

Xelda  
Quest for the Golden Apple  
by  
Andrew Dansby  
along with  
Russell Glover

## Contents

[Statement of Disclaimer](#)  
[History of game production](#)  
[Open Source software statement](#)  
[Introduction](#)  
[Credits](#)  
[Game Engine](#)  
[Loading](#)  
[Control Selection](#)  
[Playing the game](#)  
[Your Character](#)  
[Other Characters](#)  
[Monsters](#)  
[Obstacles](#)  
[Objects and Doorways](#)  
[Dungeons](#)  
[Winning the game](#)  
[Epilogue](#)

## Statement of Disclaimer

This game is not intended to be a clone of Zelda, nor follows the storyline of Zelda. The sprites and title song are derivatives of the original work by Nintendo. However, if the sprites changed and the title song changed, this game in no way would actually resemble Zelda. Therefore this game should not interfere with the rights of Nintendo and their copyright on the game. This game is a top down adventure that has been presented in many other games. This game is not meant for sale. It's presented as a programming exercise for entertainment only.

## History of the game production

Zelda was originally a programming exercise within the Mojon Twins MK2 to see if a rather seamless large scale game could be programmed for the ZX Spectrum 128k machine within the MK2 framework. I wanted to make sure the game would work across all official and classic ZX Spectrum machines.

I originally started the programming exercise around November 2016 to see how I could link levels together within the MK2 framework and have variables stay global. I gave each level and jump to each level a unique identifier. From there I could walk to a specific way-point on each level and then pop to another level, retaining the global variables.

The original programming exercise was 5x5 screens within 4x4 levels, meaning each level was 5 screens wide and 5 screen tall and the game contained 4 levels wide and 4 levels tall, with 4 dungeons for 464 screens. That proved to be much too large, so I reprogrammed the script framework again.

The second programming exercise was 4x5 screens within 4x4 levels and 4 dungeons for 420 screens, that was again too large, so I scaled back again.

The third programming exercise was 4x5 screens within 3x3 levels and 4 dungeons for 260 screens, the game seem to hold up. So I programmed with that iteration for about 2 months, placing in the level designs for the first 3 levels. I then realized I needed to have a level for inside houses. This would be a level disconnected from the rest of the game, places where you could go inside to receive a quest. This would add another 20 screens to the game, for a total of 280 rooms. This memory model held up for about a month before I started adding some puzzles. I started realizing I was running out of memory with only 3 levels programmed. I had to restart the entire design process again and scale back.

The memory model I finally settled on was 4x4 screens within 3x3 levels, 3 dungeons and 1 inside houses level for 208 screens (in theory, in practice it's around 204 screens. This model stored all puzzles I wanted to throw into the game, all dialog and all logic. It is a tight squeeze to fit everything inside. All data is compressed, uncompressing on the fly for each level (this is part of the Mojon Twins Framework).

Script programming took quite some time, and was programmed and changed around for 9 months or so until it was about right. The MK2 C framework from the Mojon Twins had to be modified to take care of a few bugs. Since I am terrible at creating music, I used a procedural music creation program (CG Music) and plugged each note into WYZ tracker.

Finally came the beta test. Since at this point I have only played the game in individual sections, since playing the entire game each time I made a change or addition to the script would be too time consuming. I asked for assistance for testing, while I performed debugging. I had assistance with a few beta testers and together we were able to fix the bugs and I played through the game about 20 some odd times, repairing logic as I went and each time on fixing a bug, releasing a new beta and triggering a new replay. It currently take me about 2 ½ hours per play, so can be time consuming.

There could still be undiscovered bugs within the game, but the game is completable. I finally finished with the script on November 2017, a year after I started the original memory test.

I hope everyone enjoys the game as a lot of effort has been put into the game.

## Open Source software statement

This game is open source and freeware. The Mojon Twins MK2 engine is given freely, however any game produced using their engine cannot be sold without being agreed to from the Mojon Twins. I never intended on selling the game, only to produce a game.

## Introduction

Xelda is a tribute game, it takes it's inspiration almost directly from another famous game – Zelda. Which just so happened to be named after novelist F. Scott Fitzgerald wife Zelda Fitzgerald.

Even though the inspiration was taken from Zelda, this is not a direct copy of any of the Zelda games, it is its own entity. Spites and objects were “borrowed” from the real game, however that is an aesthetic only.

