.5,.3,PI/2:FILL 0
1433    INK 0,4,3:FILL 1:CIRCLE x2,y2,a,.3,PI/2:FILL 0
1434    INK 0,4,3:FILL 1:CIRCLE x3,y3,a*.7,.4,PI/2:FILL 0
1435    INK 0,2,0:FILL 1:CIRCLE x4,y4,a*.8,.4,-PI/4:FILL 0:OVER 0
1436    PAUSE 4+a/10:INK 0:FILL 1:CIRCLE x1,y1,a*4,.3,PI/2:FILL 0
1437    GTime:StarMove:INK 7:FOR i=1 TO 5:POINT RND(30 TO 80),RND(5 TO 40)
1438 END FOR a
1439 END DEFine
```



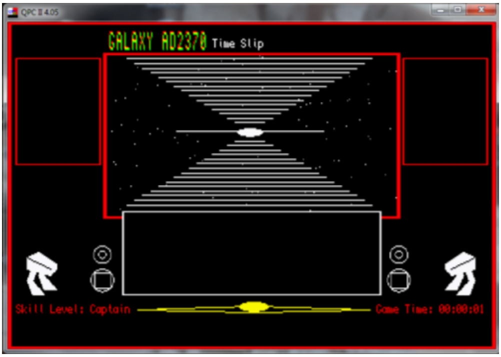
GALAXY AD2370 Passing Nebula

```
1441 DEFine PROCedure Blackhole
1442 ws=1:x=160:y=90:PRINT#16,'WARNING! >>> Blackhole <<<'
1443 FOR a=8 TO 20:BHole:x=x-a/3:y=y-a/5:GTime:StarMove:PAUSE 5
1444 IF side(Gp1,5)>50
1445     side(Gp1,5)=side(Gp1,5)-20:PRINT#16,'One Ship and Crew Lost'
1446 ELSE
1447    PRINT#16,'Taking Evasive Action'
1448 END IF
1449 FOR a=20 TO 8 STEP -1:BHole:x=x+a:y=y-a/2:GTime:StarMove:PAUSE 2
1450 StarMove:MODInfo
1451 END DEFine
```



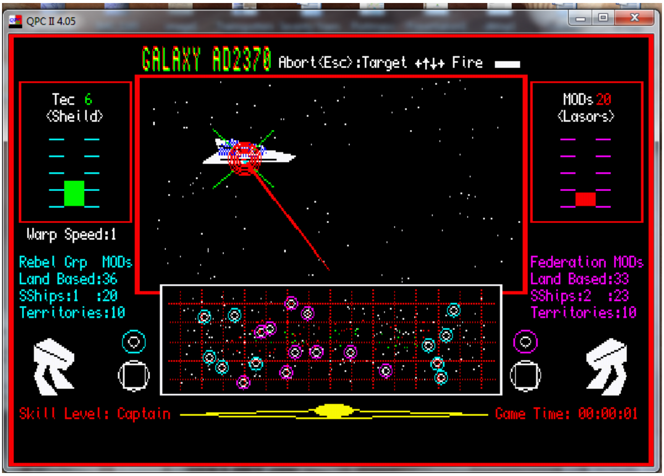
Note: If MODs are 50 or more expect to lose a **Star Ship** and crew together with **20 MODs**.

```
1453 DEFine PROCedure BHole
1454  OVER -1:INK 2:CIRCLE x,y,a*.3,.8,PI/2
1455  ARC x,y+a*.2 TO x-a*.3,y,PI/2 TO x,y-a*.3,PI/2 TO x+a*.4,y,PI/2 TO x,y+a*.4,PI/2 TO x-a*.6,y-a*.6,PI/2
1456  ARC x,y+a*.2 TO x-a*.4,y,PI/2 TO x,y-a*.4,PI/2 TO x+a*.5,y,PI/2 TO x,y+a*.5,PI/2 TO x-a*.7,y-a*.7,PI/2
1457  OVER 0:PAUSE 5:INK 0:FILL 1:CIRCLE x,y,a*4:FILL 0
1458 END DEFine
```



Note: TimeSlip Reverses the fortunes of each side for Good or Bad.

```
1460 DEFine PROCedure TimeSlip
1461 FOR i=12 TO 18:CLS#i
1462 PRINT#16,'Time Slip':cursor#2,6,110:print#2,'              ‘
1463 FOR a=1 TO  8:FILL 1: CIRCLE 92,50,a,.3,PI/2:FILL 0:PAUSE 3
1464 FOR n=1 TO 70
1465    IF astro(n,7)=1:astro(n,7)=2:ELSE IF astro(n,7)=2:astro(n,7)=1
1466 END FOR n
1467 FOR i=1 TO 5
1468    a=side(1,i):side(1,i)=side(2,i):side(2,i)=a
1469 END FOR i
1470 FOR b=16 TO 2 STEP -1
1471    LINE 92-b*3,50 TO 92+b*3,50
1472    LINE 92-b*5,50+b*3 TO 92+b*5,50+b*3
1473    LINE 92-b*5,50-b*3  TO 92+b*5,50-b*3:PAUSE 2
1474 END FOR b
1475 GTime:GMapInit:LMap:gx=-9:gy=-9:PMap:MODInfo
1476 END DEFine
```



GALAXY AD2370 Abort<Esc>:Target ←↑↓→ Fire

Tec 6
<Sheild>

MODs 20
<Lasors>

Warp Speed:1

Rebel Grp MODs
Land Based:36
SShips:1 :20
Territories:10

Federation MODs
Land Based:33
SShips:2 :23
Territories:10

Skill Level: Captain

Game Time: 00:00:01

Encounter gives the opportunity to engaging in a good old **Retro Dog Fight**. Line up your **Sights** <cursor keys **Left Right Up Down**> take aim and fire <**spacebar**>. If you destroy your **Enemy** you gain a **Star Ship** with **20 MODs** and more **Tec** credits. Your enemy can also reduce your **Tec** credits by weakening your **Shield** strength and every time you fire on your enemy reduces your **MODs** count by one. However to avoid any confrontation you can always abort with the <**Esc**> key.

```
1478 DEFine PROCedure Encounter
1479 IF side(Gp1,3)<=0 OR side(Gp1,5)<=0:RETurn
1480 PRINT#16,'Abort<Esc>:Target ←↑↓→ Fire':BLOCK#16,20,4,172,4,7
1481 Des=1:tx=92:ty=30:x=RND(80 TO 100):y=RND(40 TO 60)
1482 CLS#12:CLS#14:CLS#15:EncInit
1483 REPeat enc
1484   INK 4:Sights:ink1=7:ink2=241:Enemy
1485   IF RND(57)=13:Phaser:side(Gp1,3)=side(Gp1,3)-1
1486   k=CODE(INKEY$(5)):GTime:EncUD
1487   INK 0:Sights:ink1=0:ink2=0:Enemy
1488   IF side(Gp1,3)<=4 OR side(Gp1,5)<=6:EXIT enc
1489   IF Des=2:PRINT#16,'Enemy Destroyed':EXIT enc
1490   SELect ON k
1491    = 32:Laser:side(Gp1,5)=side(Gp1,5)-1:MODInfo:Hit
1492    = 27:PRINT#16,'Aborting Attack':EXIT enc
1493    =192:tx=tx -mx:IF tx< 10:tx= 10
1494    =200:tx=tx+mx:IF tx>170:tx=170
1495    =208:ty=ty+my:IF ty> 90:ty= 90
1496    =216:ty=ty -my:IF ty< 10:ty= 10
1497   END SELect
1498   IF x>164 OR x<20 OR y<20 OR y>90:c=RND(1 TO 8):cx=Comp(c,1):cy=Comp(c,2)
1499   IF x>cx:x=x-2:ELSE x=x+2
1500   IF y>cy:y=y-1:ELSE y=y+1
1501   INK 7:FOR i=1 TO 3:POINT x+RND(-5 TO 5),y+RND(-2 TO 2)
1502 END REPeat enc
1503 PAUSE 100:CLS#11:CLS#12:CLS#14:CLS#15:CLS#16
1504 END DEFine
```

Note Target **tx,ty** used to move **Sights**
Note mx, my set area boundaries for direct **Hit**

For a more realistic movement of the enemy's **Star Ship,** coordinates of an eight point compass are loaded into array Comp(8,2) . The enemy randomly selects one of the **cx, cy** sets of coordinates and moves toward them. Upon reaching a boundary, it selects an alternative set (see Line 1498)

EncInit initialises the display, location appearance and direction of travel for the enemy Star Ship **EncUP** Displays the Up Dates of **Sheild** strength and **MODs** available.

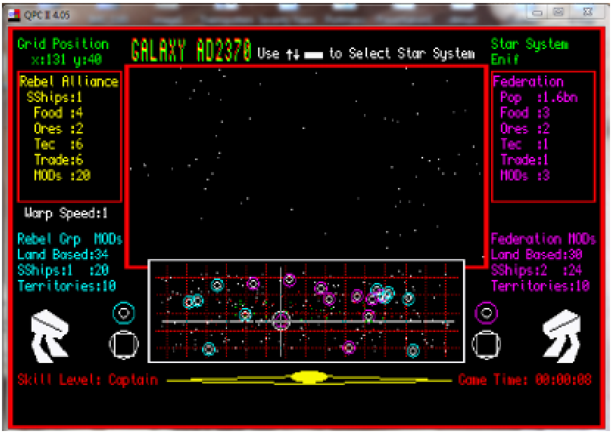
```
1506 DEFine PROCedure EncInit
1507 INK#12,7:CURSOR#12,24,4:PRINT#12,'MODs:'\'   <Lasors>'
1508 INK#11,7:CURSOR#11,24,4:PRINT#11,'Tec :'\'   <Sheild>'
1509 FOR i=1 TO 5
1510   BLOCK#11,40,1,22,22+i*10,5,0,1:BLOCK#12,40,1,22,22+i*10,3,0,1
1511 END FOR i
1512 c=RND(1 TO 8):cx=Comp(c,1):cy=Comp(c,2)          Note Random Selection of compass point cx,cy
1513 x=RND(80 TO 104):y=RND(40 TO 60)                 Note Enemy Star Ship x,y appearance coordinates
1514 END DEFine

