

GameBoy ROM Hack for infinitive continues

Game:	Absolute Invincible Raijin-Oh Zettai Muteki Raijinou
Hack by	blackdrake
Version:	July, 23th 2009

Abstract

This hack allows you to continue infinitely without returning to round 1.

Hexeditor patch summary

Hex-Offset	English ROM										Japanese ROM											
0x014e	Old	C1 31										Old	F5 30									
0x014e	New	C9 DE										New	FD DD									
0x0b51	Old	AF CD A4 2F CD 16 05 3E 01 EA 04 D1																				
0x0b51	New	CD 94 4E 00 00 00 00 00 00 00 00 00 00																				
0x8e92	Old	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00																				
0x8e92	New	18 0F FA 04 D1 FE 00 C0 CD ED 4E 3E 01 EA 04 D1 C9																				
0x8eeb	Old	00 00 00 00 00 00 00 00 00 00 00 00																				
0x8eeb	New	18 09 3E 00 CD A4 2F CD 16 05 C9																				

Disassembling explanation

Original state

0b51 is called at two places and is the action after „Start“ was pressed – so the game begins here.

ROM offset	BUS offset	Machine code	Assembler code	Comment
0x0b51	0b51	AF	XOR A	Set A=0 (A = A XOR A = 0). If this is replaced with NOP (00h) then you'll hear a part of the winning melody when the story appears.
0x0b52	0b52	CD A4 2F	CALL 2FA4h	Stop the music
0x0b55	0b55	CD 16 05	CALL 0516h	Show the story screen
0x0b58	0b58	3E 01	LD A, 01h	Load value 01
0x0b5a	0b5a	EA 04 D1	LD (D104h), A	Set it as round counter

The BUS areas 4e92..4ea2 (ROM: 0x8e92..0x8ea2) and 4eeb..4ef5 (ROM: 0x8eeb..0x8ef5) are filled with NOPs (00h). So we can insert some code there because there are no actions performed. But that does not mean, that the program counter will never reach these areas!

Part 1

In the original code the sound is stopped (A=0), the story is shown and then the round counter at offset D104h is set to 1. We have to insert some code, but we have not enough space, so we make a CALL to a free space in the ROM.

ROM offset	BUS offset	Machine code	Assembler code	Comment
0x0b51	0b51	CD 94 4E	CALL 4E94h	Jump to our inserted part and then return
0x0b54	0b54	00	NOP	Do nothing
0x0b55	0b55	00	NOP	Do nothing
0x0b56	0b56	00	NOP	Do nothing
0x0b57	0b57	00	NOP	Do nothing
0x0b58	0b58	00	NOP	Do nothing
0x0b59	0b59	00	NOP	Do nothing
0x0b5a	0b5a	00	NOP	Do nothing
0x0b5b	0b5b	00	NOP	Do nothing
0x0b5c	0b5c	00	NOP	Do nothing

Part 2

Now overwrite some hopefully unused NOPs for the insertion of the additional decision of the round counter. The round counter at offset D104 has the default value 0. If it is 0, initialize it with 1 and show the story screen. If it is not null, we go back and let the user play the game another time without story screen and without reset of the round counter.

ROM offset	BUS offset	Machine code	Assembler code	Comment
0x8e92	4e92	18 0F	JR 4EA3h	To be secure jump over our inserted part
0x8e94	4e94	FA 04 D1	LD A, (D104h)	Load the round counter into the register.
0x8e97	4e97	FE 00	CP 00h	Compare the round counter to 0
0x8e99	4e99	C0	RET NZ	If a round was set, we return!
0x8e9a	4e9a	CD ED 4E	CALL 4eed	Show the story screen only once at the beginning
0x8e9d	4e9d	3E 01	LD A, 01h	Load value 01
0x8e9f	4e9f	EA 04 D1	LD (D104h), A	Set it as round counter
0x8ea2	4ea2	C9	RET	Return to 0b54

Part 3

Overwrite another hopefully unused NOPs for the story part only for the beginning.

ROM offset	BUS offset	Machine code	Assembler code	Comment
0x8eeb	4eeb	18 09	JR 4EF6h	To be secure jump over our inserted part. I believe this is really necessary since there were no RET (C9h) before this NOP section.
0x8eed	4eed	3E 00	LD A, 00h	Equals our „XOR A“ (A = A XOR A = 0) from the original code
0x8eef	4eef	CD A4 2F	CALL 2FA4h	Change music (Silence because A=0)
0x8ef2	4ef2	CD 16 05	CALL 0516h	Show story screen
0x8ef5	4ef5	C9	RET	Return to 4e9d

Part 4

Now change the CRC checksum from c131 to c9de (English ROM) or f530 to fddd (Japanese ROM) with the hex editor as seen in the table above. If your original ROM has another checksum, then you have to change it as shown in your emulator tool. If you had the same original checksum as I had, but another checksum after the changes, then you did probably something wrong.

Notes

- No absolute jumps into the overwritten NOP areas found; relative jumps were not analyzed, but unlikely.
- I used VisualBoyAdvance and a hex editor to disassemble this game. It was really hard to work at it. Also, this page helped a lot with the CPU commands:
<http://nocash.emubase.de/pandocs.htm#cpuinstructionset> .
- Please also refer to <http://daniel-marschall.blogspot.com/2009/07/gameshark-cheats-for-absolute.html> where I posted some great cheat codes I found out by looking at the memory map.