Xelda is a high adventure game, filled with puzzles, exploration, quests and fights. Your main quest is to find the Golden Apple which wards off the monsters that fill the island and terrorize it's citizens. You will have to defeat puzzles which unlock passages, acquire objects or help others to further your quest.

Due to the sheer size of the game, it was written for 128K machines only (sorry for the 48k machine owners). However, with emulation and excellent machines like the ZX Uno, everyone who wants to play the game can.

## Credits

This game is written by your humble programmer, Andy Dansby. My second computer that I owned was a Timex Sinclair 2068 which was the American Version of the ZX Spectrum. Sadly, it was mostly incompatible with the ZX Spectrum which along with poor marketing spelled the DOOM for that machine. However I you replaced the ROM, you could have a mostly compatible computer. That is what introduced me to the wonderful world of the ZX Spectrum.

This game however was not written without some support and there is credit due to others that helped me with the design.

Andrew Dansby – Programmer, level design, music, sprites and other bits and bobs.

Russell Glover – designer of level 4, the castle level, designer of most of the Graphics and my main sounding board for ideas. Russell diligently worked with me over the course of many months.

Cthonian Godkiller – for technical help with the script.

Andy Green – Created the awesome loading screen.

Casey Jones – some of the push block puzzles.

Hikoki – Provided some game ideas and was able to track down game programming information, was able to track down some obscure links for me to research. He provided a great sounding board for ideas and concepts.

The Mojon Twins for writing the MK2 engine on which this games is built on.

My Beta testers included.

Russell Glover

Baz Har Ding

Steve Jedikiah

Steven Charlesworth

Steven Brown

## Game Engine

This game was written using the MK2 engine, designed by the Mojon Twins. It's a rather versatile game engine, allowing for multiple levels, sprites and scripting.

This is my first game within the MK2 engine and the first game I have made as an adult.

## Loading

This game will only load on a 128k version of the ZX Spectrum. All 128k Machines should be supported. This is built into the Mojon Twins MK2 engine. It has been tested on 128K +2a by one of my beta testers (Steve Jedikiah). I have tested it on my ZX Uno, set as a 128K toast-rack and on a +3.

Loading on an original 128k machine, is simple, navigate to **Tape Loader** and press enter, press **play** on your cassette player. The game take around 12 minutes to load.

Loading on a ZX Uno or on an ESXDOS equipped is slightly more complicated.

On a ZX Uno

Access your bios  
(CTRL ALT BACKSPACE)

Make sure that **NMI-DivMMC** is Enabled (If you want to use NMI for the file browser)  
also make sure your **DivMMC** is Enabled

Choose the Joystick option you want, I suggest **Kempston**

Make sure your machine is set for a 128K machine and reboot your ZX Uno

Once your Uno is booted, press CTRL-ALT-F5

Navigate to the directory that you have placed the Xelda.tap

Select the file

The file, will attempt to load and then reboot the spectrum

type in **POKE 23388,16**  
type **LOAD ""**

On a physical ESXDOS interface

The process should be the same for a classic ZX Spectrum 128K with a ESXDOS compatible interface.  
Press your NMI and select Xelda.tap. The game should load on selecting it. If not you may have to

**POKE 23388,16**  
and  
**LOAD ""**

## Control Selection

On the game loading, you are presented with 4 control options

### 1 WASD

**W** is up

**A** is left

**S** is down

**D** is right

**Space** is fire/Action. The fire button will perform actions, either swinging your sword, reading signs, dialog and flipping switches.

### 2 QAOP

**Q** is up

**O** is left

**A** is down

**P** is right

**Space** is fire/Action. The fire button will perform actions, either swinging your sword, reading signs, dialog and flipping switches.

### 3 Kempston

#### 4 Sinclair

Interface 2, port 1

Also keys

9 up

6 left

8 down

7 right

0 fire/action

Joystick directions will move you in the direction you want, the fire button will perform actions, either swinging your sword, reading signs, dialog and flipping switches.

## Examples of using your Action/Fire key



Pressing Action/Fire at a sign, performs an action, such as reading the sign. Press Fire again to return to game



Pressing Action/Fire on a NPC will trigger a conversation



Pressing Action/Fire while not on an actionable object will cause your sword to activate



Pressing fire on an object allow you to collect the object.



