**Space Game**

**A 3D Space Battle Simulation for the TRS-80 Model 100**

**By**

**Lloyd Johnson**

## SECTION 1 - INTRODUCTION

### Introduction

In 1978, I was working in Minneapolis for a company that made test equipment for telephone companies. The test organization I was in had a DEC LSI-11 computer. They had acquired this computer with the idea of using it to control test equipment over a communication bus. This never happened. The computer found other uses. As one of the users, I used the RUNOFF word processor program to write procedures for testing and calibrating the manufactured test equipment. This computer was configured with an auto answer modem connected to a phone line. Our group had a Lear Siegler ADM-3 dumb terminal and a 300 baud acoustical coupled modem that we were allowed to take home and use to connect to the LSI-11 computer at work. I took advantage of that during much of the time I worked for them.

It was using that terminal, that I developed a preliminary version of the game that I'm sharing now. I used RUNOFF to write an article about a simplified version of the game. I called the simplified game, Space Maze. The article was submitted to *Creative Computing* and was published in their January 1979 issue. It has been archived and is now available online at https://archive.org/details/creativecomputing-1979-01. Within this article I state, "I also have another version of this program in which the sentries are enemy space ships that are programmed to move toward you and attack." That is exactly what happens in this version.

Sometime after moving to St Louis in 1979, I acquired a TRS-80 Model 1 with 16K of memory. I rehosted the game to the TRS-80 and named it, Cosmic Conflict (unaware there was already an arcade game available with the same name). I self- marketed the game by selling it to computer stores as a cassette tape and a manual. I sold maybe 50 or so copies.

The Cosmic Conflict file folders I've been dragging around for about 45 years include a marked-up tractor feed fanfold computer listing, a photocopy of the manual and lots of programming notes - some handwritten, and some that were typed on a mechanical typewriter we used to own. There have been a few other variants of this game that I've worked on after the TRS-80, but unfortunately, I had lent the folder for one of the later versions to a coworker in Seattle. I didn't have that folder when I left Seattle in 1996. I've since contacted that coworker and he claims he doesn't have it either. Guess it was lost in one of our many moves. Oh well.

Space Game for the TRS-80 Model 100 and NEC PC-8201A is a rewrite of the TRS-80 model 1 game I wrote called Cosmic Conflict. Instead of calling it Cosmic Conflict, I changed its name to Space Game. Since it fit in a 16K TRS-80 Model 1, I did not think there would be a problem making it fit in a TRS-80 Model 100. However, this did turn into a challenge since I was very liberal with comments. I soon discovered that both the .DO and .BA versions would not fit on the Model 100 at the same time. I had to remove comments. The .DO versions that are provided do not have comments except for the first two lines and they load into BASIC with no

problems. The version that still has comments is provided as a .txt file. This runs well on the CloudT emulator.

Space Game commands were organized into menus and the output was formatted to fit 8 lines with 40 columns. The data was presented in two columns for the first six lines and the last two lines were reserved for status and menu selections. Some features were modified, some were left out and some other features were added. The objective was changed from destroying a planet to neutralizing an artifact. The reason for neutralizing the artifact is explained in the manual. The manual also explains the ship operation, and the abbreviations and spherical coordinate display data that is used to play the game.

The manual in the next section includes several screenshots with annotations provided by Elias S. Dennis. Dennis is a fictional character I created with a name that is shared with my wife's great-great-great-grandfather, Elias S. Dennis, Jr who also happened to be a union general under Ulysses S. Grant during the American civil war.

## SECTION 2 – GAME MANUAL

### Prelude

First mate and science officer Elias S. Dennis of the Starship Altera had a strange hobby. He enjoyed writing computer programs for a replica of an ancient laptop computer despite its primitive display and keyboard. The replica was old – perhaps 100 years. It was a working replica of a device from several centuries earlier. Still, it worked and as a means of passing away the long hours during warp jumps, Elias created a client/server relationship between this laptop and Altera's main computers. Although it was only meant as a recreational activity with no official application, this laptop became Altera's only fighting chance at survival when a major malfunction occurred as Altera entered the theater of operations. The malfunction resulted in irreparable damage to the data link between the ship's displays and controls and the ship's computer. Fortunately, Elias Dennis was able to reroute critical ship's status to his laptop and use the laptop to send critical commands to Altera's main computer.

The following documentation was provided by first mate and science officer, Elias S. Dennis:

### Intelligence Report and Background

An alien artifact in interstellar space has been located. The galactic coordinates recorded were 49.98 degrees right ascension and -43.07 degrees declination at approximately 85 lightyears from Sol. This artifact has been determined to be a grave threat to mankind. Reconnaissance spacecraft located this object on the other side of Eridani. The gravitational signature of the artifact defies many of the conventional laws of physics. Based on recovered data, it was concluded that a shielded form of dark energy was being concealed by this artifact. If this artifact is propelled into a star, the star will rapidly use most of its nuclear fuel and collapse into a neutron star. Although a Berserker[1] origin for this artifact was suspected, it was determined by other data that this artifact predates Berserkers.

Berserkers are self-replicating robotic spaceships, some the size of planets. They were created as the ultimate weapon of war between two long extinct civilizations. These death machines are guided by extremely intelligent self-aware computers, programmed to seek and destroy all life throughout the galaxy. The Berserkers know about this artifact and have stationed nine robotic sentries in the vicinity of the artifact. It is imperative that the nine sentries are destroyed, and the artifact is neutralized before a Berserker returns to use it as a weapon against mankind.

Starship Altera was dispatched to the vicinity of the artifact with orders to neutralize the artifact. A special weapons package was installed on Altera. This weapons package is activated by the neutralize artifact command. The neutralize artifact command will engage a sequence that will direct a tight beam of composite subatomic particles (exotic baryons) traveling at ultra relativistic velocities towards the artifact. The ship must have a velocity magnitude that is near

[1] Berserkers™ were first documented by Fred Saberhagen in numerous short stories and novels. (See https://www.berserker.com/JSS_Literary.html) References to Berserkers™ are made with permission from Joan Spicci Saberhagen, JSS Literary Productions, Managing the Literary Estate of Fred Saberhagen.

zero and a range of 250 Mm or less for this sequence to engage. The neutralize artifact command will require five minutes of mission time. During this command, no other commanding is possible. Successful use of this command will result in the creation of a wormhole into intergalactic space where the dark energy from the artifact will be safely dissipated.

**Coordinate System**

A brief refresher on the space coordinate system is appropriate. The orientation of the X,Y, and Z axis is the rotating galactic standard where the Z axis is parallel to the axis of galaxy rotation and the X axis is the vector from Sol to the galactic core. The Y axis is introduced to complete the right-hand coordinate system. Although described as a rotating frame, the rotation is so slow that for navigation purposes it may be considered nonrotating. The origin can be defined as anywhere but for the purposes of this mission, it will always be the location of Altera. This means that as Altera moves, the origin moves.

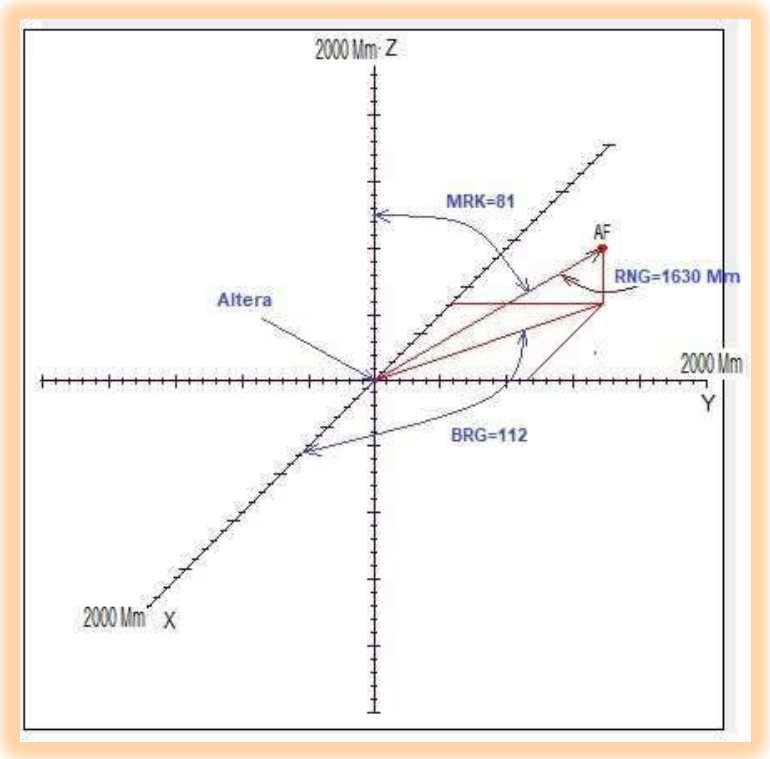


**Figure 1: Three-dimensional spherical coordinate system.**

The position of enemy ships is reported in spherical coordinates. The three components for spherical coordinates are range, bearing and mark. Range (RNG) is the distance from Altera to an object - be it either an enemy ship or the artifact. Range has units of Mm (Megameters). Bearing and mark are the angles phi and theta from conventional spherical coordinate definitions. Bearing (BRG) is measured on the XY plane and is the angle from the positive X axis to the object's projection on the XY plane. Bearing can have a value from 0 to 360 degrees. Mark (MRK) is the angle from the positive Z axis to a line from the origin to the object. It can have a value from 0 to 180 degrees.