1516 DEFine PROCedure EncUD
1517 INK#12,2:CURSOR#12,50,4:PRINT#12,side(Gp1,5);'  '
1518 INK#11,4:CURSOR#11,50,4:PRINT#11,side(Gp1,3);'  '
1519 Mh=INT(side(Gp1,5)*.4):Th=INT(side(Gp1,3)*2.5)   Note Mh Th MODs & Tec Bar height
1520 BLOCK#12,16,48,34,32,0:BLOCK#11,16,48,34,32,0
1521 BLOCK#12,16,Mh,34,72-Mh,2:BLOCK#11,16,Th,34,72-Th,4
1522 END DEFine

1524 DEFine PROCedure Enemy
1525 IF Des=2:RETurn                                  Note Des=2 returns as enemy has been destroyed.
1526 IF Gp1=1:SS03                                    Different enemy Star Ship images are deployed
1527 IF Gp1=2:SS02                                    dependant on which side Rebel or Federation.

1528 END DEFine

1530 DEFine PROCedure Sights
1531 LINE tx-15,ty+15 TO tx-5,ty+5:LINE tx+5,ty+5 TO tx+15,ty+15
1532 LINE tx-15,ty -15 TO tx-5,ty -5:LINE tx+5,ty -5 TO tx+15,ty -15
1533 END DEFine

1535 DEFine PROCedure Laser
1536 h=92:v=5:INK 2
1537 FOR i=1 TO 6
1538   OVER -1:FILL 1:LINE h,v TO tx-1,ty TO tx+1,ty TO h,v:FILL 0:OVER 0:PAUSE .5
1539 END FOR i
1540 END DEFine

1542 DEFine PROCedure Hit
1543 IF tx>x-mx AND tx<x+mx AND ty>y-my AND ty<y+my          Note mx,my Sights boundary limits
1544   INK 2:FOR i=3 TO 8:CIRCLE tx,ty,i:PAUSE .5
1545   INK 0:FOR i=8 TO 3 STEP -1:CIRCLE tx,ty,i:PAUSE .5       Note tx,ty Target x,y aim
1546   IF side(Gp1,3)<10:side(Gp1,3)=side(Gp1,3)+6
1547   IF side(Gp1,5)<80:side(Gp1,5)=side(Gp1,5)+20
1548   Des=2:MODInfo
1549 END IF
1550 END DEFine
```

To check the Graphics I added a short test Procedure **GameTest.**

```
1552 DEFine PROCedure GameTest
1553 Pulsar:Binary:Nebula:Blackhole:HJump GTime:StarView:Gp=Gp1
1554 EOrbit:Phaser:LOrbit: FOR n=1 to 70:astro(n,5)=13:astro(n,7)=Gp
1555 LMap:side(Gp,5)=63:MOFInfo:IF Gp1=1:FPs=0:ELSE RPs=0
1556 sch=1:nch=1:GEnd
1557 END DEFine
```

Note: nch=1 blocks League Table updates   sch=1 reset of save

Search a Star System ⇅
Select **SpaceBar**



Grid Position
x:131 y:40

GALAXY AD2370 Use ⇅ ▬ to Select Star System

Star System
Enif

Rebel Alliance
SShips:1
Food :4
Ores :2
Tec :6
Trade:6
MODs :20

Federation
Pop :1.6bn
Food :3
Ores :2
Tec :1
Trade:1
MODs :3

Warp Speed:1

Rebel Grp MODs
Land Based:34
SShips:1 :20
Territories:10

Federation MODs
Land Based:30
SShips:2 :24
Territories:10

Skill Level: Captain

Game Time: 00:00:08

Note PMap - Cross-wires to Highlight Star System Selected

```
1559 DEFine PROCedure PMap
1560 ch=13:INK#ch,7:OVER#ch,-1
1561 LINE#ch,gx,5 TO gx,95:LINE#ch,12,gy TO 306,gy
1562 CIRCLE#ch,gx,gy,3:INK#ch,7:CIRCLE#ch,gx,gy,9:OVER#ch,0
1563 END DEFine
```

```
1565 DEFine PROCedure MODInfo
1566 RMOD=0:RPs=0:RSSn=0:FMOD=0:FPs=0:FSSn=0
1567 FOR i=1 TO 70:IF astro(i,7)=1:RPs=RPs+1:RMOD=RMOD+astro(i,5)
1568 FOR i=1 TO 70:IF astro(i,7)=2:FPs=FPs+1:FMOD=FMOD+astro(i,5)
1569 IF mch=1:MODInit:mch=0
1570 IF Gp1=1:Rbch=17:Fdch=18:ELSE Rbch=18:Fdch=17
1571 CLS#17:CLS#18:RSSn=1+INT(side(1,5)/21):FSSn=1+INT(side(2,5)/21)
1572 INK#Rbch,5:PRINT#Rbch,'Rebel Grp  MODs'\'Land Based:';RMOD\'SShips:';RSSn;' :';side(1,5)\'Territories:';RPs
1573 INK#Fdch,3:PRINT#Fdch,'Federation  MODs'\'Land Based:';FMOD\'SShips:';FSSn;' :';side(2,5)\'Territories:';FPs
1574 END DEFine
```

Note mch=1 Initialise MODs count.

```
1576 DEFine PROCedure MODInit
1577 IF RMOD>FMOD:side(2,5)=20+RMOD-FMOD:side(1,5)=20
1578 IF FMOD>RMOD:side(1,5)=20+FMOD-RMOD:side(2,5)=20
1579 IF FMOD=RMOD:side(1,5)=20:side(2,5)=20
1580 END DEFine
```

Selecting a Star System displays its Planetary status (astro) and current Player's (side) Star Ship Resources.

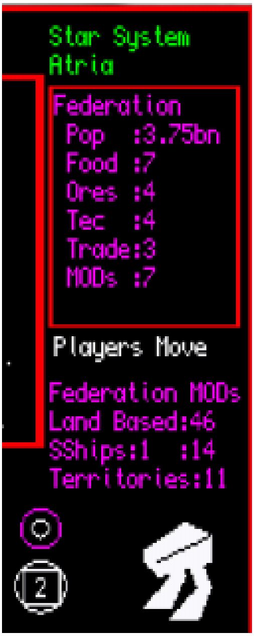
```
1582 DEFine PROCedure PTInfo
1583 PMap:ch=12   :side$='Independant   ':INK#ch,6:INK#2,6
1584 IF astro(n,7)=1:side$='Rebel Alliance':INK#ch,5:INK#2,5
1585 IF astro(n,7)=2:side$='Federation     ':INK#ch,3:INK#2,3
1586 gx=astro(n,8):gy=astro(n,9):          CIRCLE#2,132,21,3
1587 PRINT#14,'Grid Position'\'  x:';gx;' y:';gy
1588 PRINT#15,'Star System'\astro$(n)
1589 CLS#ch:Pop$=astro(n,6):PopL=LEN (Pop$)
1590 IF PopL>2:FOR i=1 TO PopL:IF Pop$(i)='.':PopL=i+2
1591 PRINT#ch,side$
1592 PRINT#ch,' Pop   :';Pop$(1 TO PopL);'bn'
1593 PRINT#ch,' Food  :';astro(n,1)
1594 PRINT#ch,' Ores  :';astro(n,2)
1595 PRINT#ch,' Tec   :';astro(n,3)
1596 PRINT#ch,' Trade:';astro(n,4)
1597 PRINT#ch,' MODs:';astro(n,5)
1598 PMap
1599 END DEFine
```

MODInfo

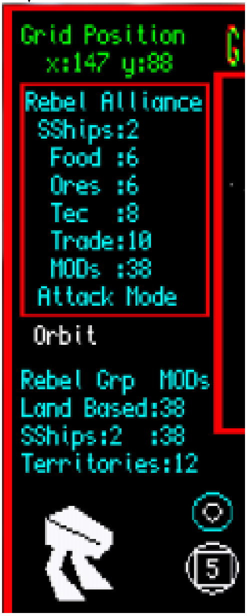
Note Lines 1583 -1586 **INK#2,CIRCLE** of Planet Colours

Star System Allegiance colours **INK 3 Federation**

Dice Roll



