The Playdates File Format Reverse Engineering

**Liquid - 2023**

**Intro**

A little while ago in 2022, I wrote a Chip8 Emulator for the Playdate. After that project I dropped development on the system, And I put the system in my desk. Today I Found it again and my interest sparked up again, Remembering that the Playdate has a custom package format and C compiler, I remember the C Api Playdate library, and its ridiculous use of unnecessary pointers, and I thought I can develop a much better Library that not only ran faster but looked less like a travesty. So these will be my notes of looking into exactly what all of the PDX, PDI, and PDZ files do and how you can make them without the use of proprietary closed source tools.

**-PDX-**

So to put it nicely and sort, The Playdates PDX Files are essentially a renamed zip file, They are nothing more than, thankfully.. Each PDX contains a ->

**pdex.bin** - The Executable

**pdxinfo** - The PDX Meta Data

**pdex.so** - The System Library

<- File, Optionally you can also use sub folders to contain assets and such. These files are not very complex..

*Curiously the Operating systems PDX files contain no bin files. They have a format called a PDZ file. There also is not a .so Library.*

The pdex.bin file is compiled with weird linking by the Playdate compiler, It would look something like ->  
  
 .bss - First

.data

.text - Last

->

Note this won't do anything if you change the alignment.

The pdex.bin file and pdex.so also do not have a header, they are Raw binary files through and through.

**-PDZ-**

**The Header -**

**00000000 50 6c 61 79 64 61 74 65 20 50 44 5a 00 00 00 00 |Playdate PDZ....|**PDZ Files Always have a unique header to them as shown above. But they look to just be a Raw Binary file with an altered header.

**How To Replicate with GCC -  
  
  
This is the Code and GCC Commands I've used to Replicate a PDZ File**

**Main.c -**

void generate\_header(void) {

asm volatile (

".section .data\n"

"header\_data:\n"

" .byte 0x50, 0x6c, 0x61, 0x79, 0x64, 0x61, 0x74, 0x65, 0x20, 0x50, 0x44, 0x5a, 0x00, 0x00, 0x00, 0x00\n"

".section .text\n"

);

}

int main(){

generate\_header();

// Example Code After

unsigned char Test = 0;

while (1){

Test = Test + 1;

}

}

**Pdx.ld -**

SECTIONS

{

. = 0x00000000;

\_start = .;

.data : {

\*(.data)

}

.bss : {

\*(.bss)

}

.text : {

\*(.text)

}

\_end = .;

}  
  
*Notes - I had to shift around the Data structure because GCC Likes to put the .data section below the .text, We need to have the Playdate PDZ Header at the top*

**Commands -**

**1:** arm-none-eabi-gcc -T pdz.ld main.c -o main.elf -nostartfiles -nostdlib

**2:** arm-none-eabi-objcopy -O binary main.elf **main.pdz**

**-PDI-**

The PDI Files are a little easier to figure out, They also have a custom header.

**The Header -**

**00000000 50 6c 61 79 64 61 74 65 20 49 4d 47 00 00 00 80 |Playdate IMG....|**