The following screen shot is a possible depiction of the main menu display upon entering the system near the artifact. The status of the enemy ships is in the columns labeled L and E for lasers and engines. It will have values of Active (A), Disabled (D) or Destroyed (X). When both the enemy ships' engines and lasers are destroyed, the enemy ship is destroyed.

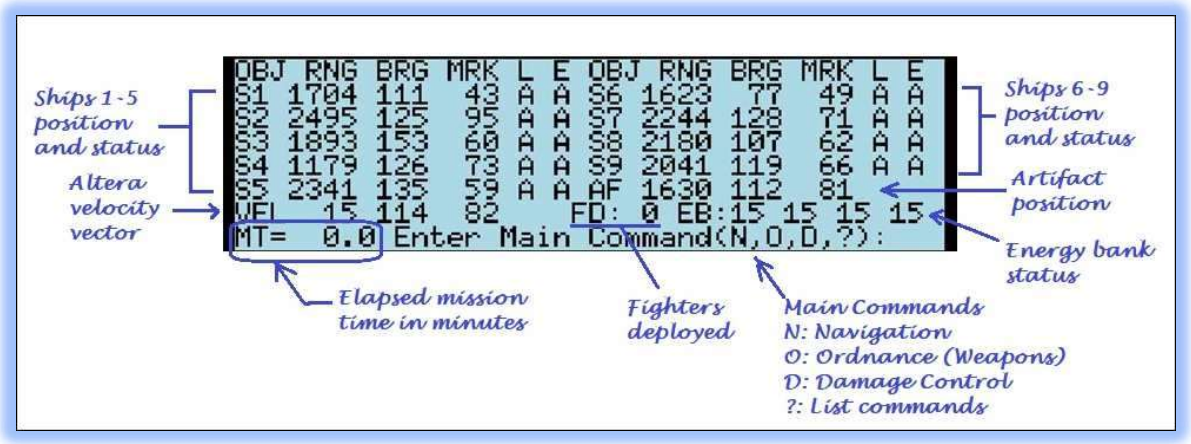


**Figure 2: Main Menu Display** *(from the notebook of Elias S. Dennis)*

Approximately 95% of the mass of the Starship Altera is composed of inert propellant. The inert propellant is used by magnetic confinement plasma impulse engines for performing in-system maneuvers. The propellant is heated by matter/anti-matter reactions into a plasma state that is magnetically confined, guided and expelled at near light speed velocities. The Altera mass ratio (before/after maneuver mass) resulting from a maximum velocity change is only slightly above 1. Due to the high specific impulse of the expelled plasma, there is sufficient propellant for the entire mission provided velocity magnitudes are kept reasonable (under 999 Mm/min). The maximum velocity change is limited to 100 Mm/min due to the capabilities of the engines. There may be further limitations on the magnitude of the velocity change if the energy banks assigned to the engines have been depleted or Altera fighters are deployed.

## Navigation

The navigation menu is available by entering the letter, "N" from Altera's main menu. The first six lines from the main menu display will remain the same but the bottom two lines will be updated with navigation status and a request for navigation commands. This is illustrated in the following screenshot.

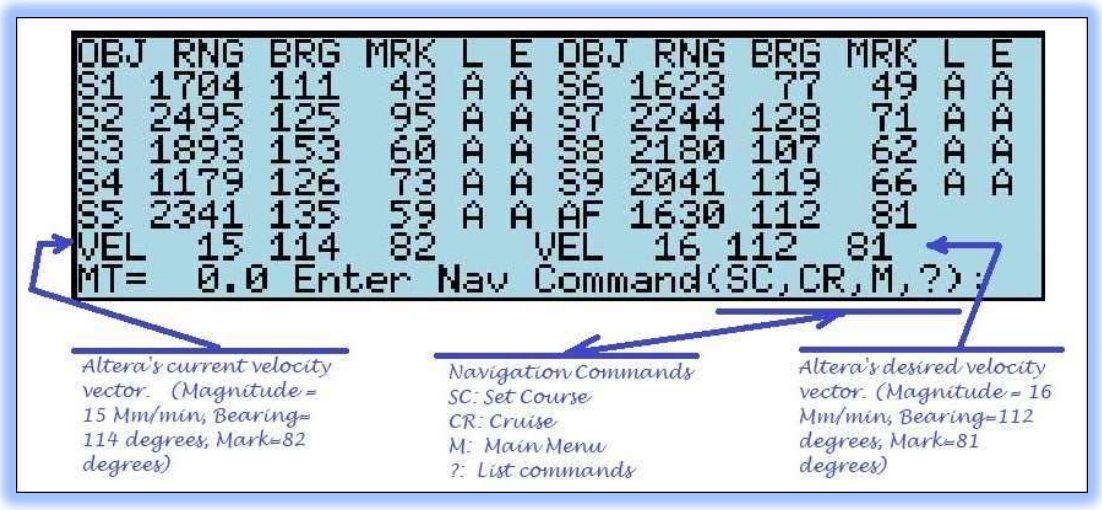


**Figure 3. Navigation Menu Display** *(from the notebook of Elias S. Dennis)*

When the Set Course (SC) command is entered, prompts will be provided for entering velocity (0-999), bearing (0-360) and mark (0-180) course setting parameters. Only numbers within the indicated range, the enter key or the letter, "X" is accepted for input. The letter, "X" will cancel the course setting. As an example, a course setting of 120, 140, and 80 was entered for velocity, bearing and mark respectively. The tactical rationale for this course is it will close the range with enemy ship 4 more quickly than the next two closest enemy ships (S1 and S6). This should allow enough time for enemy ship S4 to be destroyed before engaging enemy ships S1 and S6. It is not a good idea to fight too many ships at the same time. After the course setting is successfully entered, the message, "Course setting initiated…" is displayed followed by a delay of approximately 13 solar seconds as 0.5 minutes of mission time elapses. The number of solar seconds required for each 0.5 minutes of mission time will decrease as enemy ships are destroyed.[2] The display is updated as shown below.

[2] Elias S. Dennis did not provide any scientific explanation for this phenomenon.

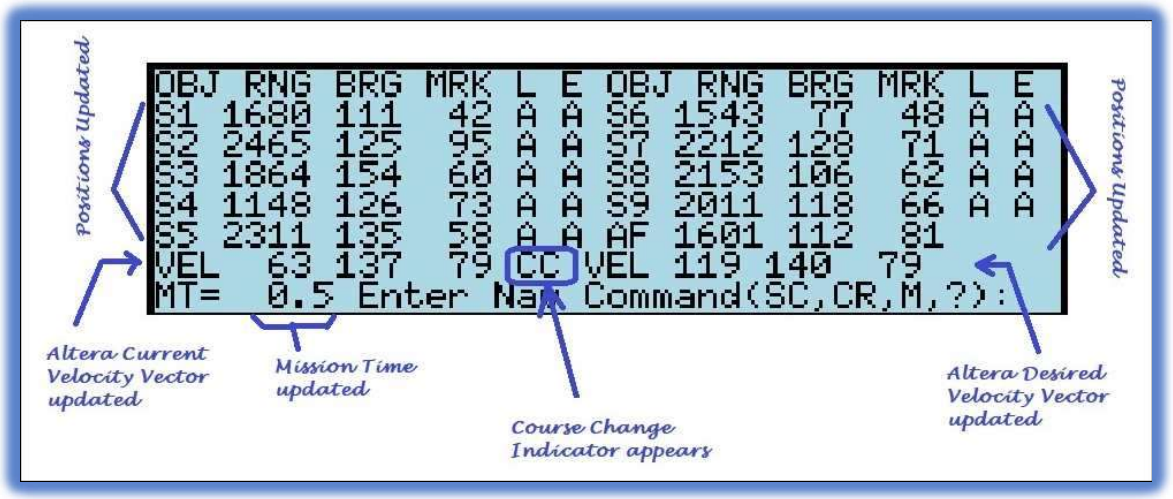


**Figure 4. New Desired Course Entered** *(from the notebook of Elias S. Dennis)*

It should be noted that the displayed desired course of 119, 140, 79 is different from the course setting of 120, 140 and 80 that was entered. To save memory space on the laptop, the coordinates for the enemy ships are recorded as integer numbers. These coordinates are converted to spherical coordinates for display purposes resulting in round off and truncation errors. These errors will not impact the mission.

When it is desired to simply pass time and not do anything else, the Cruise command (CR) is available. When this command is selected, a prompt for the number of minutes (0-9) will appear. If the letter X or the number 0 is entered, the cruise command is cancelled Any other entry (besides a number) is ignored. When a number greater than 0 is entered, the cruise mode is entered. The bottom line of the display will indicate the current mission time and the message, "Cruise in Progress (X Cancels)". Pressing the letter X will abort the cruise and the navigation menu will be displayed. Otherwise, the display and mission time are updated as each 0.5-minute mission time sequence is completed. This is repeated until the end of the cruise time. As an example, a cruise of 1 minute is performed which results in the following display.


