

Foreword

Thank you for purchasing Planet X3 for MS-DOS computers. This project was started in November 2017, immediately following the release of Planet X2 for the Commodore 64. At the time, I had no idea how to program assembly language on x86 microprocessors, having only really experienced the 6502 up to that point. I didn't even know what assembler to use, or even what hardware I wanted to support. It took about a month just to try out different assemblers and read some books on X86 assembly language.

Fast forward to 14 months later and the game is finally complete. I faced many challenges along the way. At several points I had bugs that I considered almost unfixable because they seemed random and impossible to reproduce. This included things like the game periodically crashing for no reason. There were times I wanted to give up. In fact, I actually went nearly a month at one point without writing any code at all. However, patience and persistence does eventually pay off.

The game did evolve quite a bit from my original plan, which was just a straight port of Planet X2. Once I realized how much extra RAM and code-space I had available I started adding more and more features. Some of these features were things I had originally wanted to put in Planet X2 but couldn't due to RAM limitations. However, as development went on I kept thinking of more and more things to add to make the game more complex.

However, that complexity did come at a price. For one thing, it made the entire project take longer to code. But also, I was determined that it run on a 4.77 Mhz IBM XT, so that all MS-DOS computers could run this game. The more complex it became, I started having difficulty keeping it running properly on the slower machines. So eventually, I ended up having to again leave off features I wanted to add, simply to keep it running on old hardware.

Planet X3 was also more of a team effort than Planet X2, which was done almost exclusively by me. And so, I want to talk a little about the rest of the team and what was involved.

Anders Enger Jensen, whom I had worked with on Planet X2, was the first person to officially add to the team. Much like before, his jobs would be writing musical score, laying out the artwork for the box, working with the box companies, creating an audio-cassette for inclusion in the box, getting that designed and manufactured, and designing the user manual. Anders does a great job on any task he puts his hands on.

Renaud Scheidt was the second person I hired for help with this game. I had put out a help-wanted ad for an artist. I had about 30 people apply for the job. Naturally, I asked for sample work. Most of the samples weren't bad, but weren't any better than what I could do myself. In the end it came down to about 5 possible candidates that stood out as very talented. I think the reason I went with Renaud was not only because his sample artwork was amazing, but he seemed really enthusiastic about getting started. Renaud has had to draw 256 tiles in 3 different environments (Grassy, Desert, and Snowy) on 3 different video formats (CGA, Tandy, and VGA) thus working out to $256 \times 3 \times 3 = 2,304$ tiles. Plus, he had to tweak the tiles for CGA composite and 2-color modes as well. And don't forget the menu, introduction scenery, and the box artwork. There's no question that this game is better as a result of his magic.

Alex "Shiru" Semenov was the 3rd recruit for this game. I had put out another help-wanted ad for a programmer to write the sound and music routines. It wasn't that I couldn't write them myself, but I figured it would save time to have somebody else do that, since it could be written mostly independent of the game and then added in later. I had about 20 applicants for this job. Most weren't qualified to do the work, since it require writing the player code in assembly language. I had narrowed the list down to about 3 candidates, but when Alex sent me some examples of music code he'd already written for other projects, I knew he was the one. Alex designed a 3-voice tracker that works with Adlib, Tandy, and PC-Speaker, exactly to my specifications. Once I heard his demo song I knew that hiring him was the right

choice. It was so good, I realized there was no way I could have written such a good player code myself. Thus, his addition to the team not only saved time, but actually improved the final product!



(Noah Aman)

Noah Aman was the 4th person to add to the team. Anders was having trouble getting any music written for the tracker, partly because he was overwhelmed with work on other projects, but also he wasn't accustomed to writing music in such environments. Writing music on vintage platforms is surprisingly challenging because of the 3-voice limitation on the Tandy chip, and the 1-voice on the PC-Speaker. So he suggested we bring on Noah to do some of the tracker work and possibly create some of his own tunes as well. While I was skeptical of bringing on another musician, after Noah started sending me some of his work, I knew it was a good choice. Most of the awesome music you hear during the gameplay is Noah's handywork.

Jim Leonard was the final recruit as an expert programmer on MS-DOS compatible computers, having impressed the world with his CGA demo called "8088 MPH." Originally, Jim was instrumental in helping me learn how assembly works on the 8088 processor, and also in picking out an assembler to use. However, at that point he was just helping out because he's such a nice guy. But when I started to run into some problems I didn't know how to fix, Jim stepped in and actually started contributing some code. He was instrumental in re-writing some of my screen-drawing routines. While my routines worked, his worked about twice as fast, thus making it possible to run the game on a 4.77 Mhz XT. However, he was also instrumental in getting the interrupt routine code working stable and also integrating Alex's sound routines into the game, among other things.

And so a big thank you to the entire team. Bringing a game like this to fruition is a big job. I think a lot bigger than most people realize. And while this game is somewhat profitable, every person involved has done this in their spare time and because of their love for vintage computer platforms.

-David Murray

Design Concept: David Murray
Coding: David Murray, Jim Leonard, Alex Semenov
Artwork & Graphics: Renaud Scheidt
Music: Anders Enger Jensen, Noah Aman, Alexander Semenov
User Manual: David Murray and Anders Enger Jensen
Beta Testers: ??? (Do you have a Lead tester like the old companies did?)

Contact Information:

8-Bit Productions, LLC.
www.the8bitguy.com
dfwgreencars@gmail.com

Early Development

When I announced the intention to develop the game for MS-DOS, I had put out a request for art submissions and design concepts. I told the artists to restrict themselves to either CGA or EGA artwork, since at the time I believed those would be the two modes I'd support. I've included some of the more interesting submissions here for you to take a look at.



This first submission was from Anders Enger Jensen, and the concept was shown in CGA, EGA, and VGA modes. It shows the a mockup of what he envisioned the entire game to look like.



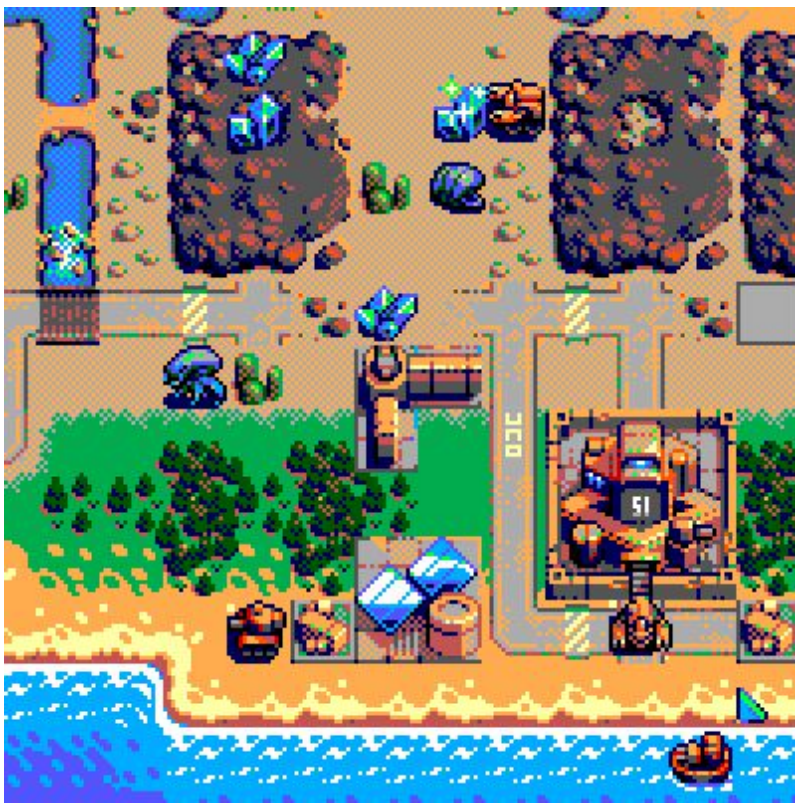
This concept art was sent in by Dave Flynn. It was designed for 16-color EGA and shows several different tiles and buildings.