```
Star System
Atria
Federation
 Pop  :3.75bn
 Food :7
 Ores :4
 Tec  :4
 Trade:3
 MODs :7
Players Move
Federation MODs
Land Based:46
SShips:1   :14
Territories:11
2
```



```
Grid Position
  x:147 y:88
Rebel Alliance
 SShips:2
  Food :6
  Ores :6
  Tec  :8
  Trade:10
  MODs :38
 Attack Mode
Orbit
Rebel Grp  MODs
Land Based:38
SShips:2   :38
Territories:12
5
```

```
1601 DEFine PROCedure SSInfo
1602 MODInfo:PTInfo:ch=11:INK#ch,6:CLS#ch
1603 IF Gp=1:Gp$='Rebel Alliance':ss=RSSn:INK#ch,5:INK#2,5
1604 IF Gp=2:Gp$='Federation      ':ss=FSSn :INK#ch,3:INK#2,3
1605 PRINT#ch,Gp$:PRINT#ch,' SShips:';ss :CIRCLE#2,30,21,3
1606 PRINT#ch,'  Food  :';side(Gp,1)
1607 PRINT#ch,'  Ores  :';side(Gp,2)
1608 PRINT#ch,'  Tec   :';side(Gp,3)
1609 PRINT#ch,'  Trade:';side(Gp,4)
1610 PRINT#ch,'  MODs:';side(Gp,5)
1611 AT#ch,7,0:PRINT#ch,Mes$:
1612 IF sch=0 AND rch=1:ch=12:AT#ch,7,0:PRINT#ch,' (R)ecruit '
1613 END DEFine
```

MODInfo

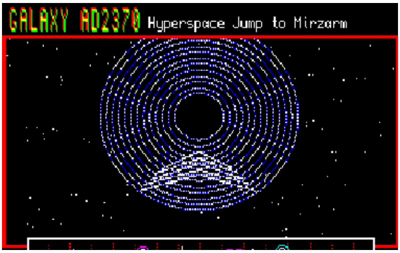
Note Lines 1603 -1605 **INK#2, CIRCLE** Star Ship Colours

INK 5 Rebel Alliance Star Ship side Colours

Dice Roll

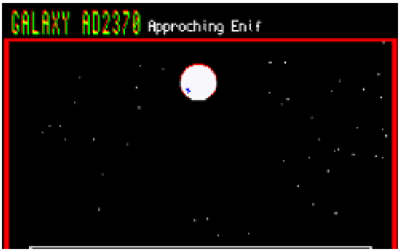
Hyperspace Jump, Warp Speed and Impulse

The average distance across a Galaxy is a 100,000 Light years. Hyperspace or Subspace travel although almost instantaneous needs accurate Star charts and calculated navigations to avoiding any collisions. Warp speeds take longer, days, weeks even months to reach a distant Star system. Travelling on Impulse (sub light speed) can still take several hours to cross a solar system. My premise at this point was to depict the journeys of space travel with a graphic sequence. This became a Hyperspace Jump to the selected Star, deceleration through Warp Speeds on approaching the Star and then a swing into Planetary Orbit on Impulse drive.



GALAXY AD2370 Hyperspace Jump to Mirzarm

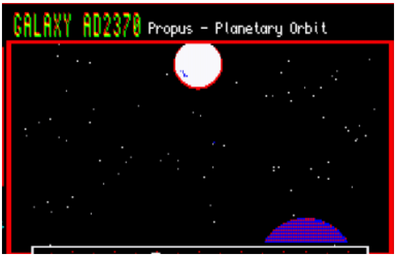
```
1615 DEFine PROCedure HJump
1616 PRINT#16,'Hyperspace Jump to ';astro$(sn)
1617 wx=92:wy=54:ws=12):CURSOR#2,6,110:PRINT#2,'
1618 FOR a=12 TO 24 STEP 3
1619    INK 241:CIRCLE wx,wy,a: Pause 5
1620    INK    7:LINE 50+a,10+a TO wx,18+a TO 132-a,10+a
1621 END FOR a
1622 GTime :PAUSE .5
1623 FOR a=24 to 48 STEP 3
1624    INK 241:CIRCLE wx,wy,a: PAUSE .5
1625 END FOR a
1626 GTime:PAUSE 3
1627 FOR a=1 TO 16
1628    INK 0:FILL 1:CIRCLE wx,wy,a*4:FILL 0:PAUSE .5
1629 END FOR a
1630 FOR a=12 TO 1 STEP -1
1631    b=24-a*2   :INK 241:CIRCLE wx,wy,a*4:ws=a
1632    PAUSE a/2:INK    0:CIRCLE wx,wy,a*4:GTime::StarMove
1633 END FOR a
1634 END DEFine

1636 DEFine PROCedure StarMove
1637 hoz=ws:ver=ws:INK#2,7:CURSOR#2,6,110:PRINT#2,'Warp Speed:'&ws&' '
1638 IF ws>0:INK#1,7:FOR w=1 TO ws*2:POINT#1,RND(40 TO 160),RND(30 TO 70)
1639 SCROLL#4,-ver*2 : SCROLL#9,ver*2 : PAN#6,-hoz*2 : PAN#7,hoz*2
1640 PAN#3,-hoz : SCROLL#3,-ver : PAN#5,  hoz : SCROLL#5,-ver
1641 PAN#8,-hoz : SCROLL#8, ver : PAN#10,hoz : SCROLL#10,ver
1642 END DEFine
```



GALAXY AD2370 Approching Enif

```
1644 DEFine PROCedure StarView
1645 sx=92:sy=80:PRINT#16,'Approching ';astro$(sn)
1646 FOR r=2 TO 24 STEP 2
1647    INK 0:FILL 1:CIRCLE sx,sy,r*.6:FILL 0
1648    StarMove:sr=r*.5:Star:PAUSE r/4
1649 END FOR r
1650 END DEFine

1652 DEFine PROCedure Star
1653  INK    2:FILL 1:CIRCLE sx,sy,sr*1  :FILL 0
1654  INK    7:FILL 1:CIRCLE sx,sy,sr*.95 :FILL 0
1655  INK 241:FILL 1:CIRCLE sx-sr*.6,sy-sr*.4,sr*.2,.3,PI/4:FILL 0
1656 END DEFine
```

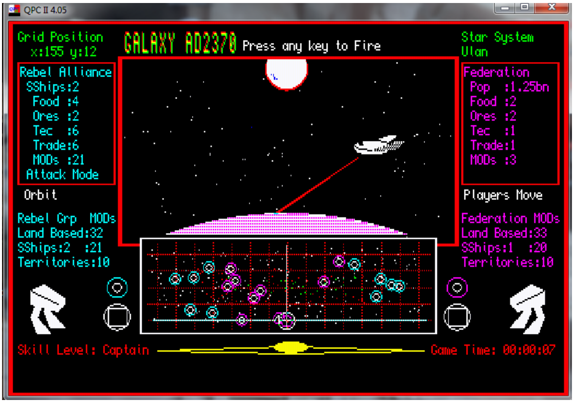
```
1658 DEFine PROCedure EOrbit
1659 PRINT#16,astro$(sn);' - Planetary Orbit'
1660 INK#2,7:CURSOR#2,6,110:PRINT#2,'Impulse      ':x=92:CURSOR 0,60
1661 FOR a=1 TO 18
1662    POINT RND(10 TO 180),RND(40 TO 80):SCROLL -1:GTime
1663    IF astro(sn,11)=1 :x=18+a*3
1664    IF astro(sn,11)=3 :x=182-(a*3)
1665    INK astro(sn,10)  :y=24-a*3
1666    FILL 1:CIRCLE x,y,a*2:FILL 0:PAUSE a/3
1667    INK 0:FILL 1:CIRCLE x,y,a*2:FILL 0
1668    INK 7:POINT RND(40 TO 160),RND(30 TO 40)
1669 END FOR a
1670 FOR b=1 TO 14
1671   INK 7:POINT RND(10 TO 180),RND(10 TO 20):GTime
1672   IF Gp=1:x=92+b*3:y=50:ELSE x=92-b*3:y=55
1673   INK astro(sn,10):FILL 1:CIRCLE 92,-200+b,198:FILL 0
1674   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
1675   INK     0:CIRCLE x,y,b-2:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b
1676 END FOR b
1677 ink1=7:ink2=248
1678 IF Gp=1:x=140:y=50:SS02:ELSE x=56:y=50:SS03
1679 INK#2,7:CURSOR#2,6,110:PRINT#2,'Orbit       '
1680 END DEFine
```