**Figure 5. Navigation Menu after 1 Minute Cruise** *(from the notebook of Elias S. Dennis)*

From the data in the above display, the goal of moving Altera into combat range with enemy ship, S4 was achieved. The mission time, current velocity vector and positions have all been updated. The CC (course change) indicator is still displayed. This will likely clear in the next 0.5-minute mission time interval.

**Ordnance**

The ordnance menu is available by entering the letter, "O" from Altera's main menu. The first six lines from the main menu display will remain the same but the bottom two lines will be updated with ordnance status and a request for ordnance commands. Until a target is selected the ordnance status consists of the number of available torpedoes and the fighter status. Altera is equipped with 7 torpedoes and 10 fighters. The Launch Fighters command is available when there are less than 5 fighters deployed and the number of fighters on-board is greater than 0. When launched, the fighters deploy outside Altera's energy shield but remain close to Altera. When fighters are deployed, Altera's velocity changes will be limited to 40 Mm/min so as not to abandon the fighters. If greater velocity changes are required, the Recall Fighters (RF) command can be issued to bring the deployed fighters back on-board and remove the velocity change restriction.


**Figure 6. Ordnance Menu Display** *(from the notebook of Elias S. Dennis)*

Fighters will attack any enemy ships within range. The fighter's max attack range is 1000 +/- 150 Mm. The max range will vary each 0.5-minute interval due to factors that are not discussed here. If more than one enemy ship is within range, a fighter will select one as its target and fire its lasers at it. The probability of a fighter laser hit is 50%. This probability is not affected by the distance to the target. When a fighter's laser hits an enemy ship, the results are as effective as a hit from Altera's laser. The fighters are a valuable weapon system that can attack multiple enemy ships while Altera is engaged in other activities. Unfortunately, since the fighters are deployed outside of Altera's energy shield, they are vulnerable to enemy ship laser fire. A single hit from an enemy laser will destroy a fighter.

Altera is also equipped with lasers and torpedoes. In order to use Altera's lasers or torpedoes, a valid target must first be selected by issuing the Select Target (ST) command and entering a valid target number. A valid target is any ship 1-9 that has ***not*** been destroyed. When a valid target

is selected, the ordnance display is updated to include the probability of torpedo hit, the probability of a laser hit and the selected target. The following screen shot is an example of this display after enemy ship 4 has been selected as a target.


**Figure 7. Target Hit Probabilities** *(from the notebook of Elias S. Dennis)*

The Select Target command is one of the few commands that do not require any mission time to execute. The other commands are the menu navigation commands (M, N, O and D) and the "?" command that is used to list commands. This allows the ST command to be used several times to check the hit probabilities for multiple enemy ships before one is selected based on the hit probability and seriousness of the threat. The range probability for both lasers and torpedoes is dependent on the range.

Altera is armed with seven Doomslinger torpedoes. A Doomslinger torpedo is a sub-light velocity, high energy mass which becomes critical upon impact. A hit from a torpedo will inflict damage to the enemy ship that is roughly equivalent to five laser hits. The probability of a torpedo hit is 99% when the range to the target is 250 Mm (or less). This probability drops off by approximately 25% as the range doubles such that at 4000 Mm, the hit probability is about 1%. There is some speculation that several of the enemy ships possess tractor beams[3]. If an enemy ship applies a tractor beam at the same time a torpedo is fired, calculations predict the probability of a hit will be 100%. The following graph illustrates the torpedo hit probability (without considering enemy ship tractor beams).

[3] A tractor beam applied to Altera will result in the addition of a velocity vector to Altera's current velocity vector. Analysis of data from reconnaissance spacecraft predict that the added velocity vector magnitude will be 10 Mm/min towards the enemy ship using the tractor beam.

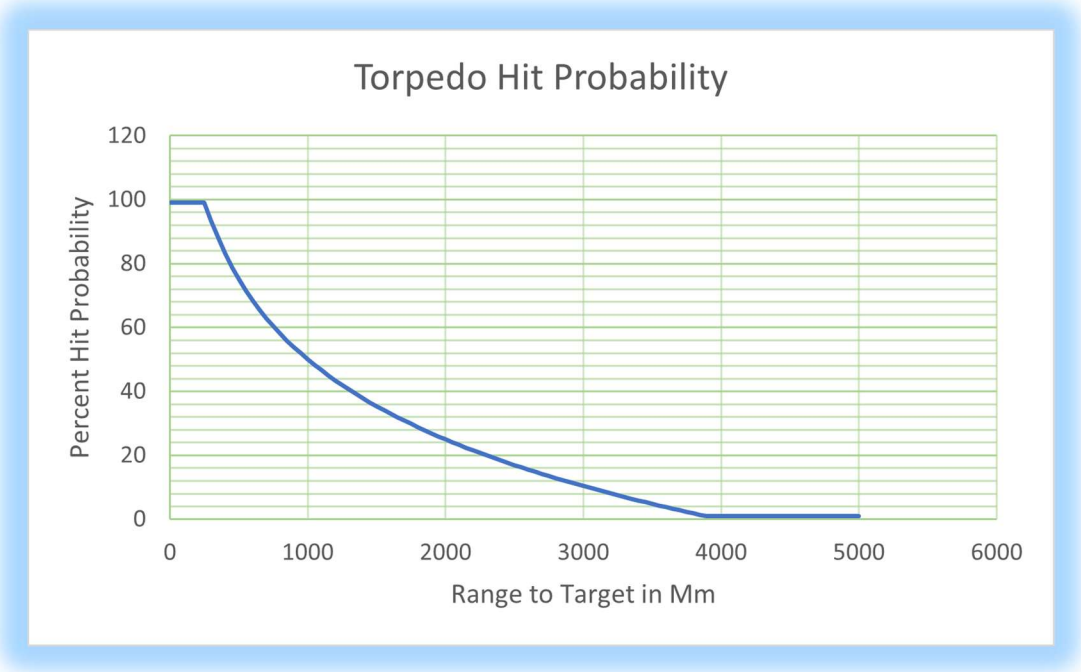


**Figure 8. Torpedo Hit Probability** *(from Doomslinger Operational Specification DSOS8214-2)*

Altera laser firepower is provided by the Triad system. Triad consists of three high energy laser rifles controlled by the Wellington focus/combine fire direction unit. When all three laser rifles are focused on a single target, considerable damage is inflicted with negligible energy expenditure. The maximum range where a 99% hit probability will occur is between 850 Mm and 1050 Mm. When the range exceeds the 99% hit probability range, the probability will decrease linearly as the range increases and will reach 1% at 200 Mm beyond the 99% hit probability range. This assumes that the energy unit level of the energy bank assigned to the lasers is 15. The 99% hit probability range varies with each 0.5-minute mission time sequence.