This 16-color EGA concept was sent in by Igor Udushlivy. He based the entire screen mockup on the layout of Planet X2.



These are two more complete concepts sent in by Patryk Zieliński and it is based on 4-color CGA and 16-color EGA.



And last by not least, this 16-color EGA artwork was submitted by Reno.



Before selecting an artist, I needed some tiles to work with and so Anders Enger Jensen and I worked closely together on the 16-color CGA-Composite mode just so we would have some tiles to work with. This is what the game looked like in the very early days. In fact, we continued to use this artwork for composite mode even after Renaud was hired because I had only hired him to design the 4-color CGA and the VGA artwork. However, at some point he decided he wanted to do all of the art in the game so that all modes would look consistent with his style. So these graphics were eventually scrapped. And that's not a bad thing, as Renaud's artwork looks much better!

Technical Information

The game is written in 100% x86 assembly language, specifically targeting the 8088. After all, if I used any 286 or 386 instructions, then the game would not run on an XT anymore. Many people ask why I didn't use another language, such as C, for example. Well, I could have. But the end result would have been a game that probably would require 25% to 50% more code space, and would have probably run at half the speed. This would have been fine if my goal was to have it played on 386 or 486 computers. But, my target was the 4.77 Mhz IBM XT and compatibles. I think a lot of people don't realize exactly how slow a 4.77 Mhz computer is.

When I started the project, I had these specific computers in mind.



IBM 5150
Tandy 1000
IBM 5155
Compaq Portable

These systems all had 4.77 Mhz processors, tended to have around 256K of RAM, relied primarily on floppy disks, used CGA graphics, and relied on the 1-voice internal speaker for sound. Well, except for the Tandy which had some extra 16-color modes and a 3-voice sound system. I figured if I could get Planet X2 working on the C64 with even less RAM and speed than these, surely I could get my new game to run on these systems.

I grew up around these systems. Even though I had mostly Commodore stuff in the early days, many of my friends had IBM or compatible computers, including the Compaq Portable and Tandy 1000. So, I was quite familiar with them. I also know that very few MS-DOS games are released these days, and the ones that are tend to be targeted towards the later versions with 486 processors and VGA graphics. So, I felt it was time to make a game that would give people a reason to bring out their early MS-DOS machines again!

Graphics Modes - The why and how!

One other personal goal I had was to exploit the seldom-used CGA Composite mode. CGA color palettes worked very different depending on whether you were connecting to a composite or an RGBI display. Composite displays offered more color, where RGBI offered a crisp image. Which meant that early gamers on IBM compatible machines often had to decide between crisp text or extra colors, unless they had two monitors sitting around or possibly a monitor that had dual inputs. It seems that RGBI won out quickly, thus meaning most games were designed to run in the 4-color RGBI mode of CGA. In fact, after a few years most people completely forgot that CGA was ever designed to do anything else. However, most of these early machines still have the composite video connection on the back.



When I discovered that programming for composite mode was actually really simple, I immediately decided to add support for this. Below you can see two screen shots. The first, is the regular 4-color CGA mode, which you may notice has more detail because it runs in a resolution of 320x200. The next is the 16 color composite mode. It may be more colorful, but it runs in a resolution of 160x200 so much detail is sacrificed. Regardless, most people will agree the trade-off is worth it. And it is great to see composite mode being used again!



(4-color CGA vs. 16-Color CGA Composite)

Then of course, there were the Tandy machines. They are backwards compatible with standard 4-color CGA. But, they also had their own 16-color mode that ran at 320x200. I didn't think I would be able to make this mode work because it would require designing yet another tile set, but more importantly the CPU already struggled to re-draw the screen fast enough. The Tandy mode would actually have twice as much data to transfer, and thus not be possible except maybe on the later models with faster CPUs. But, when I was reading up on how Tandy graphics work, I found that there was also a low-resolution Tandy mode that ran at 160x200, just like CGA composite mode. While the palette was different, I realized the CGA composite graphics could be easily converted to work on Tandy mode, and there would be no speed difference. So, shortly after, Tandy mode was added.



(16-Color CGA Composite vs. 16-Color Tandy Low-Res)

While it isn't as sharp of graphics as most Tandy games would have due to using the seldom-used low-res mode, it was easy to add and fast enough that the game still runs at full speed. The CGA composite mode is on the left, Tandy mode is on the right. You can see the colors are slightly different and the Tandy mode is very sharp because it runs on an RGBI monitor.

Honestly, I was content with the 3 video modes I had already incorporated. And for the first 7 or 8 months of development, that is all the game supported. But as news spread about the game I was making, more and more people demanded that 256-color VGA graphics be supported. And since I wanted to do a kickstarter to help fund this game, I knew that I needed to try to please the masses and not just myself. So, grudgingly I added support for a VGA version and Renaud drew up some new graphics. Of course, VGA requires 4 times the amount of data to be sent to the video display, meaning there's no way a 4.77 Mhz XT could operate in this mode. Which is fine, since most VGA machines were 286 or better anyway. So, here's how VGA looks as compared to the Tandy mode.



(16-Color Tandy Low-Res vs. 256-Color VGA)

And so VGA presented some new challenges as well. In fact, it looked “too good.” With the older video modes, there were several tricks I had been able to get away with. One issue was transparency. Since the tiles were so low-res and chunky on the CGA system I didn't really have to worry about blending a unit with the tile it was sitting on. For example, a tank sitting on grass. But with the extra detail on VGA, it just didn't look right if I didn't add transparency. That required a lot of extra code and CPU time to draw.

Of course, I've received numerous requests to support EGA graphics, the hi-res Tandy mode, CGA Plantronics mode, Hercules graphics, and even some other really obscure modes. The reality is, there just aren't enough people going to be using these to justify the extra work. But maybe someday I will consider an update that adds some more modes.

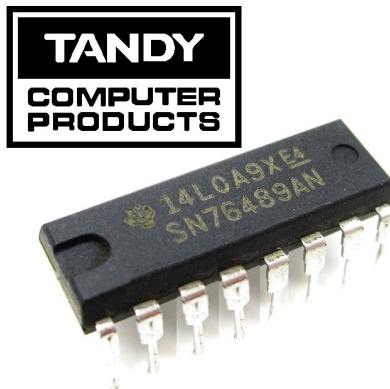
And the sad part is, I suspect 90% of the people playing this game will wind up using VGA mode, and yet it was actually always intended to run on CGA only. In fact, even after the VGA version was created, all development on the game continued to be done primarily on the CGA version first and verified to work. Once the code was working correctly there, I would port the new feature to the VGA version. So, I can only ask that when you play the game, you at least play it in CGA mode a few times, because that is what I always envisioned!

Sound Modes - The Why and How.