```
GALAXY AD2370 Propus - Planetary Orbit
```

Attack Mode

Upon arriving and Entering **Orbit** over a **Star Systems** Planet**,** if not already annexed, **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 Star Ship. If a **Victory** for the attacker, the Star System is annexed, calculations determined by the Planets Population and exiting credits results in **Food/Ores/Tec/Trade** and **MODs** being added to the Star Ship inventory.

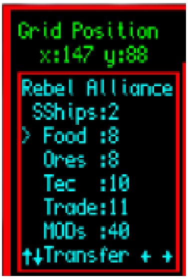


```
Grid Position        GALAXY AD2370 Press any key to Fire        Star System
  x:155 y:12                                                    Ulan
Rebel Alliance                                                  Federation
 SShips:2                                                        Pop  :1.25bn
  Food :4                                                        Food :2
  Ores :2                                                        Ores :2
  Tec  :6                                                        Tec  :1
  Trade:6                                                        Trade:1
  MODs :21                                                       MODs :3
 Attack Mode
 Orbit                                                          Players Move

Rebel Grp  MODs                                                 Federation MODs
Land Based:32                                                   Land Based:33
SShips:2   :21                                                  SShips:1   :20
Territories:10                                                  Territories:10

Skill Level: Captain                                            Game Time: 00:00:07
```

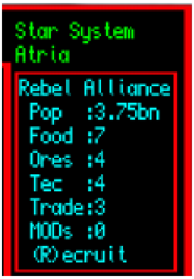
```
1682 DEFine PROCedure Phaser
1683 h=RND(80 TO 120):v=12:INK 2
1684 FOR a=1 TO 6
1685   OVER -1:FILL 1:LINE x,y TO h-1,v TO h+1,v TO x,y:FILL 0:OVER 0:PAUSE .5
1686 END FOR a
1687 END DEFine
```

```
1689 DEFine PROCedure Victory
1690 PRINT#16,'Transfer then - Recruit MODs'
1691 side(Gp,1)=side(Gp,1)+INT(astro(n,1)/3)  :REMark Food  (20)
1692 IF side(Gp,1)>20:side(Gp,1)=20
1693 side(Gp,2)=side(Gp,2)+INT(astro(n,2)/2)  :REMark Ores  (12)
1694 IF side(Gp,2)>12:side(Gp,2)=12
1695 side(Gp,3)=side(Gp,3)+INT(astro(n,3)/2)  :REMark Tec   (16)
1696 IF side(Gp,3)>16:side(Gp,3)=16
1697 side(Gp,4)=side(Gp,4)+INT(astro(n,4)/2)  :REMark Trade (24)
1698 IF side(Gp,4)>24:side(Gp,4)=24
1699 side(Gp,5)=side(Gp,5)+INT(astro(n,6)+2)  :REMark MODs (100)
1700 IF side(Gp,5)>100:side(Gp,5)=100
1701 astro(n,7)=Gp:LMap:SSInfo
1702 END DEFine
```

Victory → Transfer

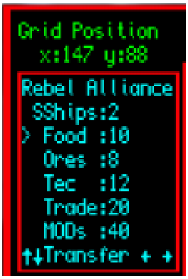


```
Grid Position
 x:147 y:88
Rebel Alliance
 SShips:2
> Food :8
  Ores :8
  Tec  :10
  Trade:11
  MODs :40
↑↓Transfer ← →
```

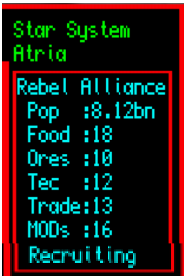


```
Star System
Atria
Rebel Alliance
 Pop  :3.75bn
 Food :7
 Ores :4
 Tec  :4
 Trade:3
 MODs :0
 (R)ecruit
```

Transfer ← Recruit



```
Grid Position
 x:147 y:88
Rebel Alliance
 SShips:2
> Food :10
  Ores :8
  Tec  :12
  Trade:20
  MODs :40
↑↓Transfer ← →
```



```
Star System
Atria
Rebel Alliance
 Pop  :8.12bn
 Food :18
 Ores :10
 Tec  :12
 Trade:13
 MODs :16
 Recruiting
```

Transfer & Recruit

If Planet is already annexed or just been taken the cursor keys,**<Left Right Up Down>** are used to select the **Transfer** of **Food/Ores/Tec/Trade** credits to the Planet. Pressing the **(R) ecruit** key, calculates the change in Population and generates credits and **MODs** in respect of new values.

```
1704 DEFine PROCedure TranSP
1705 REMark Transfer Food/Ores/Tec/Trade/MOD's from Star Ship > Planet
1706 IF t=1 AND side(Gp,1)>1 AND astro(n,1)<20
1707    side(Gp,1)=side(Gp,1)-1:astro(n,1)=astro(n,1)+1        :REMark Food  (max 20)
1708 END IF
1709 IF t=2 AND side(Gp,2)>1 AND astro(n,2)<10
1710    side(Gp,2)=side(Gp,2)-1:astro(n,2)=astro(n,2)+1        :REMark Ores  (max 10)
1711 END IF
1712 IF t=3 AND side(Gp,3)>1 AND astro(n,3)<12
1713    side(Gp,3)=side(Gp,3)-1:astro(n,3)=astro(n,3)+1        :REMark Tec   (max12)
1714 END IF
1715 IF t=4 AND side(Gp,4)>1 AND astro(n,4)<20
1716    side(Gp,4)=side(Gp,4)-1:astro(n,4)=astro(n,4)+1        :REMark Trade (max 20)
1717 END IF
1718 IF t=5 AND side(Gp,5)>1 AND astro(n,5)<20
1719    side(Gp,5)=side(Gp,5)-1:astro(n,5)=astro(n,5)+1        :REMark MODs  (max 20)
1720 END IF
1721 SSInfo
1722 END DEFine
```

```
1724 DEFine PROCedure Recruit
1725 REMark Trade Tec/Trade/Food/Ores Credits Increase Population & MODs
1726 ch=12:AT#ch,7,0:PRINT#ch,' Recruiting':PAUSE 20:AT#ch,6,0:CLS#ch,4
1727 astro(n,6)=astro(n,1)/3.2+astro(n,3)/3.2 :REMark Population
1728 IF astro(n,6)>10:astro(n,6)=10
1729 astro(n,1)=astro(n,1)+INT(astro(n,6)/2)     :REMark Food  (20)
1730 IF astro(n,1)>20:astro(n,1)=20
1731 astro(n,2)=astro(n,2)+INT(astro(n,3)/4)     :REMark Ores  (10)
1732 IF astro(n,2)>10:astro(n,2)=10
1733 astro(n,3)=astro(n,3)+INT(astro(n,6)/2)     :REMark Tec   (12)
1734 IF astro(n,3)>12:astro(n,3)=12
1735 astro(n,4)=astro(n,4)+INT(astro(n,1)/2)     :REMark Trade (20)
1736 IF astro(n,4)>20:astro(n,4)=20
1737 astro(n,5)=INT(astro(n,6)*2)                :REMark MODs  (20)
1738 PRINT#16,'To leave Orbit press':BLOCK#16,16,4,130,4,71739 SSInfo
1739 MODInfo:PTInfo
1740 END DEFine
```

At this point **Transfer** of **Food/Ores/Tec/Trade** and **MODs** is made between Planet and Star Ship. This is in part to strengthen forces for future Attacks and partly to improve annexed planets defences and make them less vulnerable to opponent attacks.

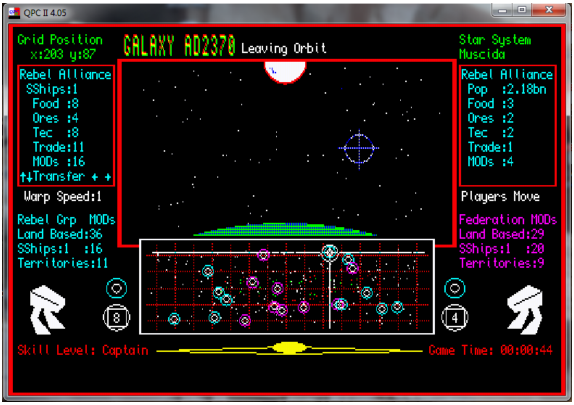
```
1742 DEFine PROCedure TranPS
1743 REMark Transfer Food/Ores/Tec/Trade/MOD's from Planet > Star Ship
1744 SShip=1+INT(side(Gp,5)/20)
1745 IF t=1 AND astro(n,1)>1 AND side(Gp,1)<10+2*SShip
1746    astro(n,1)=astro(n,1)-1:side(Gp,1)=side(Gp,1)+1       :REMark Food  (max 20)
1747    IF astro(n,1)<astro(n,6):astro(n,6)=astro(n,1)         :REMark Population
1748 END IF
1749 IF t=2 AND astro(n,2)>1 AND side(Gp,2)<2+2*SShip
1750    astro(n,2)=astro(n,2)-1:side(Gp,2)=side(Gp,2)+1       :REMark Ores  (max 12)
1751 END IF
1752 IF t=3 AND astro(n,3)>1 AND side(Gp,3)<6+2*SShip
1753    astro(n,3)=astro(n,3)-1:side(Gp,3)=side(Gp,3)+1       :REMark Tec   (max 16)
1754 END IF
1755 IF t=4 AND astro(n,4)>1 AND side(Gp,4)<14+2*SShip
1756    astro(n,4)=astro(n,4)-1:side(Gp,4)=side(Gp,4)+1       :REMark Trade (max 24)
1757 END IF
1758 IF t=5 AND astro(n,5)>1 AND side(Gp,5)<99
1759    astro(n,5)=astro(n,5)-1:side(Gp,5)=side(Gp,5)+1       :REMark MODs (max 100)
1760 END IF
1761 SSInfo
1762 END DEFine
```

Note In both **TransSP** & **TranPS** variable **t** provides the link to selection of credits and MODs being transferred. The variable **SShip** limits the transfers between Planet and Star ships. This is based loosely on the maximum number of Credits and MODs divided among the ships in the Players fleet.

Food 10+2***SShip** : **Ores** 2+2***SShip** : **Tec** 6+2***SShip** : **Trade** 14+2***SShip** : **MODs** 100 max

Transactions complete Leaving **Orbit** and returning to deep space is a short graphic sequence to clear the view screen and information panels ready for the next sortie.