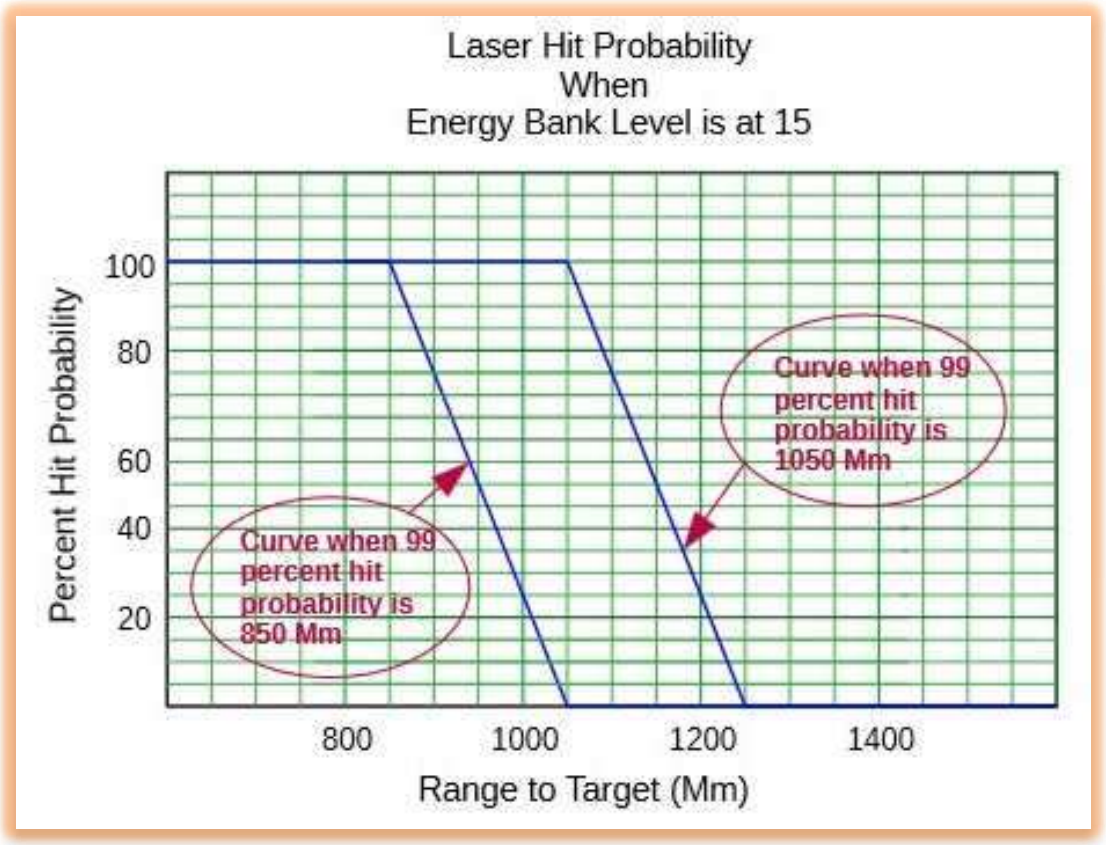


**Figure 9. Laser Hit Probability with Energy Bank at 15** *(from the notebook of Elias S. Dennis)*

When the energy bank assigned to Altera's laser is depleted from the maximum level of 15, the 99% hit probability range is decreased. The amount that it is decreased is approximately equal to 6.67% for each energy unit the energy bank is below 15. For example, if the 99% hit probability range was 1000 Mm and the energy bank assigned to lasers was at 13 energy units, the 99% hit probability range will now be 867 Mm. In this case the range for when the hit probability drops to 1% is 1067 Mm. A characteristic of the Triad laser system is the 1% hit probability range is always 200 Mm beyond the 99% hit probability. Fortunately, the 99% hit probability range can be restored by either repairing the energy bank or reassigning lasers to a different energy bank. This is described in the Damage Control section of this document.

After the course setting and completion of the cruise command described in the Navigation section, the probability of a torpedo hit increases to 51% and the probability of a laser hit increases to 99% as shown in the following display:

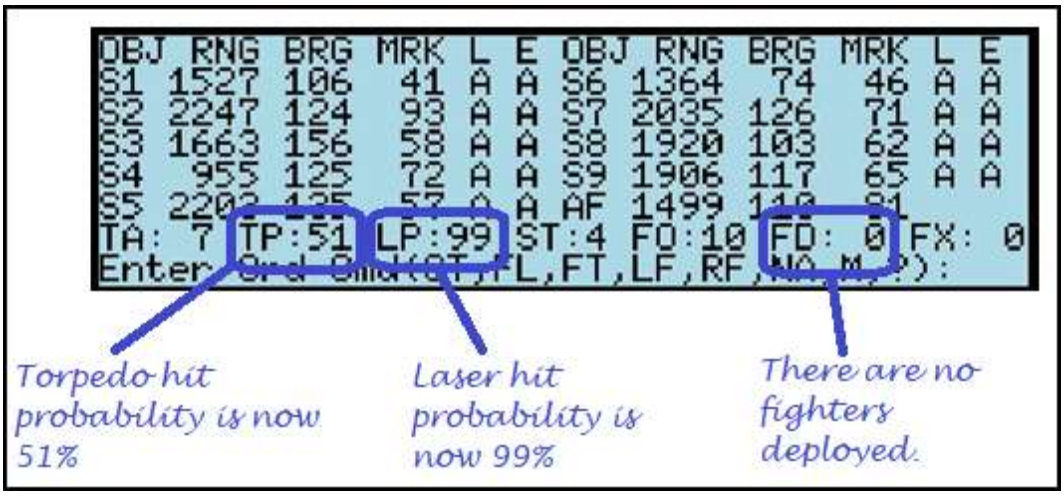


**Figure 10. Ordnance Menu after 1 Minute Cruise** *(from the notebook of Elias S. Dennis)*

In the above situation, it would be tempting to use the Fire Laser (FL) command, but a better choice might be the Launch Fighters (LF) command. This could result in five laser hits on enemy ship 4 from fighters instead of a single "for sure" laser hit from Altera. As an example, the Launch Fighters command was issued. Whenever there is activity with the enemy ships, a status notification display will appear and tell you what that activity was. When there is more activity than can be printed on a single display, multiple displays are used. The status notification displays that resulted from launching fighters are as follows:

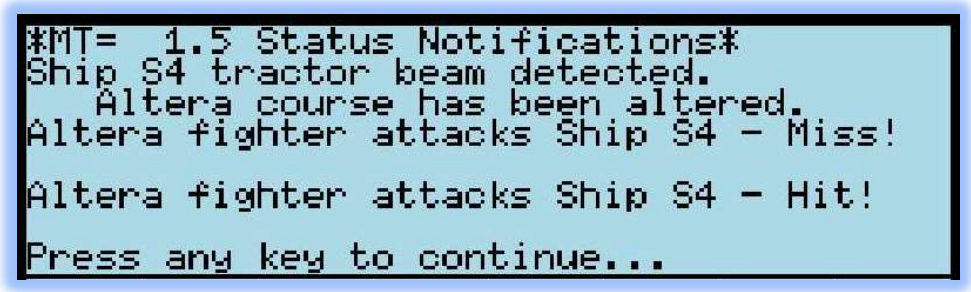


**Figure 11. Status Notification Screen 1** (from the notebook of Elias S. Dennis)

After pressing any key, the next Status Notification screen appeared.

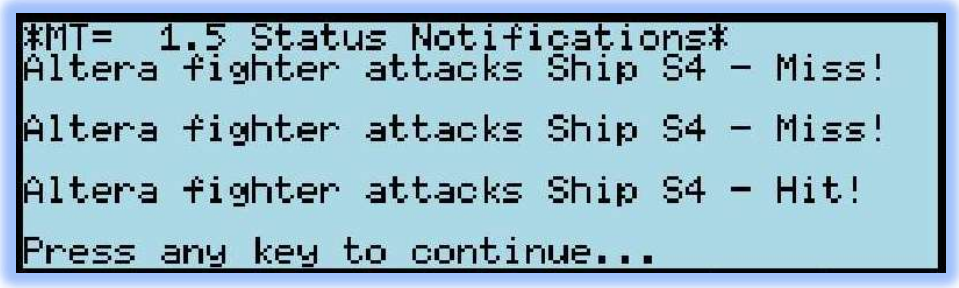


**Figure 12. Status Notifications Screen 2** *(from the notebook of Elias S. Dennis)*

So, Altera's fighters scored two hits on enemy ship S4 and enemy ship S4 used a tractor beam on Altera. After pressing a key, the ordnance menu is displayed with updated status.

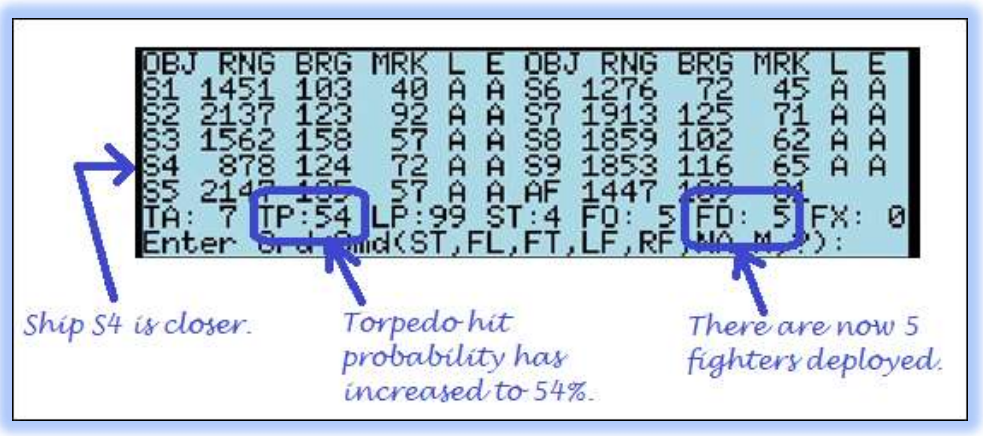


**Figure 13. Ordnance Menu after Launching Fighters** *(from the notebook of Elias S. Dennis)*

Enemy ship S4 range is decreasing. Although the probability of a torpedo hit now shows 54%, the probability is much higher since enemy ship S4 has tractor beams.

## Damage Control

Altera's armor consists of an impenetrable energy shield. The energy from any enemy missiles or lasers will always be harmlessly dissipated. However, this is not without a drain on the ship's resources. Each hit absorbed by the shields results in a drain on one or more of Altera's four energy banks which are designated A-D. Each energy bank starts at 15 energy units. Energy units are depleted from the Altera's four energy banks to absorb the energy from enemy missiles and lasers.