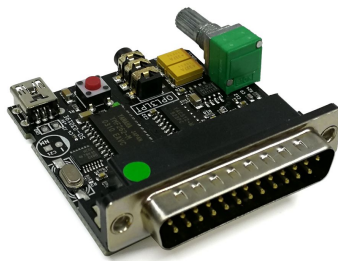


(PC-Speaker)

Obviously, because I wanted to support the original IBM XT class machines, I knew I needed to support the PC Speaker. That is, the little beeper speaker that sits inside of these machines. It's not known for making great music. There are many different software techniques for generating music and sound effects on this speaker. And there are trade-offs for each method. Since there is little to no hardware for the sound system, most everything is done by the CPU. And so, with every method there is a trade-off between quality of sound and the amount of CPU speed and RAM required to make it happen. Although Alex was hired to write the sound routines, we chatted back and forth quite a bit before he started work. We decided that we would use a system where only one tone is generated at a time, but can quickly be alternated between different parts of a song or sound effect. The CPU would be constantly deciding which tone would take priority. This can create the illusion of having 2 or 3 voices. This is actually quite challenging to design for, for both for the programmer and the musician writing the song. But in the end, it worked out really well. And I would say that Planet X3 ranks among the best games in its ability to produce sound and music on the Internal speaker.

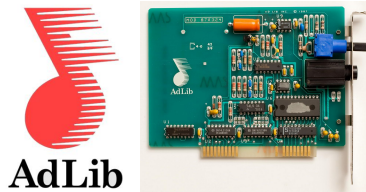


Another decision that was made early on, and this was my decision alone, was to limit the music to 3 voices. Many people were outraged by this since the Ad-Lib can do 9 voices. However, if a musician designs a song for 9 voices, then it is difficult to get that song to play on something with only 3 voices, like the Tandy sound system, much less 1 voice like the PC-Speaker. I actually told my musicians to write the music for the 3-Voice Tandy sound system first. And then they could enhance it some for the Ad-Lib. Since the code only supports 3 voices, the only difference in sound would be the wider range of notes and waveforms that the Ad-Lib can support. So the Ad-Lib version does sound better, but it doesn't have any more parts to the song than the Tandy. Then, of course, they had to reduce the song to work on the PC-Speaker.



(OPL3LPT)

Another piece of hardware I wanted to support was the OPL2LPT (now being called the OPL3LPT) which is basically an Ad-Lib card that was recently designed in 2016 by Serdashop that runs on the parallel port. This is very handy for laptops or other computers that have no internal slots for a sound card. Normally, this requires a TSR driver to be loaded before the game which will redirect the data out the parallel port. However, the TSR requires a 386 or better CPU to function. But, it occurred to me that it would be easy to support this product directly in Planet X3, so that you could use the OPL2LPT on an XT or 286 laptop since no TSR driver would be required.



And so, I can honestly say that I'm very proud of the Tandy music we have and feel it is among the best music you'll find on any Tandy game. And while the Ad-Lib is only using three voices, it doesn't sound bad at all. In fact, I'd say it still sounds better than 75% of MS-DOS games that support Ad-Lib. And the reason for this is that most MS-DOS games tended to write their music for the Roland MT-32 and then use software to emulate a MIDI system on the Ad-Lib. Meaning, the music wasn't designed natively for the Ad-Lib. And even though they had access to all 9 voices, they didn't do any tricks like changing a voice to something else in the middle of a song to create the illusion of more than 3 voices. And so, our 3-voice system actually sounds better than many of the 9-voice players used back in the day.

The sound effects are only produced by the PC-Speaker. Of course, if that's all your machine has, then no problem. This works out great on the Tandy machines since that sound is mixed with the 3 channel Tandy sound before it makes it to the speaker or audio output. Thus, it acts like a 4th voice. A similar situation exists for laptops that had an integrated Ad-Lib, since it all comes out of the same speaker. This is also true with emulators such as DOSBOX. The only time this may be awkward is when you have a desktop computer with an Ad-Lib or SoundBlaster card attached to external speakers or headphones. In this case, the sound effects will still be coming from the speaker inside the computer.

The Soundtrack Cassette

Say something here about the cassette.. *[The cassette will contain all the MT-32 music and Adlib if not also Tandy, but at least the first two. Depends on how much you want to spend on it. It will most likely be 2x30 min to cater most of the songs in their desired forms]*

Back Story

Show the VGA graphics for the introduction story with some explanation.

System Requirements / Installation

CGA/Tandy/PCjr MODE

- CPU: 8088 @ 4.77 Mhz, recommended 8 Mhz
- RAM: 256K
- Storage: 360K floppy disk, hard drive recommended

VGA MODE / MCGA MODE

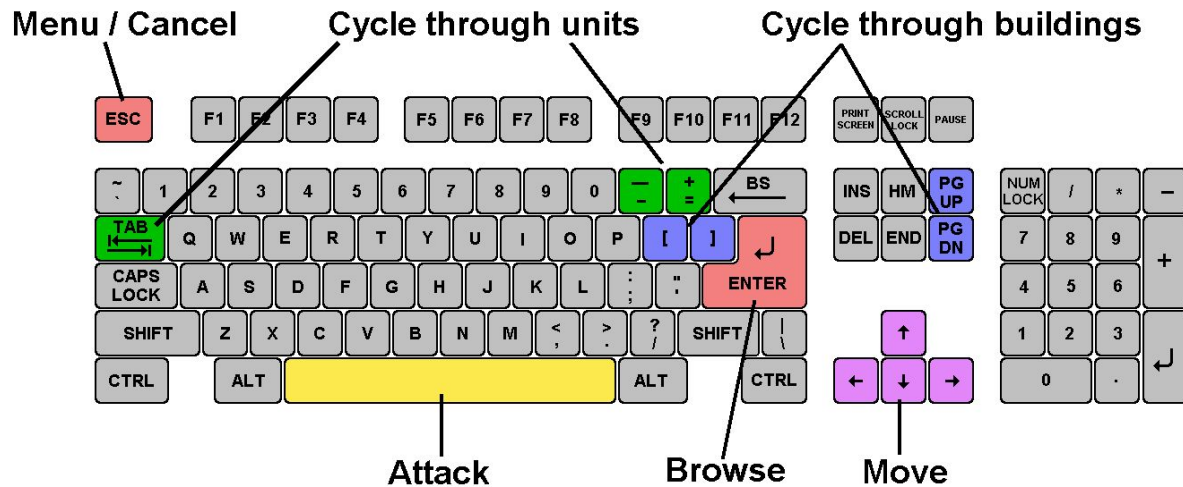
- CPU: 8088 @ 8 Mhz, recommended 286 @ 12 Mhz
- RAM: 384K
- Storage: 720K floppy disk, hard drive recommended

Sound Systems supported

- Internal PC Speaker
- Tandy / PCjr 3-Voice
- Ad-Lib, Sound-Blaster, or compatible.
- OPL2LPT / OPL3LPT on parallel port (for Ad-Lib sound)

Say something about how to install and run the game from floppy and hard drive.

How to play



Although there are many other keys to learn, these are the main keys you need to operate the game. Some keys do the exact same thing, such as the bracket keys or page up/page down keys. This will mostly be your preference, and possibly what sort of keyboard your computer has.



At the top of the screen you'll find three boxes displaying your current resources, which include minerals, gas, and energy. You will need to collect more of these, which we'll get into later. The maximum number you can store on any of these is 255. Next to the resources is a running clock showing how long the game has been going.