```
1764 DEFine PROCedure LOrbit
1765 PRINT#16,'Leaving Orbit':
1766 ink1=0:ink2=0:IF Gp=1:x=140:y=50:SS02:ELSE x=56:y=50:SS03
1767 CURSOR 0,60:INK#2,7:CURSOR#2,6,110:PRINT#2,'Warp Speed:1'
1768 FOR b=15 TO 1 STEP -1
1769   GTime:IF Gp=1:x=140-42+(b*3):y=50:ELSE x=60+42-(b*3):y=55
1770   INK 241:CIRCLE x,y,b-4:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b: PAUSE b/2
1771   INK     0:CIRCLE x,y,b-4:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b
1772   INK 7:POINT RND(20 TO 180),RND(20 TO 70):SCROLL -2,1:PAN#6,-1
1773   INK 7:POINT RND(20 TO 180),RND(20 TO 70):SCROLL  2,2:PAN#7, 1
1774 END FOR b
1775 CLS#16
1776 END DEFine
```



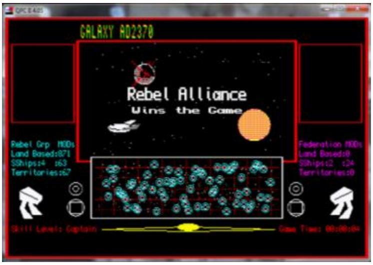
The Pro's and Con's of Conflict

As the Game progresses if a Star Ships MODs fall too low in number, then attacking Star Systems without a numerical advantage is aborted. It then becomes necessary to seek out annexed Star Systems to carry out Transfer and Recruit more Food/Ores/Tec/Trade credits and MODs. The Hazard Encounter gives the Interactive player the advantage of quickly gaining Tec credits and additional MODs, but it comes with the risk of just as easily losing them !!!

Computer Generated Interaction

Increasing or decreasing the period when the Computer is locked out of play gives the pretence of a variable Skill level. The Computers flow of decisions are reliant on preset conditions, time delays and to some extent random numbers. To guard against the worst outcomes, avoidance mechanisms are in place as part of the code. For example, if the number of Star Ship MODs are less than 20 the program will avoid Attacks and seek out already annexed Star Systems to Recruit more Star Ship Resources.

```
1778 REMark Part 3 End Game

1780 DEFine PROCedure GEnd
1781 wx=  92:wy=5:px=156:py=28:DeathStar
1782 wx=156:wy=28:dx=56:dy=72:CSIZE 2,1
1783 IF FPs=0
1784    DSDefeat:PAUSE 10:INK 7:OVER 1
1785    FOR i=1 TO 2:CURSOR 64+i,46:PRINT 'Rebel Alliance'
1786    OVER 0:ink1=7:ink2=248:x=34:y=24:SS02
1787 END IF
1788 IF RPs=0
1789    DSAttack:PAUSE 10:INK 7:OVER 1
1790    FOR i=1 TO 2:CURSOR 64+i,46:PRINT '  Federation  '
1791    OVER 0:ink1=7:ink2=248:x=40:y=24:SS03
1792 END IF
1793 CSIZE 2,0:INK 7:OVER 1
1794 FOR i=1 TO 2:CURSOR 70+i,68:PRINT#ch,'Wins the Game'
1795 OVER 0:CSIZE 0,0:INK 4
1796 PAUSE 100:LName:PAUSE 200:GMenu
1797 END DEFine
```



```
1799 DEFine PROCedure DeathStar
1800 ws=1:STRIP#2,0:CURSOR#2,0,110:PRINT#2,'            '
1801 FOR a=1 TO 18
1802   INK astro(n,10):wx=80+a*4:wy=-46+a*4
1803   FILL 1:CIRCLE wx,wy,50-a*2:FILL 0
1804   INK 248:FILL 1:CIRCLE 20+a*2,90-a,a/3:FILL 0:PAUSE 2
1805   INK    0:FILL 1:CIRCLE 20+a*2,90-a,a/3:FILL 0
1806   FILL 1:CIRCLE wx,wy,50-a*2:FILL 0
1807 END FOR a
1808 INK astro(n,10)
1809 FILL 1:CIRCLE wx,wy,14:FILL 0
1810 ink1=7:ink2=248:x=56:y=72:SS01
1811 END DEFine
```

```
1813 DEFine PROCedure DSAttack
1814 FOR a=1 TO 6
1815   INK 2:FILL 1
1816   LINE dx+2,dy+4 TO wx-10,wy+6 TO wx-12,wy+4 TO dx+2,dy+4
1817   FILL 0:PAUSE 3:INK 0:FILL 1
1818   LINE dx+2,dy+4 TO wx-10,wy+6 TO wx-12,wy+4 TO dx+2,dy+4
1819   FILL 0:PAUSE 2
1820 END FOR a
1821 INK 7:FILL 1:CIRCLE dx+2,dy+4,4:FILL 0
1822 FOR b=1 TO 12:INK 0 :CIRCLE 140+b,32,b:PAUSE .5
1823 FOR c=1 TO 12:FILL 1:CIRCLE 150+c*2,28+RND(-4 TO 8),RND(2 TO 5):FILL 0
1824 END DEFine
```

```
1826 DEFine PROCedure DSDefeat
1827 FOR i=1 TO 18
1828   INK 2:x1=RND(-12 TO 12):y1=RND(-12 TO 12):LINE dx,dy TO dx+x1,dy+y1
1829   INK 0:FILL 1:CIRCLE dx+RND(-2 TO 6),dy+RND(-2 TO 6),2:FILL 0:PAUSE 2
1830 END FOR i
1831 END DEFine
```

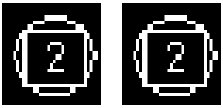
```
1833 REMark Galaxy League Table

1835 DEFine PROCedure LTDefault
1836 REMark Score League Table Default
1837 name$(1)='SPlay    ':score(1,1)=5072:score(1,2)=1024:score(1,3)=58
1838 name$(2)='CPlay    ':score(2,1)=3765:score(2,2)=1185:score(2,3)=62
1839 name$(3)='QBITS    ':score(3,1)=2960:score(3,2)=1296:score(3,3)=66
1840 LSave
1841 END DEFine

1843 DEFine PROCedure LTReset
1844 REMark Resets League Table Reset
1845 name$(1)='        ':score(1,1)=0:score(1,2)=0:score(1,3)=0
1846 name$(2)='        ':score(2,1)=0:score(2,2)=0:score(2,3)=0
1847 name$(3)='        ':score(3,1)=0:score(3,2)=0:score(3,3)=0
1848 LSave
1849 END DEFine

1851 DEFine PROCedure LScore
1852 ch=13:CLS#ch:OVER#ch,1:CSIZE#ch,2,0:INK#ch,7
1853 FOR i=1 TO 2:CURSOR#ch,60+i,2:PRINT#ch,'League Table'
1854 OVER#ch,0:CSIZE#ch,0,0:INK#ch,5
1855 CURSOR#ch,6,16:PRINT#ch,'Skill  Time h:m:s MODs Planets Gamer'
1856 CURSOR#ch,6,28:PRINT#ch,'Marshal'\' Admiral'\' Captain'
1857 FOR a=1 TO 3
1858  HST$=DATE$(score(a,1)):HS2=score(a,2):HS3=score(a,3)
1859  INK#ch,3:CURSOR#ch,60,18+a*10:PRINT#ch,HST$(13 TO 20);'  ';HS2
1860  CURSOR#ch,168,18+a*10:PRINT#ch,HS3,:INK#ch,6:PRINT#ch,name$(a)
1861 END FOR a
1862 END DEFine