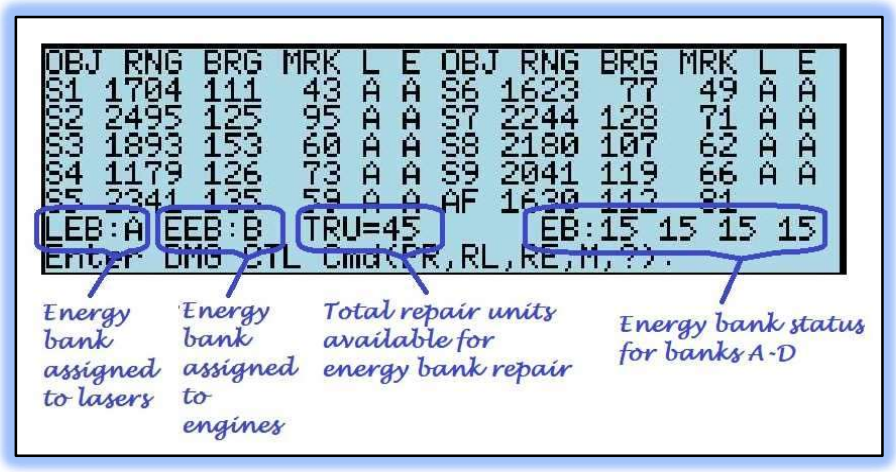


**Figure 14. Damage Control Status Line** *(from the notebook of Elias S. Dennis)*

There are consequences to depleted energy banks. The performance of Altera's engines and lasers are both degraded as the energy bank assigned to them is depleted. Initially lasers are assigned to energy bank A and engines are assigned to energy bank B. The Reconfigure Lasers (RL) and Reconfigure Engines (RE) in the Damage Control menu can be used to reassign lasers or engines to a less damaged energy bank provided the new energy bank is not destroyed (has 0 energy units) or has not already been assigned.

In the damage control menu, there is a Perform Repair (PR) command. This command allows you to allocate up to 7 energy units to distribute to the four energy banks. This operation

requires two minutes of mission time to complete. During this time, no other commanding is possible. If the letter, “X” is pressed during the repair sequence, the repair is cancelled, and new commands can be entered at the next 0.5-minute mission time sequence.

Once an energy bank is depleted to 0, it is destroyed and cannot be repaired with the Perform Repair (PR) command. If two energy banks are destroyed, the mission will be aborted, and the warp drives will be engaged returning Altera to the nearest star base for repair and refurbishment. Altera is equipped with 45 energy units that can be distributed to the energy banks during the Perform Repair command. Once these are used, no further repair is possible.

**Postlude**

Before completion of this document, Elias S. Dennis fell gravely ill with space sickness and has been confined to sick bay. He will not be able to assist in the coming operation to neutralize the artifact. You are on your own. Good luck!

## SECTION 3 – PROGRAM DESCRIPTION

The Space Game program consists of an initialization section, a main logic section, an end section and subroutines.

### Initialization Section

The initialization section immediately follows the header comments and begins at line 40. All variables from A-Z are defined as integer unless identified by a "!" or "$" for floating point and string respectively. Throughout the initialization section the dimension of array variables and the initial value of constants and variables are set.

The random number function, RND will always generate the same random number series each time it is used. By making multiple calls to RND(1) while waiting for the user to press a key at the start of the game, new random initial conditions are achieved each time the game is played.

Two sets of rectangular coordinates are maintained. One has the artifact as the origin and the other has the Altera as the origin. The initial positions of the Altera rectangular coordinates referenced to the artifact are set randomly between +/-3000 Mm. An initial velocity towards the artifact is set by dividing each of the rectangular coordinates by 100. The energy banks are all set to 15 and a message is displayed indicating the initial velocity towards the artifact with instructions to standby.

There is a delay of about 15 seconds as the remaining code in the Initialization Section is executed starting with the initialization of positions and parameters of the nine enemy ships. The initial x, y, and z coordinates with respect to the artifact of the nine enemy ships is set randomly between +/-1000 Mm. The laser and engine status of each enemy ship is set to 10. The rectangular coordinates referenced to the Altera of each enemy ship are converted to spherical coordinates by calling the Rect2Sphere subroutine located at line 1930. The initialization section ends by setting the initial value of several Altera parameters (number of torpedoes, number of fighters, repair units, etc.) and setting the desired velocity equal to the actual velocity (more or less). Finally, three of the nine enemy ships are equipped with 5 to 10 missiles and three other enemy ships are equipped with tractor beams. Program execution then drops into the Main Logic Section.

### Main Logic Section

The main logic section consists of a menu subsection and a time passes subsection. At the beginning of the main logic section, the screen is cleared and a subroutine is called to display positions. The beginning of the main logic section is branched to whenever data on the first six lines can change. This happens at the end of the time passes subsection or if the screen is used to display available commands by pressing the "?" key.

The menu subsection begins with an ONMNGOSUB statement to direct the program flow to the menu identified by the variable, MN. It is then followed by code for each of the four menus. These menus are the main menu and the three sub-menus for navigation, ordnance and damage control. Generic status on the velocity, fighter deployment and energy banks is provided on the 7$^{th}$ line of the display. The mission time and a request for main menu commands is displayed on the 8$^{th}$ line. The main menu's sole purpose is to provide commands to navigate to the submenus.

When a submenu command (N, O, or D) is entered, the variable MN, is updated to a value corresponding to the desired submenu. When a branch back to menu subsection (the ONMNGOSUB statement) is executed, the program branches to appropriate submenu code based on the variable, MN.

The code for the three sub-menus behaves similarly. Each of the three submenus begins by displaying status on line 7 that is relevant to commands provided by the submenu. For example, for navigation, the actual velocity, a course correction flag and the desired velocity are displayed on line 7. Each of the submenus has one command that requires more than one 0.5-minute game sequence to complete. For navigation, it is the cruise command, for ordnance, it is the neutralize artifact command and for damage control, it is the perform repair command. If any of these commands were entered, during a previous 0.5-minute game sequence, the variable ET is set to a value greater than 0. If it is still greater than zero, when the submenu is executed, a message indicating the command is still in progress is displayed on line 8 and the program jumps to the time passes subsection of the main logic section.

If the variable ET is 0, a command is requested on line 8. The available commands are listed in parenthesis with this request. Only valid upper-case letters are accepted as commands. The INKEY$ function is used for obtaining data one character at a time. The first character of each command is checked to see if it matches the first character of a valid command. Once the first character matches, a second character is obtained and checked. If it matches valid commands corresponding to the first character, command processing begins by the setting or obtaining of parameters. Regardless of the command, only legal parameters are accepted as inputs and illegal key presses are ignored. The letter X can be pressed during the command entry to set the cancel flag. If the cancel flag is set, the command entry and the command are canceled, and the program will branch to the menu subsection (the ONMNGOSUB statement) where it will branch back to the submenu where it previously was and a command will once more be requested.

Similarly, if an invalid command is entered, no error message is printed, the entry is ignored and the program branches to the beginning of main menu section where it is routed by the ONMNGOSUB statement to return to the appropriate submenu and the command is then requested again.

If a legal command is entered along with legal parameters (if any), program execution will branch to the time passes subsection of the main logic section, unless the command is “ST”, “?”, “N”, “O”, “D” or “M”. These commands do not require any mission time to execute. All other commands require at least 0.5-minutes of mission time to execute. The commands that do not require any mission time to execute will branch back to the menu subsection (the ONMNGOSUB statement), except for the “?” command. After completing the “?” command, the program branches back to the beginning of the main logic section so the screen can be refreshed.

The time passes subsection consists of twenty lines of code from 1480 to 1670 followed by a branch back to the beginning of the Main Logic section. Included in this section is code to update random range parameters and increment the mission time by 0.5. There is code to check if the extended time for the cruise, perform repair and neutralize artifact has expired. If the neutralize artifact timer expires, the game is over and a branch is made to the end logic section for a victory message. This is the only branch (GOTO) within the first twenty lines of the

subsection. All other statements are performed in order. Here are the major activities that are performed during each 0.5-minute mission time sequence:

| Line | Subroutine Called | Time Passes - Action Performed | Subroutine Location |
|---|---|---|---|
| 1480 | Update Altera Velocity | Perform velocity update if course correction flag is set. | 3110 |
| 1490 | Move Altera | Move Altera by updating the position vector using the velocity vector. | 3220 |
| 1530 | Tractor Beams | There is a 33% chance that an enemy ship with tractor beams will use them if within range. | 3280 |
| 1540 | Altera Fires Lasers or Torpedoes | If firing of lasers or a torpedo was commanded, the hit/miss is determined, and damage is distributed accordingly to the enemy ship. If neither command was issued, the subroutine returns with nothing happening. | 3430 |
| 1550 | Enemy Ships Attack | Each enemy ship will evaluate if Altera is in range and will attack the Altera with missiles (if it possesses them) or will attack fighters (if deployed) or the Altera. | 3620 |
| 1560 | Fighters Attack | If fighters are deployed, a subroutine is called to perform the fighters check range and attack. Only enemy ships within the fighter's range | 3900 |

| Line | Subroutine Called | Time Passes - Action Performed | Subroutine Location |
|---|---|---|---|
| | | (previously determined) will be attacked. | |
| 1570 | Update enemy ship velocity. | A call to this subroutine may result (33% probability) in an enemy ship updating its velocity vector to move it towards its destination (either Altera or the artifact). | 4130 |
| 1580 | Update enemy ship positions | Each enemy ship that is not destroyed will have its position updated by adding the velocity vector to the position vector in rectangular coordinates then converting the coordinates to spherical coordinates. This chews up time and is why turns pass faster with more enemy ships destroyed. | 4280 |
| 1590 | Perform Repairs | Repairs are performed to both the Altera (if the Perform Repair command is in progress) and the enemy ships. One of the nine enemy ships is selected randomly and if the selected ship is not destroyed, the status for its engines and lasers are incremented. | 4350 |
| 1660 | Convert Altera Velocity to Spherical Coordinates | If the Update Velocity (UV) flag is set, Altera's velocity is | 1930 |

| Line | Subroutine Called | Time Passes - Action Performed | Subroutine Location |
|---|---|---|---|
| | | converted to spherical coordinates so that it is display properly in the menu subsection. | |
| 1670 | Compute torpedo/laser hit probability | If the target selected (TS) is valid (greater than 0), the new torpedo and laser hit probabilities are computed. This is necessary since the range to the enemy ships has more than likely changed. | 2620 |

At the completion of the time passes subsection, a GOTO statement is executed to branch back to the beginning of the main logic section where the menu subsection is located.