On the bottom you'll find your current coordinates. This is very important for knowing where your units are, especially in relation to the enemy bases. More on this later. The bottom three windows are there to give you different types of information.

Switching between units

In this game, you will be controlling dozens of different units, so it's important to know how to switch between them. There is more than one way to accomplish it, and you should learn all of the ways.

- **Browse** - If the unit you want to control is visible on the screen, you can press return to enter *browse mode*. At this point, a small box-like cursor will appear on the screen. Use the arrow keys to move this box around until it is positioned on the unit or building you wish to select, then press enter again. You can even select enemy units, but you can only look at their status, not control them.
- **Cycle Units** - For things like tanks, builders, and frigates. If the unit you want is not currently visible and you don't feel like driving to the unit, you can press the TAB key to cycle to the next unit, wherever it may be. The cycle is based on unit number, so sometimes it may seem there is no rhyme or reason as to which unit it goes to next. You can also use the plus and minus keys on the keyboard to move forward and backwards through your units. With practice, this is the fastest way to switch between units.
- **Cycle Buildings** - For things like factories, smelters, and radar stations. If the building you want is not currently visible, you may press page-up/page-down to cycle back and forth between them, very much like the unit cycle. You can also use the bracket keys [and] to accomplish the same thing. Solar panels are purposely skipped during the cycle because there are often numerous panels and you can't actually control them in anyway. If you want to select a solar panel you must use the browse feature.
- **Hot Keys** - You can use the number row 0-9 at the top of your keyboard to assign specific units or buildings that you need to access more quickly. Hold down SHIFT to assign a unit to a key, then you can use that key to switch to that unit quickly. This is especially useful in battles.

There are essentially two ways to fight them. One way is with your tanks and boats onboard weapons. You can also attempt to use the self-destruct on a tank to wipe out a large chunk of a base, but this is tricky and you'll need practice to that. The sure-fire reliable way to take out a base is with a missile attack. But for that, you need the exact coordinates. Read up on *missile silos* in the chapter on buildings.

If you destroy part of an alien base, they'll immediately start to rebuild. So, it's important once you start attacking them, that you are ready with enough tanks or missiles to follow through until the entire base is gone. Otherwise, you'll have wasted the effort.



The enemy base can have quite formidable defenses, especially after the game has been going long enough for him to completely build out everything. Read up more on the *enemy base* to learn about what each building does and strategies for attacking it.

You must destroy all Protoid buildings, including sentry pods in order to win the game. It is not necessary to destroy all Protoid ground units.

Buildings

Headquarters



Cost	
Minerals	20
Gas	5
Energy	5
Health	200

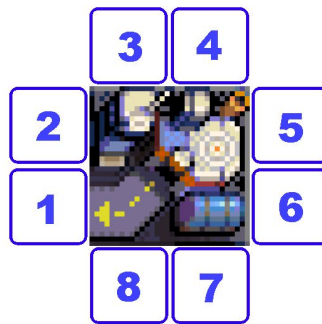
The Headquarters is the building you always start. It's actually part of the ship that took you to the planet, which is now permanently landed on the surface. It's the only building on the map. You can even build more of them. However, it actually doesn't do much for you, and if it is destroyed you probably won't miss it. One feature it does have, however, is that if you select it and look at the status screen, it will tell you how many units and buildings you currently have.

Smelter



Cost	
Minerals	35
Gas	10
Energy	10
Health	
	145

The smelter is an essential building. You will have to build one early in the game. It's purpose is to process minerals. Ideally, you would want to build it near where minerals exist. But sometimes, it may be difficult to protect it from attack there. So you'll have to make the call. You will need to use your builder to pick up rocks and crystals and place them around the edges of the smelter. It will automatically pick up the minerals and start processing them. Once it is finished processing one item, it will pick up the next item. There is an actual order for where it looks for minerals, starting with the bottom-left by the entry ramp, and working its way around clockwise. (see diagram) So, if you have more valuable minerals like crystals, you may wish to place them ahead of the queue for something like a small rock, which takes longer to process.



The smelter is completely automated, so there is no need to interact with it, other than to see the status screen. However, you will need to check up on it every so often to see if it is running low on minerals and bring it more, if needed. You may build more than one smelter for faster mineral processing, or if there are minerals in different places on the map.

If your mineral count reaches the maximum of 255, the smelter will stop production until you use some minerals. So if you notice the progress bar in the status screen appears to be frozen, that's probably the reason.

TIP: Besides smelting rocks and crystals, you can pick up your dead tanks and builders and recycle them at the smelter for some extra minerals!

Refinery



Cost	
Minerals	16
Gas	0
Energy	3
Health	
	125

The refinery is for collecting and processing methane gas. It must be built on top of a gas vent, sometimes referred to as a volcano in the game. The station is completely automated and there is nothing you need to do once it is built. It will automatically start contributing to your gas reserves. You can build more of them, and it is highly recommended if you can find more sources of gas.

Power Station / Solar Panels



	Cost	Power Station	Solar Panel
Minerals		25	5
Gas		5	0
Energy		0	0
Health		150	75

Power stations and solar panels must be built in order to supply you with energy. You technically only need one power station, and you can connect many solar panels to it. Either one by themselves is useless. So, if you wish to build solar panels in a different part of the map, you will need a second power station to go with them. You may daisy chain solar panels to each other both vertically and horizontally. If you check the status screen they should say which power station they are connected to. If you select the power station it should show how many solar panels are connected. If you do not align the panels exactly, they will not connect, and thus will not generate power.



(incorrect placement / correct placement)

Take these two examples. In the first image, only 2 of the panels will actually be working, because the other two are misaligned. In the second photo, all 4 panels will contribute power. You can attach panels to any of the 4 sides of the power station, or any 4 sides of another solar panel.

Factory

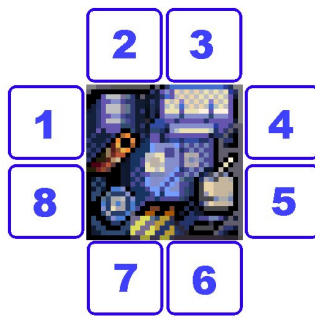


Cost	
Minerals	50
Gas	10
Energy	10
Health	175

The Factory is an essential building where you can build various units that you can use to build, explore, defend, and attack:

- **Builders** - For making things and gathering resources.
- **Tanks** - For exploration and attacking.
- **Heavy Tanks / Sentry Tank** - For defending your base.
- **Frigate** - For exploring rivers and oceans, and even attacking bases.

If you want to make frigates (boats) then you need to build the factory next to water, otherwise the option will not show up in the menu. You can build more than one factory if you need to build things faster. The factory will automatically place your new unit next to it. If the area is blocked, it will pick the next available slot. See the diagram for the order in which it will place items. If all 8 exits are blocked, your unit will not be delivered and you will lose the resources you spent making it. So, be sure not to block your factory!



Radar Station



Cost	Station	Scan
Minerals	32	0
Gas	10	0
Energy	12	30

Health	100
--------	-----

The radar station is an optional building that can assist you in locating the enemy bases, additional minerals, additional gas vents, water, etc. It offers 4 types of scans:

- **Metallic scan** - This will show both player and enemy bases and ground units.
- **Hydroscan** - This will show any water formations on the map such as rivers and oceans.
- **Thermal Scan** - This will show strong heat signatures on the map, such as lava and volcanoes/gas-vents.
- **Oscillating scan** - This type of scan searches for crystalline objects that resonate, such as rocks, mineral resource deposits, and even your own walls, since they are made from local minerals.