The object is now in your inventory

## Screen Layout



The screen is laid out rather simply. Your life force is on the top left, indicating how much life you have until the game is over. Your inventory of collected objects is at the bottom of the screen. Your hero which you control is all over the place.

## Playing the game

Xelda is a top down / overhead adventure game. The goal is to find and collect the “Golden Apple” which disappeared under mysterious conditions (got stolen in the dark of the night). When the “Golden Apple” disappeared, monsters and beasts filled the island, terrorizing the citizens. Return the “Golden Apple” to the statue and all will be well.

Guide your hero around the island, talking to people to help guide you to your goal. Avoid or kill the monsters that block you from achieving your goal. Collect various object to open gates, flip switches to open passages and search the island until you find the “Golden Apple”. Solve the puzzles around the island to achieve your goal.

## Your Character



### Our Hero

Our hero, small in stature, yet has a huge heart (he needs to cut down on the cholesterol) wants to save the island from the nasties wandering about. He can do so one by one, yet they always seems to regenerate after a time (thanks mr. programmer) or he can do so by returning the “Golden Apple” to its proper place.

Can't quite remember his name (Dink, or something like that), so he goes by the name Bubba. Bubba is quite unassuming, and needs your guidance to move about. He just doesn't know how to defend himself properly, so guide him on when to use his sword.

He has 25 life points and a sword (and kinda knows how to use it). Help guide him through the island, swashbuckling, exploring and solving puzzles.

## Other Characters



### Island Keeper

*The island keeper is an elected position, his real name is Bob. Bob always wins the position, because no one else wants the job. Like most politicians, he's more talk than action, and boy can he ramble on, stating the obvious and treating it like it's brand new information.*



### Blacksmith

*Here is a man with a hammer and he knows how to bring it down. He's quite handy when it comes to fixing metal. His mustache is related to the woodsman.*



### Woodsman

*He is a lumberjack and that's OK, he sleeps all night and works all day. Well known for his flavor saving mustache.*



### Dungeon Gatekeeper

*She knows how to get into the dungeon, she rather likes dungeons and has gotten herself grounded because of it.*



### Swamp Gatekeeper

*Here's an odd fellow, perhaps from living in his cave. It smells kinda funny in there. His sceptre has gone missing.*



### Doug Bag

*He fancies himself a magician and has got the hat to prove it. Doug is always hungry. Doug has no claims no relation to Anthony.*



Anthony Hole

*Anthony is related to Doug, also a magician. His magic is found in his hair. He always keeps it tidy in a permanent wave.*



Desert Gate master

*She guards the second dungeon and knows how to open the gate, but is a bit forgetful. She forgot the magic word, but no fear, she wrote it down in a book. Now, where did she place that book?*



Librarian

*The librarian has surrounded himself by books of all types. Old and musty books and slightly less older and musty books, etc.. He's kinda grouchy, perhaps because no one likes to file away the books he keeps. No one seems to respect the Dewey Decimal system anymore.*



Island Wizard

*The Wizard and his mighty staff can open gateways, making certain parts of the island more accessible. Don't be too offended if he ask you to find his staff for him, you should be honored to handle his staff.*



Boom Betty

*Boom Betty like to blow up things. It wasn't always that way, but once she discovered how to make thing explode, there was no stopping her. Of course, her name is just Betty, she used to be called Boom Boom Betty, but shortened it to Boom Betty.*

## Monsters



Quadrock, these unassuming mindless monsters hurt when they touch you.



Pig faced devils. Equally ugly and mean, they are as mean as they look.



Crab. Grouchy, yet tasty beasts. Those powerful pinchers can really hurt.



Snakes are found all over the island, they can slither just about anywhere. Why did it have to be snakes.



Giant Spider. Spiders like forests, great places to wander about and catch prey. They can put a powerful bite on you, unless you get them first.



Scorpions. They tend to hang around the desert, watch out for their stingers.



Knight. They will hurt you just by touching you. What's he doing with that finger. Hey!



BoneHead, about as smart as that empty skull on top of his shoulders, but boy can they hurt.



Dark Knight. This is not a Bat Man, this is a dark knight, mean and nasty.