**End Section**

The end section consists of two messages – one for victory and the other for defeat. The victory message is displayed upon successful completion of the neutralize artifact command. This message reads as follows:

> **Congratulations! You have successfully neutralized the artifact. Altera will now engage its warp drive to return to the Epsilon Eridani starbase. Promotions and other rewards will be waiting for you. Nice job!**

The defeat message is branched to directly during the Evaluate Altera Damage subroutine that is found at line 4520. This occurs if the value of two energy banks has reached zero. The defeat message reads as follows:

> **With the loss of the second energy bank, Altera is no longer capable of completing the mission and will return to the Epsilon Eridani starbase for" refurbishment. Perhaps the next attempt will be more successful.**

After either the victory or defeat message is printed, the screen is cleared, a thanks for playing message is displayed, the program ends, and the OK prompt reappears.

**Subroutine Section**

This program uses twenty-six subroutines. They are found following the end section of the program. The subroutines that are common or are used in the initialization section or the menu subsection appear first. The subroutines that are called from the time passes subsection appear next. (At least that was my intention.) A list of the subroutines, a description of what they do, and the lines that call them is found in Appendix B following the list of variables in Appendix A.

## APPENDIX A – VARIABLE LIST

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| A | ASCII value for capital A. This is a constant set to 65. | 40 270 1160(2) 1400 4550 |
| A$ | Temporary variable used for character at a time input (INKEY$) | 430(2) 440 450 460 540(3) 550 570 580(3) 600(2) 610 640 650(3) 840(3) 850 870 880(3) 900(2) 910 940 950(3) 1000 1010(3) 1040 1050(4) 1060 1110 1120(3) 1220(3) 1230 1240 1250(4) 1260 1350 1360(3) 2680(2) 2690 2740(2) 2750 2760 2770 2810(2) 2820 2830 2840 2920(2) 2930 4310 4330(2) |
| BR(3) | Repair units allocated for repairing energy banks, A-D (0-3) | 1380 1420 4410 |
| BS(11) | Spherical Coordinate Array Bearing component. Index 0 represents artifact bearing (phi) component to the player's ship on the xy plane. Bearing represents the angle to the object's position projected on the xy plane. It is measured from the positive x axis to the positive y axis and can vary from 0 to 359. Indices 1-9 is the bearing (phi) component of enemy ships (to players ship). Index 10 is the player ship's actual velocity range component. Index 11 is the player ship's desired velocity bearing (phi) component. | 60 280(2) 380 490 510 1940 1950(3) 1960(3) 1970(3) 2090 2140 |
| C1(9) | Indices 1-9 represent the condition of the enemy ship's lasers. A=active (10-6), D=Disabled (1-5), X=Destroyed (0). | 230 2100(2) 2970(2) 2980 3750 4450(3) 4470(2) |
| C2(9) | Indices 1-9 represent the condition of the enemy ship's engines. A=active (10-6), D=Disabled (1-5), X=Destroyed (0). | 230 2110(2) 2970(2) 2980 3650 4130 4460(3) 4480(2) |
| CC | Course Correction Flag<br>CC=0: No course change in progress. CC=1: Course change in progress. | 390 500 760 1480 3130 |
| CF | Cancel Flag – Set to 1 if X is pressed during the enemy position ship update subroutine or press any key subroutine. This will cancel the RP, NA or CR command. Otherwise, this CF=0. | 670(2) 690(2) 710(2) 1410 1430(2) 1650(2) 2750 2820 2930(2) 4310 4330 |

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| CS(3) | Course Setting Parameters. Indices 0-2 are temporary input values for velocity magnitude, bearing and mark accordingly. These values are entered during the Set Course command. Index 3 is a temporary variable for the maximum allowed values. This will be set to 999, 360, 180 depending upon if magnitude, bearing or mark is being requested for input. | 670(2) 690(2) 710(2) 720(3) 730(3) 740(2) 2770 |
| CT | Cruise Time (number of 0.5-minute turns to execute). The user entered value of minutes is doubled and assigned to this variable. ET is then set 1. | 620 1610(3) 1640 1650 |
| D(9) | Enemy ship destroyed/distance flag. Indices 1-9 correspond to each enemy ship. Values are defined as follows:<br>0: The maximum distance from the artifact allowed is 3000 to 5000 MM.<br>1: The maximum distance from the artifact allowed is 1000 to 2000 MM.<br>2: The enemy ship has been destroyed. | 210(2) 910 2080 2990 3280 3620 3650 3990 4080 4130 4160 4280 |
| D9 | Damage inflicted by enemy ships. Set to 1 for lasers, 5 for missiles, and anywhere from 10 to 20 when an enemy ship self-destructs (dependent on range). | 3030 3660 3730 3850 |
| DR! | Degrees to Radians conversion factor. Set equal to 180/pi or 57.296. | 40(2) 720(2) 730(2) 740 |
| DV | Delta V. The magnitude of the velocity difference between the actual velocity and the desired velocity. | 3120 3130 3140(2) |
| DV! | Dummy Variable used for storing a random number between 0 and 1. | 90 4140 4160(2) |
| E0 | Identifies the first Energy bank that was destroyed. | 4520 4530 4550 |
| E1 | The number of energy banks that were destroyed during previous turns. | 270 4550 4570 4580 |
| E2 | The number of energy banks that have been found destroyed as of this turn. If it is 2, the game is over. | 4520 4530(2) 4550 4560 4570 4580 |
| EB!(3) | Indices 0-3 correspond to energy banks A-D respectively. Value represents the current status of each energy bank in energy units. | 150 410 1060 1190 1520 2700 3030(4) 3110 4400 4410(4) 4530(2) |
| EE | Engines' Energy Bank (0-3). Represents the energy bank A-D that the engines use for operation. | 270 1160 1290 2700 3110 |
| ET | Extended Time. This is set to 1 or more whenever the PR (Perform Repair) command from the damage control menu (MN=4), the CR (Cruise) command from the Navigation menu (MN=3) or the NA (Neutralize Artifact) from the Ordnance menu (MN=2) is selected. The value that is set | 520 620 820 1130 1180 1200 1440 1610(2) 1620(2) 1630(2) 1640(2) 1650 |

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| | represents the number of 0.5 mission time sequences (turns) that will be performed. | |
| EW(9) | Extra Weapons. Identifies the extra weapons each enemy ship has available. A value of 0 means no extra weapons. A value of 1 means the enemy ship has missiles. Value of 2 or 3 indicates tractor beams. A value of 3 indicates tractor beams have been used during the turn. The value is set to 2 if the ship decides not to use the tractor beams. | 60 300(2) 310(2) 3280 3290 3390 3550 3620 |
| FA | Fighters Attacking – initially equal to the number of fighters that will be attacking. This variable is decremented each time a fighter attacks and is set to zero if no available targets remain. | 3920 3930(2) 4010 4030 |
| FC(2) | Fighter craft: index= 0 is number on board, =1 is the number that are deployed, =2 is the number that were destroyed. | 60 270 400 810(3) 940(2) 960(9) 970(5) 1000 1020(4) 1560 3110 3660(4) 3760 3810(4) 3920 |
| FT(9) | Fighter Targets. Index 0 is the number of targets in range. Indices 1 to the value of FC(0)is the index of the enemy ship that is in range. | 3910 3930(2) 4010 4070 4090(2) |
| I | Array index. This variable is usually used to index enemy ships in several subroutines including the rectangular to spherical coordinates conversion subroutine and the enemy ship damage distribution subroutine. | 140 150(2) 190 200 210(2) 220(3) 230(2) 240(6) 290 750 1660 1930(4) 1940(3) 1950(6) 1960(3) 1970(3) 1980(2) 1990(4) 2000(3) 2040 2050 2060 2080 2090(4) 2100(2) 2110(2) 2970(4) 2980(2) 2990(2) 3240 3280(3) 3290(2) 3320 3350(2) 3360(2) 3370(2) 3390 3450 3480 3510 3540 3550 3570 3620(4) 3630 3650(4) 3660 3690 3710 3720(2) 3750 3780 3830 3870 3930 3960 3990 4000 4080(3) 4090 4130(3) 4140(4) |