1864 DEFine PROCedure LName
1865 GTS=DATE-Gclk+GTS:Gclk$=DATE$(GTS)
1866 IF nch=1:nch=0: LScore:RETurn              :REMark F2 Gametest
1867 IF score(GSk,1)=0:GO TO 1869               :REMark League Table Reset
1868 IF score(GSk,1)<GTS: LScore:RETurn         :REMark LT GTS Check < new GTS
1869 IF Gp=1:HS2=RMOD+side(1,5):HS3=Rt:ELSE HS2=FMOD+side(2,5):HS3=Ft
1870 CURSOR#2,444,216:PRINT#2,'       '
1871 score(GSk,1)=GTS:score(GSk,2)=HS2:score(GSk,3)=HS3:LScore
1872 ch=20:OPEN#ch,con_10x10a0x0_10:WINDOW#ch,60,10,314,167+GSk*10
1873 PAPER#ch,0:CLS#ch:INK#ch,6:INPUT#ch,name$(GSk):CLOSE#ch: LSave
1874 END DEFine

1876 DEFine PROCedure LSave
1877 DELETE Dv$&'QBAD2370DLT':OPEN_NEW#99,Dv$&'QBAD2370DLT'
1878 FOR a=1 TO 3:PRINT#99,name$(a)\score(a,1)\score(a,2)\score(a,3)
1879 CLOSE#99
1880 END DEFine

1882 DEFine PROCedure LLoad
1883 OPEN_IN#99,Dv$&'QBAD2370DLT'
1884 FOR a=1 TO 3:INPUT#99,name$(a)\score(a,1)\score(a,2)\score(a,3)
1885 CLOSE#99
1886 END DEFine
```

```
1888 REMark Game Notes / DiceRoll

1890 DEFine PROCedure GNotes
1891 RESTORE 1063:INK 7:CURSOR 92,6:PRINT 'Galaxy AD2370 Notes':INK 4
1892 FOR a=2 TO 6
1893   READ i,c,r,s,str$:c=6:r=a*10:MPrn:IF a=2:BLOCK 16,4,166,24,5
1894 END FOR a
1895 INK 3:CURSOR 26, 74:PRINT '(F1) Simulation Mode (F2) Graphics Check'
1896 INK 3:CURSOR 26, 84:PRINT '(F3) Timeslip Check - swaps Sides Assets'
1897 INK 3:CURSOR 26, 94:PRINT '(F4) Enemy Encounter (F5) Tec/MODs Cheat'
1898 INK 3:CURSOR 90,108:PRINT '(G)ame Notes (M)enu'
1899 INK 7:OVER 1:CURSOR 96,108:PRINT 'G          M':OVER 0
1900 PAUSE:CLS:FOR i=1 TO 120:POINT RND(10 TO 200),RND(5 TO 95)
1901 END DEFine
```

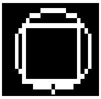
QBITS Galaxy AD2370 - Roll of the Dice

The Roll of the Dice decides which side losses MODs. The Attacking side losses to lower or even numbers rolled by the defending Star System. For an **Interactive** Player, if their Star Ship has less than four MODs the Attack is aborted.

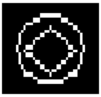
Variable (**dn**) holds the current Star System, (**SPch**) controls the action:- a PAUSE for the **Interactive** player to press any key and Roll the Dice. For the **Simulator** or **Opponent** move this action is instant. (**Gp**) holds the selected side [**Rebel Alliance** or **Federation**] for updating sides **MOD** results



```
1903 DEFine PROCedure RollDice
1904 ch=2:GTime:dn=n
1905 IF SPch=0 AND Gp=Gp1:PRINT#16,'Press any key to Fire' :PAUSE
1906 IF Gp=1:x=132:y=44:Phaser:ELSE x=70:y=48:Phaser
1907 FOR a=1 TO RND(4 TO 6)
1908   dx=  30:DiceA:PAUSE 1:DiceB
1909   dx=132:DiceA:PAUSE 1:DiceB
1910 END FOR a
1911 dn1=RND(1 TO 8):dx=  30:DiceA:CURSOR#ch,dx,dy,-3,-4:PRINT#ch,dn1
1912 dn2=RND(1 TO 8):dx=132:DiceA:CURSOR#ch,dx,dy,-3,-4:PRINT#ch,dn2
1913 IF dn1>dn2:astro(dn,5)=astro(dn,5) -DMs
1914 IF dn1=dn2: side(Gp,5)=side(Gp,5) -DMs
1915 IF dn2>dn1: side(Gp,5)=side(Gp,5) -DMs
1916 IF astro(dn,5)<0:astro(dn,5)=0
1917 END DEFine
```



```
1919 DEFine PROCedure DiceA
1920 ch=2:dy=12:l=2.8:INK#ch,0:FILL#ch,1:CIRCLE#ch,dx,dy,4:FILL#ch,0
1921 INK#ch,7:CIRCLE#ch,dx,dy,3.9
1922 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
1923 END DEFine
```



```
1925 DEFine PROCedure DiceB
1926 ch=2:dy=12:l=2.8:INK#ch,0:FILL#ch,1:CIRCLE#ch,dx,dy,4:FILL#ch,0
1927 INK#ch,7:CIRCLE#ch,dx,dy,3.9
1928 LINE#ch,dx,dy+l TO dx+l,dy TO dx,dy-l TO dx-l,dy TO dx,dy+l
1929 END DEFine
```

QBITS Galaxy AD2370 Star Ships & MODS

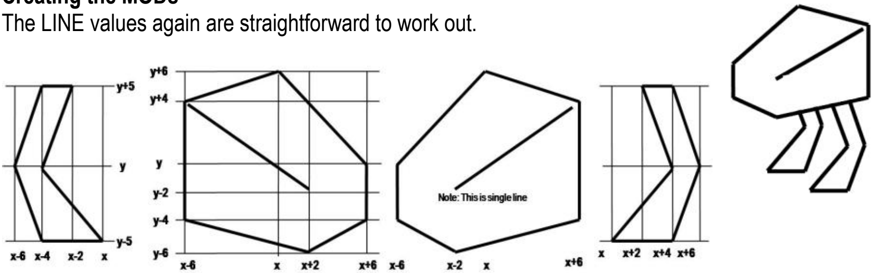
A number of ideas to incorporate Star Ship images was part of my agenda. Creating these can be done in several ways, with a pixel by pixel grid, generated from a scanned image or simply by use of SuperBASIC commands ARC, CIRCLE, LINE, POINT. That is not forgetting the accompanying, INK, FILL, OVER, PI,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

Creating Simple Images

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 shape, I then drew it over a grid. As a child I remember being busy for hours linking up sets of dots to create a picture. With a little imagination employing this technique an object can be broken down into a number of arcs, circles and lines. Then the shape can be build up from overlapping layers.

Creating the MODs

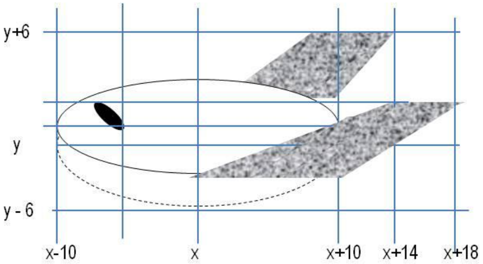
The LINE values again are straightforward to work out.



```
1931 DEFine PROCedure MODL
1932 ch=2:INK#ch,7:x=14:y=14
1933 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
1934 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
1935 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
1936 END DEFine
```

```
1938 DEFine PROCedure MODR
1939 ch=2:INK#ch,7:x=148:y =14
1940 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
1941 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
1942 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
1943 END DEFine
```

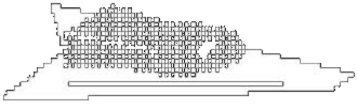
The other images can be generated in much the same way: -



This has overlapping shapes, a filled far side rectangular wing, the main body is a filled white circle, with a black ink circle drawn over the top to give an outline, the near side wing and another filled circle for the flight deck portal.

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

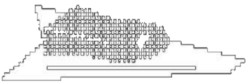
Being able to set the ink colours deployed by an image Procedure, allows it to be redraw with all colours set to the background. In effect, the image can be made to appear and disappear.