All scans will show a white dot on the map. This dot represents the radar station itself so that you will know the orientation of the other objects shown. This is an example of a hydro-scan showing the river formation on this map.



(Example of a Hydroscan)

When the map is shown, just press any key to return to playing. Keep in mind, even while the map is shown on screen, the game is still playing in the background.

Missile Silo



Cost	Silo	Missile
Minerals	48	16
Gas	10	32
Energy	10	32

Health	220
Damage	255

The missile silo allows you to construct and launch missiles, which can finish off an enemy base from a distance. You will need to set the coordinates (see the status screen for the current coordinates) which can be adjusted using the following keys:

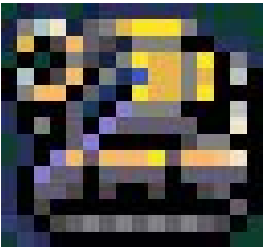
X / Shift-X : Sets the horizontal coordinates.

Y / Shift-Y : Sets the vertical coordinates.

You'll need to explore the map and locate the enemy base to get its exact coordinates. Before you can launch a missile, you'll need to press B to build a missile. Once built, the status screen will say "ARMED." At this point you can launch. You will hear a sound and the screen will flash white for a second from the blast. You can build more than one missile silo, which is actually recommended. That way, when you discover the base, you can offset the coordinates slightly of each silo and launch multiple missiles at the same time, thus obliterating the base.

Units

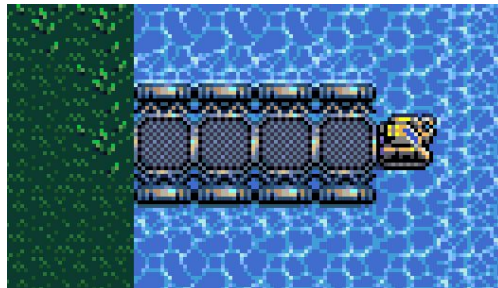
Builder



Cost	
Minerals	16
Gas	2
Energy	1
Health	20

The builder is your most important unit. Without it, you can't do much of anything. It performs more features than any other unit on the map.

- **Pick Up / Drop Off** - You can pick up many things on the map and carry them around. Of course, while the builder is carrying something, it can't do any other function. This is useful for carrying minerals to your smelter, for example. But you'll find other items on the map you can pick up which serve no purpose.
- **Walls** - You can build walls, although they must be built one piece at a time.
- **Bridges** - If you need to get across the water, you can build a bridge. It will automatically select the correct direction and it will keep building until it reaches land, reaches the edge of the map, or runs out of resources. If a bridge is interrupted, you can start it up again when you are ready.
- **Bulldoze** - If there is something in your way, you can bulldoze it. Examples are trees, small rocks, dead aliens, etc. Sometimes it is even useful to bulldoze spots as markers along a path just so you can find your way back to a spot.
- **Build Buildings** - And of course, you can build a variety of buildings, which were described earlier.

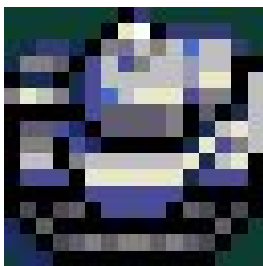


(builder building a bridge)

Tip: Bridges work best when building from a straight boundary between water and land. The builder guesses the direction based on the orientation of the water and land. But if you build on a corner or diagonal piece of land there is a 50% chance the builder will guess the direction wrong.

Tip: You can use rocks as walls, or even as a doorway to your permanent wall.

Tank



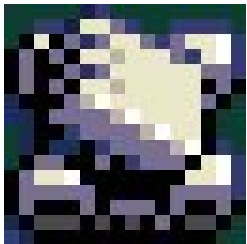
Cost	
Minerals	30
Gas	5
Energy	3
Health	85
Damage	10 / 25

The Tank is your unit for exploring and combat. It is the fastest unit, capable of withstanding more damage than a builder, and of course it can shoot things. There are 3 different ways you can use the tank for assault.

- **Manual Targeting** - By pressing M you can manually select exactly what you want the tank to shoot. You can shoot at trees, walls, and rocks if you need them out of your way.
- **Automatic Targeting** - By pressing space it will automatically fire at any enemy in range. If no enemy units are within range, it will default back to manual targeting.
- **Self Destruct** - By overloading the core, it can purposefully destroy itself and a large area around it. It does take some time for the overload to occur, so it takes some practice to be able to use this feature against an enemy base.

TIP: The Tank does two and a half times more damage when using manual targeting.

Heavy Tank



Cost	
Minerals	40
Gas	10
Energy	5
Health	115
Damage	20 / 50

The Heavy Tank has heavier armor than the regular tank, but it is slow and mostly meant for protecting the base. It has two modes of operation:

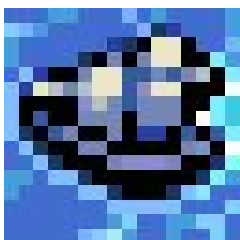
- **Assault Mode** - In this mode, you can drive it around and manually target and shoot things. The main turret is very powerful, but it can be hard to attack moving enemy units with it. It does work well against the enemy bases, however.
- **Sentry Mode** - In this mode, it converts to a stationary state and begins to search for any enemy units nearby. If anything gets within range, it will automatically fire at them. It is good to position 5 to 10 sentry tanks around your base.



(in sentry mode)

TIP: The Heavy Tank does two and a half times as much damage when using assault mode.