Dog Faced meanie. Mean because he's called Dog Faced.



Bat. Flies over trees and obstacles.



Ghost. Sucks the synergy right out of you.



Robed Skull, a more modest version of Bone head.



Slimy Ghost. Who you gonna call?



Jumper. They have a powerful bite and can hop right on top of you.



Blob, he's big, he's bad, he's Blob.



Screamer. Runs around in the dark and screams, fortunately he screams at a pitch above your hearing level, but can hurt you any way just by touching you.



Walking Cuke, popular with all the girls.



Walking Thing Fiend (or WTF for short). Shoot balls of energy at you that can hurt you.



Ghost. Can scare the life out of you, especially with the way he's rendered.



Shielded knight. His shield is studded for his pleasure, but not yours.



Skating mouth. Why, who knows, they tend to wander around in the dark, skating through life and biting things.

## Obstacles



Wall. Can't get past a wall, unless somehow the wall is weak, you might be able to destroy it.



Fence, somehow our hero cannot figure out how to climb a fence, better find a way around.



Our hero cannot climb a rock either, best to find a way around.



Deep water, our hero cannot swim well either.



Shallow water, other than getting your feet wet, what harm does it cause walking through a puddle.



Tree, can't go through, can't go over, can't go above, must go around.



Blocks. Our hero should take a step stool with himself so he could get over these things.



Push Blocks, they can be push around to solve puzzles.



Reset Button. Makes puzzles reset back, might be a good thing, might be a bad thing.



Gate, must find a way to open it.



Gravestone, of all the hero's that came before you. Surely you won't be the next.



Mushroom. You might like mushrooms, but these are funny, so you know better than eat some wild mushroom you find, no reason to take another trip.



Spike Pits. What sort of moron places spike pits all around the island. Monsters seems unaffected by them, so they only hurt you.



Spike-heads. These unmoving guys just sit around and don't move. Don't touch them. They can be temporarily killed, but move off screen and come back and there they are again.



Spikes. Probably the same idiot or a lazy cousin that placed the spike pits, placed these irritating spikes around the island.



Small Cactus. You can destroy these with your sword. They do grow back fast though. They won't grow if you look at it. Try turning away and you will be surprised on how fast it grows. Don't touch it though.



Cactus, the biggest spikiest thing you will run into. So don't.



A weird oddball boulder.

## Objects and Doorways



Sign, signs say stuff, you like to read, match made.



Houses lead to people, people are interesting to talk to, so why don't you walk in uninvited, I'm sure there will be a lively conversation.



Teddy Bear. Helping kids sleep for years.



Bush. Who doesn't like a little bush.



Gateway, they lead to places, you like to go to places.



Life, you get a full refill of life force when you grab this. You can only grab a life once, but there are a number of them around, use carefully.



Fire Flower. These little flowers can burn and light up your fire.



Torch. Helps you to explore dungeons.



Axe. When wielded by a properly trained woodsman, it can bring down trees.



Switch, some move up and down, some just move up or down. Things happen when you yank on one.



Clam. Once opened you'll find a sweet surprise inside.



Crab Leg. I don't know how crabs loose their legs, but yummy.



Orbs, big shiny balls. Everyone likes their balls shiny.



Keys. They open gates.



Key Hole. Guess what the keys plug into



Dynamite. Boom Betty like to place these in cracks and blow up things.



Wizard Staff. Beware the wizard with two mighty staffs.



Golden Apple. Don't try to nibble on this apple, made out of metal.

## Dungeons

Dungeons are dark places, you will need some lighting to go inside. Even then, your light is not all powerful, you can only see a limited distance. The darkness tends to hide things, monsters, switches, passageways, be sure to explore thoroughly as a missed area might just hide a switch you are looking for to open a gate.

## Winning the game

The game is won, once you find the Golden Apple and return it to the statue. The sign has instructions on what to do, just read the silly sign, it's there on purpose.

## Epilogue

I really do hope that you enjoy this game, I've put a lot of time into creating it. I had 3 goals in mind when I created this game, first to create a good top down graphic adventure game that people might enjoy playing, secondly to provide a template for others to use to create their own full scale MK2 game that should be rather easy to understand with the source code that will be released. Finally, I intend to popularize the MK2 engine for game creation. Enjoy.

Andy Dansby