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| | | 4150(3) 4160 4170(5) 4180(5) 4190(5) 4210(5) 4220(5) 4230(5) 4280(2) 4290(9) 4300(6) 4340 |
| J | Temporary loop variable. | 300(4) 310(3) 410(2) 1190(2) 1380(2) 1390 1400 1410 1420(2) 1430 3030 3560 3930(2) 4070 4090(4) 4400(2) 4410(5) 4440 4450(3) 4460(3) 4470(2) 4480(2) 4520 4530(3) |
| K | Temporary loop variable used in distributing and evaluating damage to the player's ship. | 3030(5) 4520 4530 4570 |
| LE | Lasers Energy Bank (0-3). Represents the energy bank A-D that the lasers use for operation. | 270 1060 1160 1320 1520 2700 |
| LF | Lasers Fire flag. This flag is set to 1 when lasers are to be fired. It is cleared when the lasers are actually fired. | 1070 3430 3480 |
| LP | Line Pairs printed. This variable is incremented each time a two-line message is displayed. (Used for screen header & continuation message) | 1500 3300 3310(2) 3340(2) 3440 3450(2) 3580(2) 3640 3650(2) 3700 3710(2) 3770 3780(2) 3820 3830(2) 3860(2) 3940 3950(2) 4020(2) |
| LR | Laser Range for Altera. This represents the range where the probability of a laser hit is 99%. | 1510 1520(2) 2640 |
| MC(9) | Missile Count. This is the number of missiles each enemy ship has. It is zero for all but 3 enemy ships. Each time an enemy ship fires a missile, the missile count for this ship gets decremented. | 60 300 3690 3720(2) |
| MN | Menu Number – 1 is Main, 2=Navigation, 3=Ordnance 4=Damage Control. This directs which menu will be displayed. The value is only changed by the menu navigation commands, N, O, D, and M. | 270 360 440 450 460 550 850 1230 |
| MR | Maximum Range of enemy ships. This is randomly set to 1000+/-150MM each 0.5 minute turn sequence. | 1500 3290 3620 3900 4080 |

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| MS(11) | Spherical Coordinate Array Mark (theta) component. Index 0 represents the artifact's mark component from the player's ship. Mark is measured from the positive z axis towards the object. It can vary from 0 to 180 degrees. (A mark of 0 degrees is straight up and a mark of 180 is straight down.) Indices 1-9 represent enemy ships' 1-9 mark component to the player's ship. | 60 280(2) 380 490 510 1980 1990 2000 2090 2140 |
| MT! | Mission Time – Increments by 0.5 during each turn sequence. | 270 420 520 530 820 1600(2) 2880 |
| MV | Maximum Velocity change of player's ship due to assigned energy bank and fighter craft deployment. | 3110(3) 3140(2) |
| NT | Neutralize Artifact Time - Set to 10 when the player selects the Neutralize Artifact command. This represents the number of 0.5-minute turns that will be executed before declaring the game won. | 1130 1620(3) 1640 1650 |
| P2! | Pi/2 – A constant initialized to 1.570796. | 40 2000 |
| PL | Probability of a laser hit. The value will be 0 if a target isn't selected and will be from 1 to 99 when valid target is selected. | 800 2640(5) 3490 |
| PT | Probability of a torpedo hit. The value will be 0 if a target isn't selected and will be from 1 to 99 when valid target is selected. | 800 2620 2630(4) 3550 |
| R1 | Total Repair units allocated by the PR (Perform Repair) command. | 1380 1420(2) 1440 |
| RA | Range to Artifact. This variable is used to temporarily store the range to the artifact for an enemy ship. | 4140 4160(2) |
| RD! | Radians to Degrees conversion constant. This value is initialized to 1/DR!. | 40 1950 2000 |
| RS(11) | Spherical Coordinate Array Range component. Index 0 represents artifact's range to the Altera. Indices 1-9 is the range component of enemy ships to Altera. Index 10 is the Altera's present velocity range component. Index 11 is the Alter's desired velocity range component. | 60 170 280(2) 380 490 510 1110(2) 1930 1980 1990 2000 2090 2140 2620 2640 3290 3350 3360 3370 3620 3630 3660 4080 4140 |
| RT | Repair Time - Set to 4 (the number of 0.5 minute turns to execute) when the player selects the PR (Perform Repair) command. | 1440 1630(3) 1640 1650 4380 |
| RU | Repair units that are available to allocate. | 1370(2) 1400 1420(3) 2840 |
| SL | Screen location – used with PRINT@ statements. | 2050 2060 2070(2) 2080 2090 2130(2) 2140 |

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| TF | Torpedo Fired flag – set to 1 when a torpedo is fired. | 1090 3430 3460 3540 |
| TN | Temporary number. This variable is used in the subroutine to get a 3-digit number | 2760(4) 2770(4) 2830(4) 2840(3) |
| TP | Number of torpedoes available to use (initialized at 7). | 270 790 1080 1090(2) |
| TR | Total Repair Units available (initialized to 45. | 270 1170 1350 1370(2) 1440(2) |
| TS | Target Selected = 1-9 if a valid target is selected, otherwise it is 0. | 800(2) 920 1040 1670 2620 2640 2990(2) 3450 3510 3570 3650 |
| TV | Temporary Variable – used during target selection, energy bank reassignment and maybe a few other places. | 610(3) 620 670 690 710 910(4) 920 1290(2) 1320(2) 1420(3) 2690(4) 2700(4) 2730 2770(3) 2800 2840 |
| UV | Update Velocity flag. Set to 1 if the rectangular coordinates for the Altera velocity vector have been changed by a course change or an enemy ship tractor beam. This will trigger a conversion to spherical coordinates at the end of the 0.5-minute turn. | 1660(2) 3180 3380 |
| VF! | Velocity Factor – Amount of delta V of the desired course that will be used to update Altera's velocity vector. Also used to for computing the velocity vector of enemy ships. | 3140(2) 3150 3160 3170 4140 4150(3) |
| X2(9) | X component of enemy ship 1-9 velocity. | 4150 4170(3) 4210(3) 4290 |
| XA(9) | Rectangular Coordinate Array X coordinate. Index 0 is the player ship referenced to the artifact. Indices 1-9 represent enemy ships 1-9 referenced to the artifact. | 50 110 120 220 240(2) 3230 4140 4210(2) 4290(2) 4300(2) |
| XS(11) | Rectangular Coordinate Array X coordinate. Index 0 represents artifact position referenced to the player's ship. Indices 1-9 represent ships 1-9 referenced to the player's ship. Index 10 represents the player ship's velocity vector x component. Index 11 represents the player ship's desired velocity x component. | 50 120 130(2) 240 720 1930 1940 1950(2) 3120(2) 3150(4) 3220(3) 3230 3350(3) 4170(2) 4300 |
| Y2(9) | Y component of enemy ship 1-9 velocity. | 4150 4180(3) 4220(3) 4290 |
| YA(9) | Rectangular Coordinate Array Y component. Index 0 is the player ship referenced to the artifact. Indices 1-9 represent ships 1-9 referenced to the artifact. | 50 110 120 220 240(2) 3230 4140 4220(2) 4290(2) 4300(2) |

| Variable Name | Variable Description | Lines Referenced |
|---|---|---|
| YS(11) | Rectangular Coordinate Array Y coordinate. Index 0 represents artifact position referenced to the player's ship. Indices 1-9 represent ships 1-9 referenced to the player's ship. Index 10 represents the player ship's velocity vector y component. Index 11 represents the player ship's desired velocity y component. | 50 120 130(2) 240 730 1930 1940 1950 3120(2) 3160(4) 3220(3) 3230 3360(3) 4180(2) 4300 |
| Z | Used in DEFINT statement. | 40 |
| Z2(9) | Z component of enemy ship 1-9 velocity. | 4150 4190(3) 4230(3) 4290 |
| ZA(9) | Rectangular Coordinate Array Z coordinate. Index 0 is the player ship referenced to the artifact. Indices 1-9 represent ships 1-9 referenced to the artifact. | 50 110 120 220 240(2) 3230 4140 4230(2) 4290(2) 4300(2) |
| ZR! | Temporary variable indicating the ratio of the Z coordinate to the R coordinate in the rectangular to spherical conversion subroutine. | 2000(3) |
| ZS(11) | Rectangular Coordinate Array Y coordinate. Index 0 represents artifact position referenced to the player's ship. Indices 1-9 represent ships 1-9 referenced to the player's ship. Index 10 represents the player ship's velocity vector Y component. Index 11 represents the player ship's desired velocity Y component. | 50 120 130(2) 240 740 1930 1990(2) 2000 3120(2) 3170(4) 3220(3) 3230 3370(3) 4190(2) 4300 |