Frigate

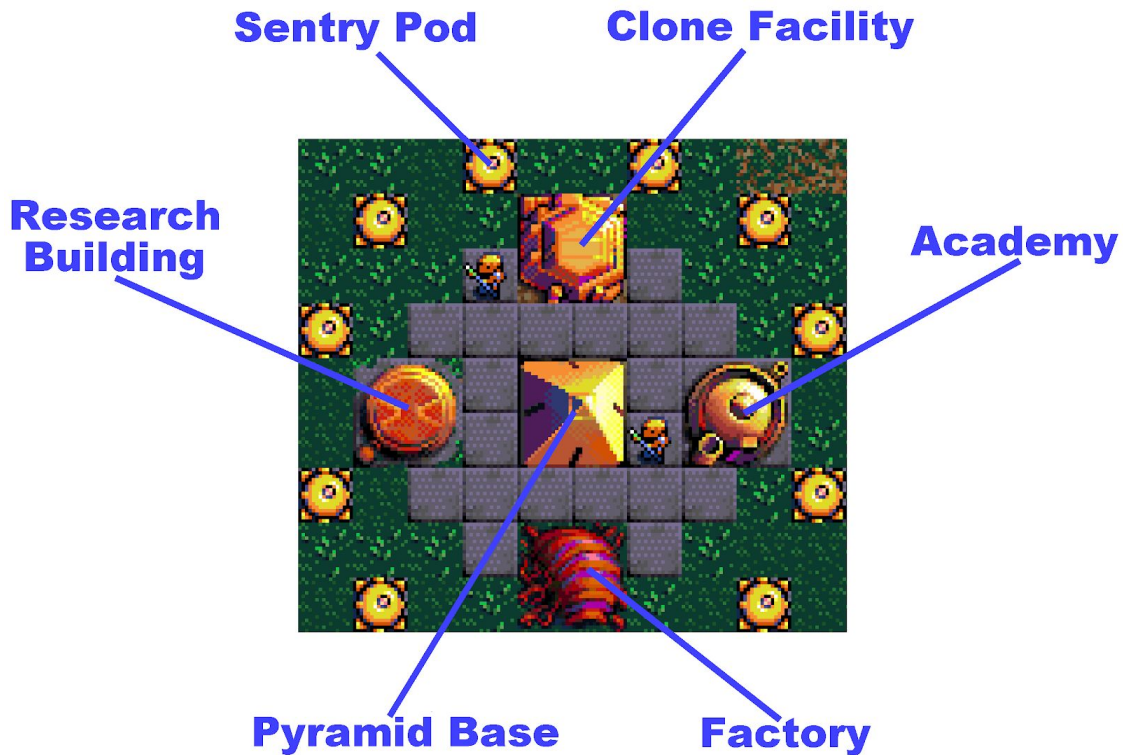


Cost	
Minerals	30
Gas	5
Energy	5
Health	125
Damage	25

The Frigate is a water-based assault vehicle. In order to make one, you must position your factory at the edge of water. The frigate only has manual targeting mode, but on the bright side, protoids move slowly in the water. The Frigate is good for exploration and in some cases you can even use it to attack an enemy base. On some maps, the enemy bases themselves are in the water and thus the only way to reach them is with a frigate.

The Enemy Base

The Protoids have a standard base design. The base starts out as a single pyramid. It may or may not have concrete around it, depending on the map. Over time, it will build out more and more buildings.



- **The Pyramid** - This is the core base. It's the only building the Protoids start with. It is responsible for building all of the other buildings, so if you destroy it before other buildings are made, then that base is finished.
- **Sentry Pods** - These are the first line of defense around the pyramid and the first thing that is built after the game begins. The sentry pods will automatically attack any human unit that gets close enough. The pods are built one at a time, so it takes several minutes before they are all built out, meaning if you find the base early it may be defenseless. Depending on the difficulty level you select, will affect how many sentry pods will be built around each base.
 - EASY - 4 Sentry Pods Max
 - NORMAL - 8 Sentry Pods Max
 - HARD - 12 Sentry Pods Max
- **Clone Facility** - This is where scout clones are replicated. This is the first building created after the sentry pod defenses are finished being built. No ground units will be created until at least this building is finished.
- **Academy** - This is the second building that will be created. This is where cloned soldiers are replicated. These soldiers are faster, stronger, and can shoot from a distance.
- **Factory** - This is where the Protoid Tanks are manufactured
- **Research Building** - This is where the Protoids do research and strategic planning. It's existence really doesn't affect you much as a player, except that it is one more building you must destroy to defeat the base. If it remains, eventually the whole base will be rebuilt.

In order to understand the Protoid base, it is important to know that all construction of Protoid buildings occurs from the pyramid itself, this includes the sentry pods, clone facility, academy, etc. However, all of the other buildings (except the sentry pods) monitor the pyramid. If the pyramid is destroyed, the other buildings will initiate a new construction of a replacement pyramid. That's why you must finish off all of the buildings or they will rebuild the entire base eventually.

Although the sentry pods will not initiate new construction, they must still eventually be destroyed in order to win the game.

The speed at which the Protoids build out the base is determined somewhat by which difficulty level you select. If you hit the base earlier in the game, it will be easier to destroy.

Enemy Units

Clone Scouts



Health	20
Damage	5

The basic clone scout will be the first thing to attack you. They do require being in direct contact with whatever they are attacking. They are easily blocked by walls and other obstacles.

Clone Soldiers



Health	30
Damage	10

The soldiers are a little faster, a little stronger, and their weapon can fire at a distance. However, their target distance is only 3 spaces, so your sentry tanks have a longer range.

Protoid Tanks



-Not yet implemented.

Error Messages

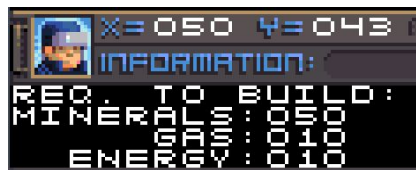
Below are all of the error messages that you may encounter while playing the game. Some are obvious, but some may require further explanation.

CONSTRUCTION EXIT BLOCKED - It is possible to trap your builder like in the example shown. When constructing a building, the builder has the option to exit to the West, or to the South. If neither are available, then it will be trapped. The only solution is to use a tank to destroy either the building or the builder. Or, you could just leave it there and make another builder at the factory.



(trapped builder)

REQUIRED TO BUILD: - If you see this error, with a listing below of minerals, gas, and energy it means you tried to build something, but didn't have enough of at least one resource. The numbers shown are the amounts of each resource that you actually needed to create the building. You'll have to look to see which one you were short on. This was a consolation to the fact there wasn't enough screen real-estate to show the required amounts before building something. However, with the addition of this manual you can obviously look up the amounts beforehand.



CANNOT MOVE THIS OBJECT - This one is fairly self-explanatory. You've attempted to pick something up with your builder that is not allowed.

SOMETHING IS IN THE WAY - Most likely this is because you tried to drop something from your builder in an area where something already exists.

MAX UNIT LIMIT REACHED - This error will occur if you attempt to build more than 20 ground/water units. Also, you will not get your resources back for the unit you tried to make. This was a compromise made due to issues with not wanting the player to have multiple factories and be allowed to queue up many units at once when 19 units exist on the map. So be warned, it will allow you to build it, but it will never emerge from the factory. However, in testing most players never build more than 12 units anyway.

MAX BUILDING LIMIT REACHED - This is a similar error and means you've reached your limit of 44 buildings, thus you cannot build anymore. Most players rarely produce more than 20 buildings tops, so you aren't likely to see this.

FACTORY EXIT IS BLOCKED - You will see this if all 8 delivery locations around a factory are being blocked by something. You will not get your resources back if this happens, so be warned.

CANNOT BUILD THERE - You have selected to build something in a spot that is not allowed.

MUST DROP ITEM FIRST - Your builder is carrying something and you told it to build something or pick up another object.

MUST BUILD ON GAS VENTS - You have attempted to build a gas refinery on some place that isn't a gas vent.

CAN'T BULLDOZE THAT - You have attempted to bulldoze something that is not allowed.

TARGET OUT OF RANGE - You have attempted to fire at something with your tank that was out of range.

MUST BUILD ON WATER - You have attempted to build a bridge on land.

NOT IN BROWSE MODE - You have attempted to issue a command, change units, or basically just about anything while the interface is still in browse mode. So, you need to press ENTER or ESCAPE to exit browse mode, then try your command again.

NO UNITS FOUND - If by some chance all of your ground/water units are destroyed, and you attempt to use TAB or the plus/minus keys to fast-switch to another unit, this will be displayed.

Strategy and Tips

These are some tips that might come in useful as you are playing.