```
1945 DEFine PROCedure SS01
1946 ch=1:REMark DeathStar  x=56:y=76
1947 INK#ch,ink2:FILL#ch,1:CIRCLE#ch,x,y,9:FILL#ch,0
1948 INK#ch,ink1:CIRCLE#ch,x,y,9:INK#ch,0:ARC x-12,y TO x+12,y,PI/3
1949 INK#ch,ink1:FILL#ch,1:CIRCLE#ch,x+2,y+4,4:FILL#ch,0
1950 END DEFine
```

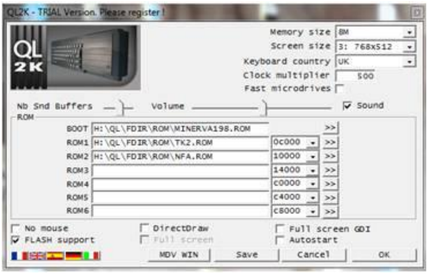
```
1952 DEFine PROCedure SS02
1953 ch=1:ink3=0:REMark Rebel SShip x=120:y=40
1954 INK#ch,ink1:FILL#ch,1:LINE#ch,x-6,y TO x+10,y+6 TO x+14,y+6 TO x+6,y TO x-8,y:FILL#ch,0
1955 INK#ch,ink1:FILL#ch,1:CIRCLE#ch,x,y,10,.4,PI/2:FILL#ch,0
1956 INK#ch,ink3:CIRCLE#ch,x,y+1,10,.28,PI/2:INK#ch,ink2
1957 FILL#ch,1:LINE#ch,x,y-2 TO x+14,y+3 TO x+18,y+3 TO x+14,y-1 TO x,y-2:FILL#ch,0
1958 INK#ch,0:FILL#ch,1:CIRCLE#ch,x-6,y+2,2,.3,PI/3:FILL#ch,0
1959 END DEFine
```