## APPENDIX B – SUBROUTINE LIST

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| 1930 | Rect2Sphere | Converts the rectangular coordinates for arrays XS, YS, and ZS to spherical coordinate arrays, RS, BS and MS for the item indexed by the variable I | 140 250 750 1660 3240 4320 |
| 2030 | Main Nav Display | Populates the first 6 lines of the main display with column headings, the position and status of enemy ships 1-9 and the position of the artifact. | 350 |
| 2180 | Clear Bottom 2 Lines | Clears data on the last two lines of the display by printing 79 spaces. | 440 450 460 550 580 650 670 690 710 760 850 880 950 980 1010 1020 1050 1060 1070 1080 1090 1120 1130 1230 1250 1290 1320 1330 1360 1430 1440 |
| 2210 | Main Help | List commands that are available on the main menu. | 430 |
| 2300 | Nav Help | List commands that are available on the nav menu. | 540 |
| 2390 | Ordnance Help | List commands that are available on the ordnance menu. | 840 |
| 2520 | Damage Control Help | List commands that are available on the damage control menu. | 1220 |
| 2620 | Torpedo/Laser Hit Probability | Calculates the torpedo and laser hit probability for the selected target designated by the variable, TS. The probabilities are saved in variables PT and PL. | 920 1670 |
| 2680 | Get Valid Energy Bank | Used by the reassignment of lasers or engines energy bank commands. Checks the keyboard for a valid energy bank letter A-D that is above zero and isn't already assigned to lasers or engines. The variable TV is set to 0-3 for valid energy banks or -1 for invalid. | 1290 1320 |
| 2730 | Get 3 digit number. | Used by the set course command to get values for velocity, bearing and mark. Only numbers that do not | 670 690 710 |

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| | | exceed max allowed, the enter key and the letter "X" are allowed inputs. If the letter X is pressed, the set course command is canceled. | |
| 2800 | Get a 1 digit number | Used by the perform repair command for getting the number of repair units allocated for an energy bank. Only numbers that do not exceed the repair units available (RU), the enter key and the letter "X" are allowed inputs. If the letter "X" is pressed, the perform repair command is canceled. | 1400 |
| 2880 | Status Header | After clearing the display, the mission time and the "Status Notifications" header is displayed on the first line of the display. | 3300 3440 3640 3700 3770 3820 3940 |
| 2910 | Press Any Key | Displays a message on the last line to press any key. The program loops until a key is pressed. Once a key is pressed, the message on the last line is replaced with the message, "Continuing…". If the key that was pressed was the letter "X", then the cancel flag is set such that any "long" commands are canceled. | 1870 3340 3580 3860 4020 4560 |
| 2970 | Distribute damage to enemy ship, I | Damage from one laser hit is distributed on an enemy ship to either its lasers or its engines. Each time this routine is called, 2 units are decremented from the corresponding status array, C1 or C2. When both fall to zero or below, the enemy ship is destroyed, a destroyed flag is set and the selected target variable, TS is cleared. | 3500 3560 3980 |
| 3030 | Distribute damage to Altera | An energy bank is chosen randomly and decremented. This is repeated for the number stored in the variable, D9. D9 is set based on the weapon inflicting the damage. | 3660 3730 3850 |
| 3110 | Update Altera Velocity | The Altera velocity is updated by the maximum amount available based on the status of the energy bank assigned to engines and fighter deployment | 1480 |

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| | | status. The Update Velocity (UV) flag is set to 1 to trigger a call to the rectangular to spherical subroutine. | |
| 3220 | Move Altera | Update the position of ship by adding ½ of each X, Y and Z component of the velocity vector to the artifact's position referenced to the ship. The ship's position referenced to the artifact is the negative of these items. A call is then made to the rectangular to spherical coordinate subroutine. | 1490 |
| 3280 | Tractor Beams | Each of the enemy ships is evaluated to see if they possess tractor beams. If they have them, there is a 33% chance they will use them. When an enemy ship uses a | 1530 |
| 3430 | Altera Fires Lasers or Torpedoes | This subroutine is called during each 0.5 min game segment regardless of if lasers or torpedoes were fired. If neither was, then an early return is taken. Otherwise, the subroutine will perform either the fire laser or fire torpedo section where hit/miss messages are displayed, and damage is distributed to the enemy ship appropriately. This subroutine (like other time passes subroutines with messages) contain a wrapper that will print out message headers prior to the message or press any key messages after the message based on the value of the line pairs printed variable, LP. | 1540 |
| 3620 | Enemy Ships Attack | This subroutine is called during each 0.5 min game segment. Each enemy ship is examined to verify it is alive, sees the Altera as within range and has not used tractor beams. If it passes those criteria, it might self-destruct causing considerable damage to Altera. There is a 10 percent chance this will happen during each 0.5 min game sequence if the range to Altera is less than 100 Mm. If the enemy ship doesn't self-destruct it is | 1550 |

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| | | checked if it has missiles. If it has a missiles, it will fire one at Altera. The hit/miss probability for enemy ship missiles is 50% regardless of range. A hit inflicted on Altera with a missile is equivalent to five hits with lasers. The enemy ship's missile count is decremented each time it fires a missile.<br>If the enemy ship does not have missiles or used them all, it will use its lasers (provided they aren't disabled or destroyed). The enemy ship's target will be Altera if no fighters are deployed. If fighters are deployed, then it is a 50% chance the target will be a fighter or the Altera. The hit probability on fighters is 50% and is 80% on Altera. A hit on a fighter, destroys it. The variable, D9 denotes the damage to Altera in energy units. The subroutine to distribute damage is called the number of times denoted by D9. After damage is distributed, a call is made to the subroutine to evaluate Altera damage. This subroutine (like other time passes subroutines with messages) contains a wrapper that will print out message headers prior to the message or press any key messages after the message based on the value of the line pairs printed variable, LP. | |
| 3900 | Fighters Attack | This subroutine is called during each 0.5 min game segment if the number of fighters deployed is greater than 0. At the beginning of this subroutine, the maximum range for the fighters is determined for the 0.5 min segment. Next the subroutine to determine the enemy ships within range is called. If there are no ships within range, this subroutine returns early. Each | 1560 |

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| | | fighter deployed will select a target from the list of available targets and attack it. A message is displayed indicating the attack and the hit/miss results. If the enemy ship is destroyed another call to the subroutine to find targets is made. If there are no more targets, the variable tracking the number of fighters to attack (FA) is set to 0. This subroutine (like other time passes subroutines with messages) contains a wrapper that will print out message headers prior to the message or press any key messages after the message based on the value of the line pairs printed variable, LP. | |
| 4070 | Find Targets for Fighters | The indices of enemy ships that are not destroyed and within range are put in an array starting at index 1. Index 0 of that array stores the number of enemy ships in range of the fighters. | 3900 4010 |
| 4130 | Update Enemy Ship Velocity | For each of the nine ships that do not have disabled or destroyed engines, there is a 33% chance they will update their velocity vector during a 0.5 min game segment. A velocity factor, VF! is computed from the distance the enemy ship is from the artifact. The greater the distance from the artifact, the greater the velocity factor will be. A random number between -0.5 and 0.5 is multiplied by 75 and VF! and assigned to the X, Y and Z components of the enemy ship's velocity vector. The signs of the components might be adjusted such that the resulting random vector will always be pointing in the same octant as the destination. The enemy ship's destination will either be the Altera or the artifact depending on if the | 1570 |

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| | | maximum allowed distance from the artifact was exceeded. The maximum allowed distance is recomputed each time the velocity is updated. The maximum will fall between 3000 and 5000 Mm for six of the nine enemy ships and between 1000 and 2000 Mm for the other three. | |
| 4280 | Update Enemy Ship Positions | During each 0.5-minute game sequence the position of each enemy ship that is not destroyed is updated. This is done by adding the enemy ship's velocity vector to the rectangular coordinates of the enemy ship position referenced to the artifact. The enemy ship position referenced to Altera is calculated by subtracting Altera's position referenced to the artifact from the enemy ship's position referenced to the artifact. The positions are then converted to spherical coordinates for the display. If during this loop the cancel flag is not set and the letter, "X" is pressed, the cancel flag will be set and a message on the bottom of the screen will be displayed. | 1580 |
| 4380 | Perform Repairs | This subroutine is called during each 0.5-minute game sequence and addresses repairs to both Altera and the enemy ships. Altera repair is performed if the Repair Time (RT) variable is not zero. When Altera repair is performed, 25% of the repair units that is allocated to each energy bank is added to the energy bank, provided the energy bank was not destroyed. If the resulting energy unit level of the energy bank exceeds 15, the energy bank is set to 15. During the enemy ship repair section of this subroutine, one of the nine enemy ships is selected randomly. | 1590 |

| Subroutine Line | Subroutine Name | Subroutine Description | Line(s) Calling Subroutine |
|---|---|---|---|
| | | One unit is added to the enemy ships laser and engine status provided this system is not already destroyed. If the laser or engine status exceeds 10 units, the status is set to 10. | |
| 4520 | Evaluate Altera Damage | This subroutine is called after damage is distributed to Altera. The purpose is to report if any energy banks are at a critical level or have been destroyed.<br><br>If two energy banks are destroyed, the program will branch to a "You Lose" message in the End Section. | 3660 3730 3850 |