- Sometimes it is a problem when you build a wall around your base because then you can't get out yourself when you need to. One solution is to use a rock as a door. That way the builder can simply pick it up when you need to get out, and drop it back off when you are done.
- Once you find the enemy base, if you can browse to any object, even a sentry pod, then you can select that object and it will become the center-focus of your interface. Then, you can browse again to the pyramid to get the exact center of the base, even if your tank or builder is already dead.
- When firing off missiles, one way to make sure you eliminate the base is to fire 4 missiles. Each one can be offset by 2 in both directions. It may be helpful to have more than one missile silo. So for example, if your enemy is located at 100,100 then you could fire 4 missiles at the following locations:
 - X=098 Y=098
 - X=102 Y=098
 - X=098 Y=102
 - X=102 Y=102
- Generally you can setup sentry tanks next to water without needing a wall. The protoid units swim so slowly that the sentry tanks should get them before they come within range.
- Your sentry tanks have a longer range than the Protoid soldiers. So if you build a wall, set your sentry tanks back a few spaces. That way, they can shoot the enemy units without risk of being attacked.
- Protoid ground units aren't smart enough to find their way around large obstacles. So, for example, if there is a small path they are using to travel to your base, it may be sufficient to block that path with walls or rocks.
- Protoids have their own radar system and they know where your buildings are. However, they do not know where your ground units are unless you get really close to one of their units and they see you. So, for example, if you find a remote gas vent you want to utilize, you can build a wall around the gas vent first before building the refinery. That way, by the time the refinery is built, they will not be able to attack it.
- If you do pass near Protoid ground units, they will change course and follow you until they are able to get close enough to attack. However, you can almost always outrun them. Once you are out of range for a few seconds, they will forget about you and continue on their way.
- You can destroy a Protoid base with tanks if you have several lined up near to the base. Use the tanks one at a time and get only as close as you have to to be in range of their sentry pods. Then, attack the sentry pods one at a time. Alternatively, you can attempt to drive a tank inside the base and initiate self-destruct. The fewer sentry pods in range of the tank, the better. If you have several tanks nearby you can finish them off with 4 or 5 tanks using self-destruct. It takes practice, though. This may not be possible in "hard" mode.
- If you built a bridge in the wrong place or otherwise realize it would be helpful if the bridge could turn at a 90 degree angle, it is possible. Abort the bridge building process, then bring in a tank. Use the tank to shoot the side rails off the bridge in whichever direction you need to build. Then use the builder to start a new bridge in the required direction.
- You can build on concrete. Also you can build buildings right next to each other to save space. Just make sure there is an exit point for your factories.
- Once you find an enemy base, you can select it and assign it a hot-key, just like your own units. So if you launch a missile, for example, you may want to jump back to the base to see what happened. Of course, if the base was destroyed your hot-key won't work.
- You can assign many tanks and heavy-tanks to hot keys. Drive the tanks near the enemy base, but just out of range. By using the hot-keys, you can quickly switch to each tank, move it close to the base, and set it to sentry-mode. Thus you will have multiple units attacking the base at once.

Performance Issues

Planet X3 was designed to work on very slow computers. However, depending on your selection of audio and video, as well as how many units are active on the map, you may notice some slow-downs. There are several options provided to help speed things up.

- **Disable Sound and Music** - Actually, you have to do both. When both sound and music are disabled, the sound routines are completely bypassed, including the ones that actually check to see if a sound should be happening in the first place. That's because there is a scanning routine any time an event happens, such as an explosion, that has to check if it happened in your viewing window. This uses some CPU time.
- **Reduce Screen Width** - This option will reduce your horizontal screen size from 19 tiles to 13 tiles. This actually speeds up the screen redraw by about 50%. The game is still fully playable in this mode. In fact, when reduced, it is essentially the same play area as Planet X2 had.
- **Disable Transparency** - (VGA version only) This will save you about 20% CPU load by showing black bars around an otherwise transparent object. It does not affect gameplay other than the appearance.
- **Use the CGA version instead** - If your machine is slow, especially an XT class machine, even if you have a VGA card installed, it may be best to run the CGA version instead, as it will run much faster. VGA mode, even with transparency off, still requires about 4 times the CPU time to draw the screen.

CGA Composite Mode

There is only one way to use CGA composite mode, and that's to have an actual CGA card or a computer with CGA built onto the motherboard, but offering a composite video jack on the back. While EGA and VGA cards are "backwards compatible" with CGA, that isn't entirely true because they do not support or even emulate the composite mode. Even if your VGA card has a composite port, the color encoding works entirely different, and thus you will still not be able to use it for CGA Composite.

If you start composite mode on a true IBM or at least a true IBM CGA card, the colors should look correct. However, many clone machines don't always show the correct colors on bootup. For example, the Laser XT picks one of 4 random timings on the color signal so there is only a 25% chance you'll get the correct colors on bootup. Other machines, like the Tandy 1400 laptop for example, always show the wrong colors. Planet X3 includes a work-around for this. In composite mode, the menu options **CGA PALETTE** will rotate between the 4 possible color palettes you might encounter on these machines. Unfortunately, it will only help during the game play and will not help during title or menu graphics.

DOSBOX - If you want to experience CGA composite mode in DOSBOX, you will have to configure DOSBOX as a CGA machine. You might think it is illogical since it is configured for VGA by default, and VGA is backwards compatible with CGA. But, in many ways this is correct because VGA cards still can't use the composite mode. You will need to edit the DOSBOX.CONF file and edit the machine type setting so that it looks like "MACHINE=CGA" If you want to emulate Tandy video and sound you'll need to edit it to "MACHINE=TANDY"

The **CGA PALETTE** menu option does 3 different things depending on your mode:

- CGA 4-color mode - Toggles between cool and warm CGA palettes.
- CGA Composite - Cycles between 4 possible composite color palettes.
- CGA 2-color mode - inverts the screen like a photo negative for laptops that have inverted screens.
- Tandy Mode - No effect.
- VGA mode - menu option is not even present.

Example of DOSBox config parameters to get sharp, clear graphics. Play with the settings according to your host machine and how it responds:

```
[sdl]
fullscreen=false
fulldouble=false
fullresolution=original
windowresolution=1280x1024
```

```
output=openglfb
autolock=true
sensitivity=100
waitonerror=true
priority=higher,normal
mapperfile=mapper-0.74.map
usesccodes=true
```

[dosbox]

#CGA COMPOSITE - You can add various copies of this section with headers like this that let's you easily change between the modes by removing and adding the # to the sections at your desire.

```
language=
machine=cga
captures=capture
memsize=16
```

[cpu]

```
core=normal
cputype=auto
cycles=fixed 7000
cycleup=10
cycledown=20
```

IBM PCjr

The PCjr (PC Junior) doesn't typically have enough RAM to play Planet X3. It normally includes either 64K or 128K internally. However, it is possible to upgrade it further with external and other aftermarket RAM expansions. This will allow you to not only run the game, but you'll be able to use the PCjr graphics and sound modes, which are the same as the Tandy modes. You will need at least a 512K machine in order to run the game, even though it uses the same code as the Tandy version, which only requires 256K. This has to do with the way the PCjr memory layout is handled. In fact, you will need a special driver (beyond the scope of this manual) to force programs to load into a higher memory address before games like Planet X3 will work.