```
1961 DEFine PROCedure SS03
1962 ch=1:REMark Federation SShip x=50:y=40
1963 INK#ch,ink1:FILL#ch,1:LINE#ch,x-20,y-2 TO x-12,y+4 TO x-14,y+10 TO x+25,y+1 TO x+22,y-1 TO x-20,y-2:FILL#ch,0
1964 INK#ch,ink2:FILL#ch,1:LINE#ch,x-12,y+6 TO x-6,y+10 TO x+4,y+6 TO x,y+2 TO x-12,y+2 TO x-12,y+6:FILL#ch,0
1965 FILL#ch,1:CIRCLE#ch,x+2,y+4,9,.35,PI/2:FILL#ch,0:INK#ch,0
1966 FILL#ch,1:CIRCLE#ch,x+6,y+4,.7,3,PI/4:FILL#ch,0
1967 LINE#ch,x-12,y+6 TO x,y+4:LINE#ch,x+16,y-.5 TO x-12,y-.5
1968 END DEFine
```



QBITS use of QL Emulators

My mainstay in writing SuperBASIC Progs since 2008 has been the **QL2K** but recently I tried out the **QPC2** and **SMQL** emulators. They have some differences but both present the opportunity to use higher resolutions and an expanded colour range.

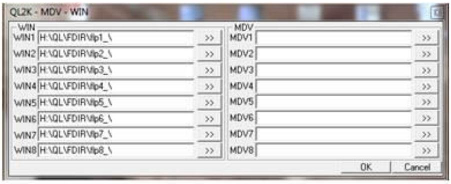


QL2K - TRIAL Version. Please register !

Memory size: 8M
Screen size: 3: 768x512
Keyboard country: UK
Clock multiplier: 500
□ Fast microdrives
☑ Sound

ROM
BOOT: H:\QL\FDIR\ROM\MINERVA198.ROM
ROM1: H:\QL\FDIR\ROM\TK2.ROM 0C000
ROM2: H:\QL\FDIR\ROM\NFA.ROM 10000
ROM3: 14000
ROM4: C0000
ROM5: C4000
ROM6: C8000

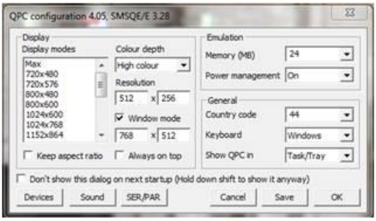
□ No mouse
☑ FLASH support
□ DirectDraw
□ Full screen
□ Full screen GDI
□ Autostart



QL2K - MDV - WIN

WIN
WIN1: H:\QL\FDIR\flp1_\
WIN2: H:\QL\FDIR\flp2_\
WIN3: H:\QL\FDIR\flp3_\
WIN4: H:\QL\FDIR\flp4_\
WIN5: H:\QL\FDIR\flp5_\
WIN6: H:\QL\FDIR\flp6_\
WIN7: H:\QL\FDIR\flp7_\
WIN8: H:\QL\FDIR\flp8_\

Advantage of the Clock Multiplier allows test runs at original QL speed.



QPC configuration 4.05, SMSQE/E 3.28

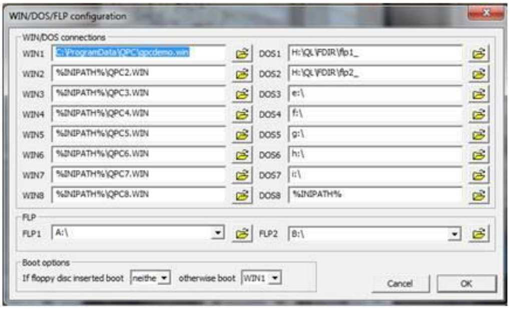
Display
Display modes: Max, 720x480, 720x576, 800x480, 800x600, 1024x600, 1024x768, 1152x864
Colour depth: High colour
Resolution: 512 x 256
☑ Window mode
768 x 512
□ Keep aspect ratio
□ Always on top

Emulation
Memory (MB): 24
Power management: On

General
Country code: 44
Keyboard: Windows
Show QPC in: Task/Tray

□ Don't show this dialog on next startup (Hold down shift to show it anyway)

DOS A: & B: Direct Floppy access

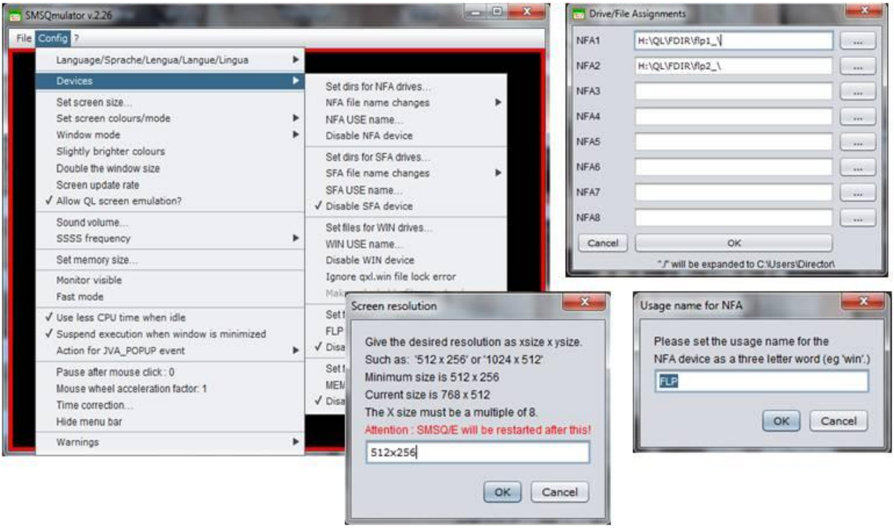


WIN/DOS/FLP configuration

WIN/DOS connections
WIN1: C:\ProgramData\QPC\qpcdemo.win
WIN2: %INIPATH%\QPC2.WIN
WIN3: %INIPATH%\QPC3.WIN
WIN4: %INIPATH%\QPC4.WIN
WIN5: %INIPATH%\QPC5.WIN
WIN6: %INIPATH%\QPC6.WIN
WIN7: %INIPATH%\QPC7.WIN
WIN8: %INIPATH%\QPC8.WIN
DOS1: H:\QL\FDIR\flp1_
DOS2: H:\QL\FDIR\flp2_
DOS3: e:\
DOS4: f:\
DOS5: g:\
DOS6: h:\
DOS7: i:\
DOS8: %INIPATH%

FLP
FLP1: A:\
FLP2: B:\

Boot options
If floppy disc inserted boot: neithe
otherwise boot: WIN1



SMSQmulator v.2.26

Drive/File Assignments

NFA1: H:\QL\FDIR\flp1_\
NFA2: H:\QL\FDIR\flp2_\
NFA3:
NFA4:
NFA5:
NFA6:
NFA7:
NFA8:

"./" will be expanded to C:\Users\Director\

Screen resolution

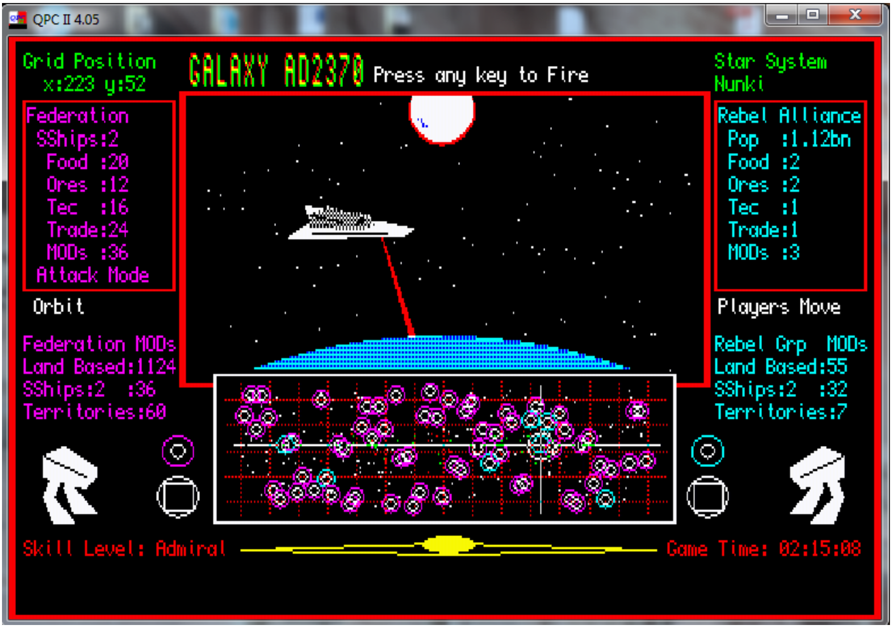
Give the desired resolution as xsize x ysize.
Such as: '512 x 256' or '1024 x 512'
Minimum size is 512 x 256
Current size is 768 x 512
The X size must be a multiple of 8.
Attention : SMSQ/E will be restarted after this!

512x256

Usage name for NFA

Please set the usage name for the
NFA device as a three letter word (eg 'win'.)

FLP



Grid Position
x:223 y:52

GALAXY AD2370 Press any key to Fire

Star System
Nunki

Federation
SShips:2
Food :20
Ores :12
Tec :16
Trade:24
MODs :36
Attack Mode

Orbit

Rebel Alliance
Pop :1.12bn
Food :2
Ores :2
Tec :1
Trade:1
MODs :3

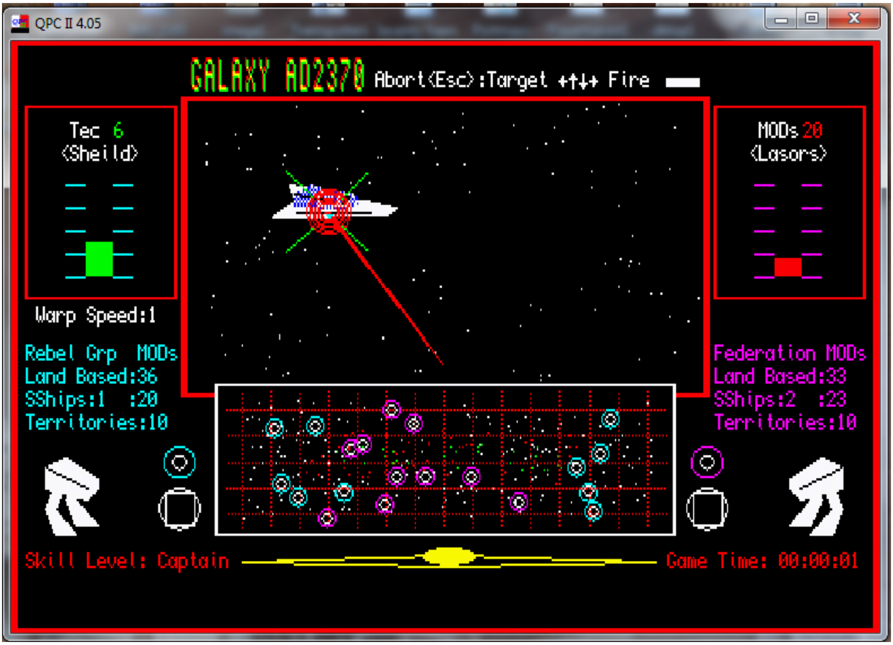
Players Move

Federation MODs
Land Based:1124
SShips:2 :36
Territories:60

Rebel Grp MODs
Land Based:55
SShips:2 :32
Territories:7

Skill Level: Admiral

Game Time: 02:15:08



GALAXY AD2370 Abort(Esc):Target ←↑↓→ Fire ▬

Tec 6
(Sheild)

MODs 20
(Lasors)

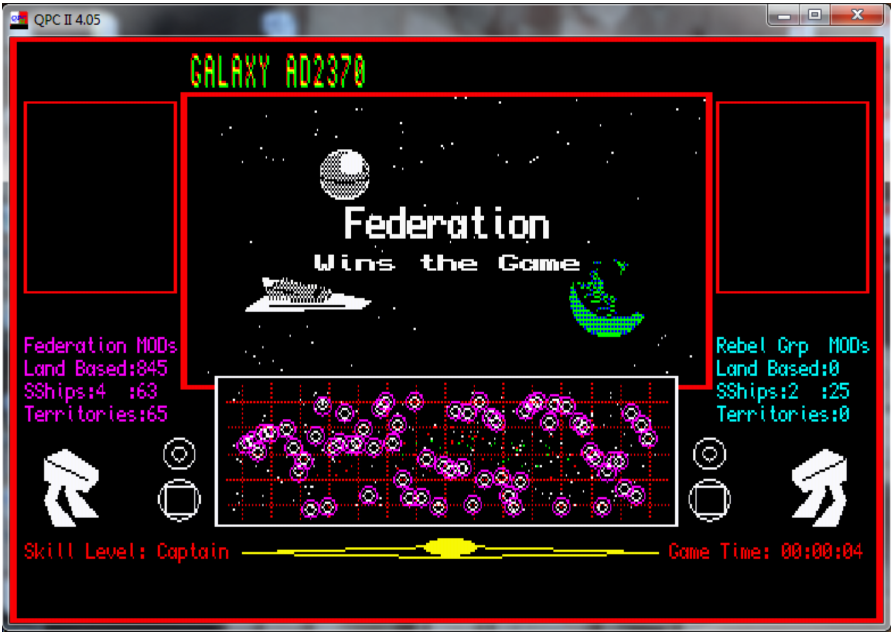
Warp Speed:1

Rebel Grp MODs
Land Based:36
SShips:1 :20
Territories:10

Federation MODs
Land Based:33
SShips:2 :23
Territories:10

Skill Level: Captain

Game Time: 00:00:01



GALAXY AD2370

Federation

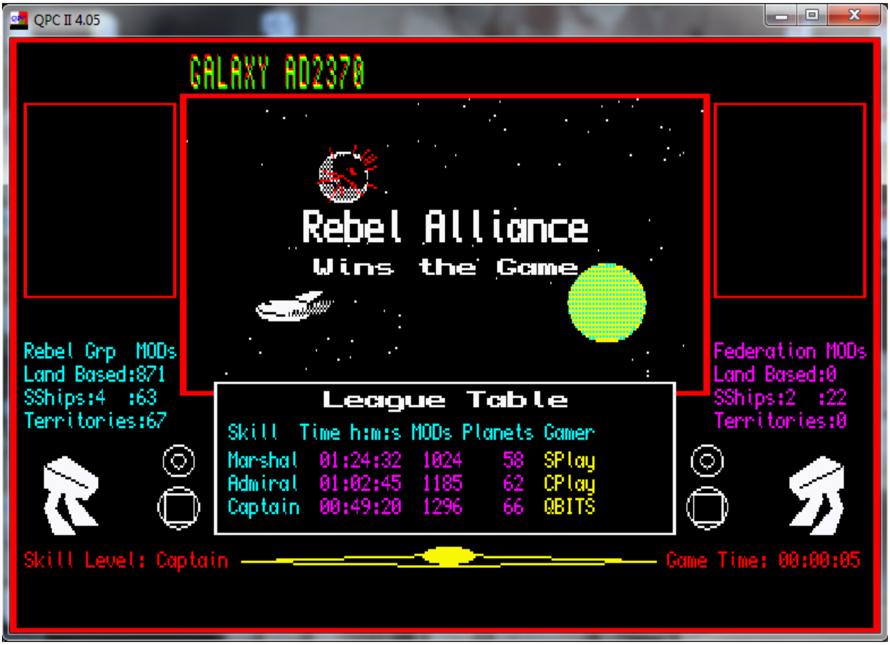
Wins the Game

Federation MODs
Land Based:845
SShips:4 :63
Territories:65

Rebel Grp MODs
Land Based:0
SShips:2 :25
Territories:0

Skill Level: Captain

Game Time: 00:00:04



GALAXY AD2370

Rebel Alliance

Wins the Game

Rebel Grp MODs
Land Based:871
SShips:4 :63
Territories:67

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

League Table

Skill	Time h:m:s	MODs	Planets	Gamer
Marshal	01:24:32	1024	58	SPlay
Admiral	01:02:45	1185	62	CPlay
Captain	00:49:20	1296	66	QBITS

Skill Level: Captain

Game Time: 00:00:05