Reference

	Human Units				
	Minerals	Gas	Energy	Health	Damage
Headquarters	20	5	5	200	
Power Station	25	5	0	150	
Solar Panel	5	0	0	75	
Gas Refinery	16	0	3	125	
Radar Station	32	10	12	100	
Factory	50	10	10	175	
Missile Silo	48	10	10	220	
Smelter	35	10	10	145	
Builder	16	2	1	20	
Tank	30	5	3	85	10 / 25
Heavy Tank	40	10	5	115	20 / 50
Frigate	30	5	5	125	25
Brick Wall	3	0	1		
Bridge Piece	4	0	2		
Radar Scan	0	0	30		
Missile	16	32	32		255

	Protoid Units	
	Health	Damage
Pyramid Base	250	
Clone Facility	150	
Academy	250	
Factory	175	
Research	150	
Sentry Pod	100	5
Clone Scout	20	5
Clone Warrior	30	10
Tank		

Maps

(this may be in a totally different manual)

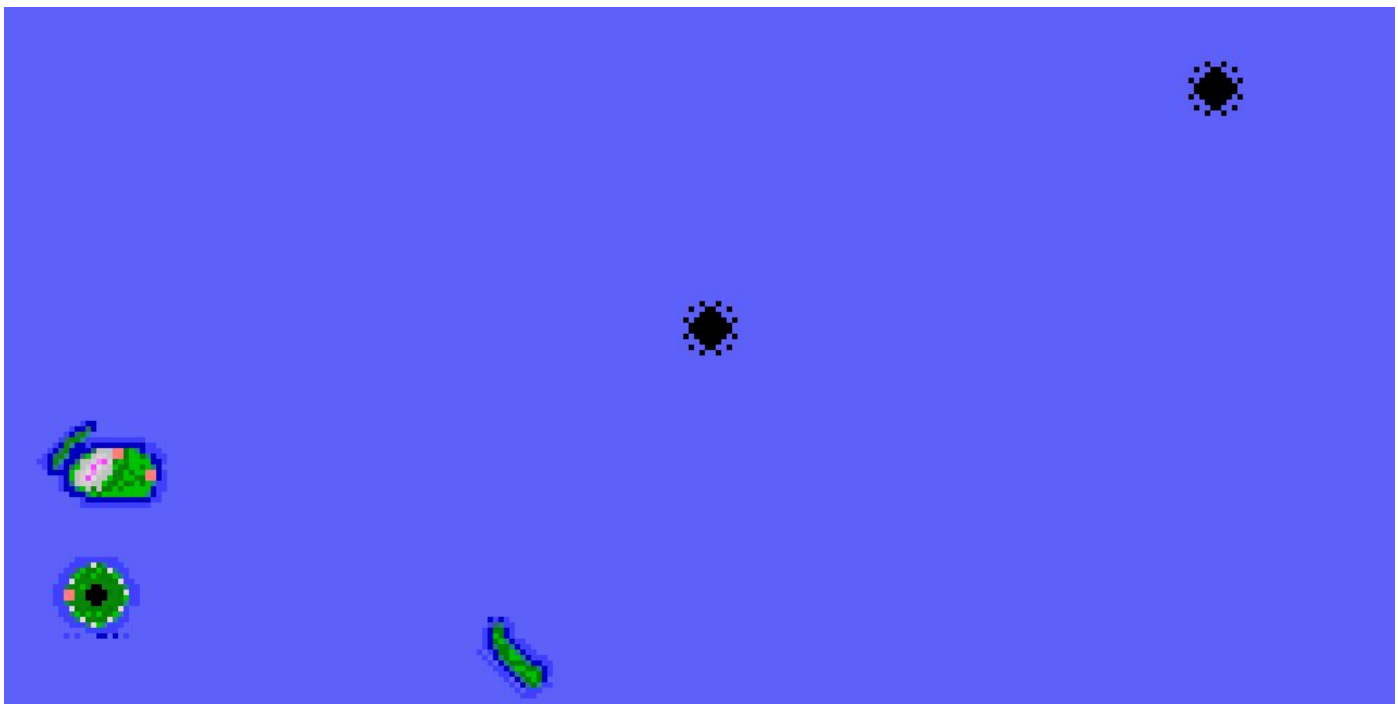
Spoiler Alert! Spoiler Alert! Spoiler Alert! Spoiler Alert! Spoiler Alert!



River Divide - By David Murray
Music - "Fight for the Future" by Noah Aman



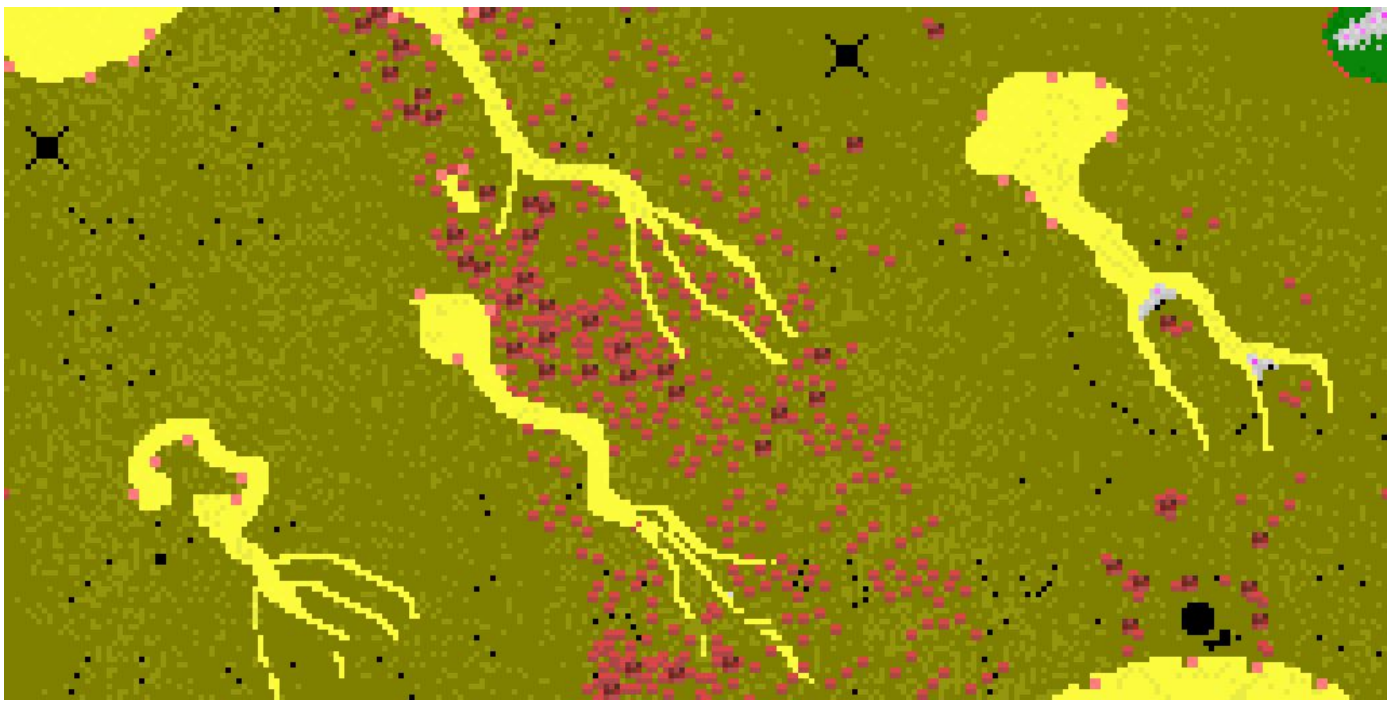
The Atoll - By David Murray
Music - «Menu titles» By Anders Enger Jensen.
Tandy/Adlib conversion: Noah Aman



Two Islands - By David Murray
Music - "Plan Ahead" by Noah Aman



The Jungle - By David Murray
Music - "Jungle Vibes" by Noah Aman



Inferno - By David Murray

Music - "Klepackin' Heat" by Noah Aman

