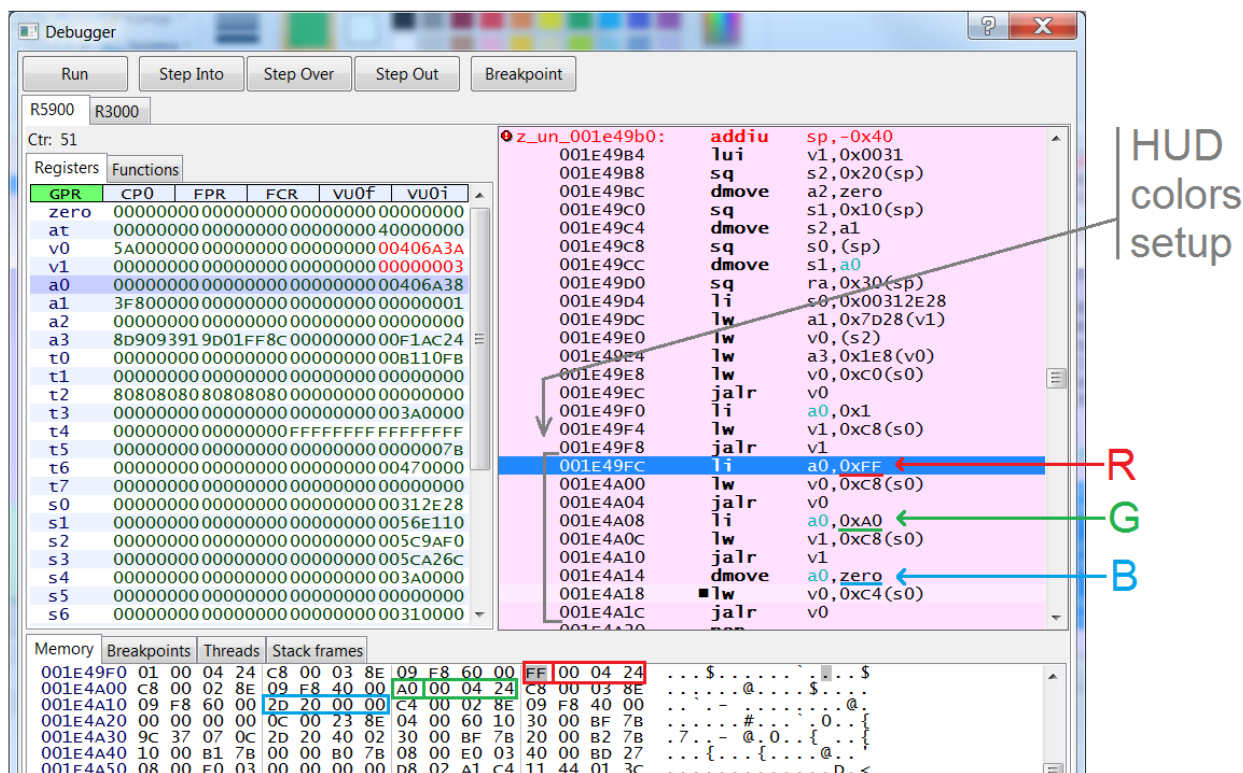


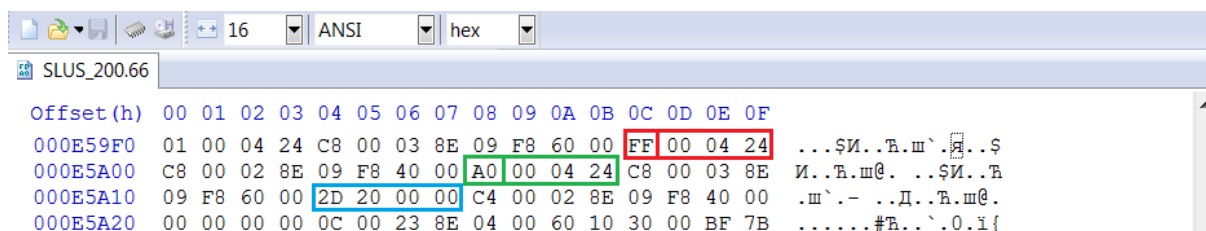
1 Changing color of the HUD

HUD color is baked into the executable file so the only way to change it is to do hacks. I was able to track down the point where the HUD color is set in PCSX2 debugger:

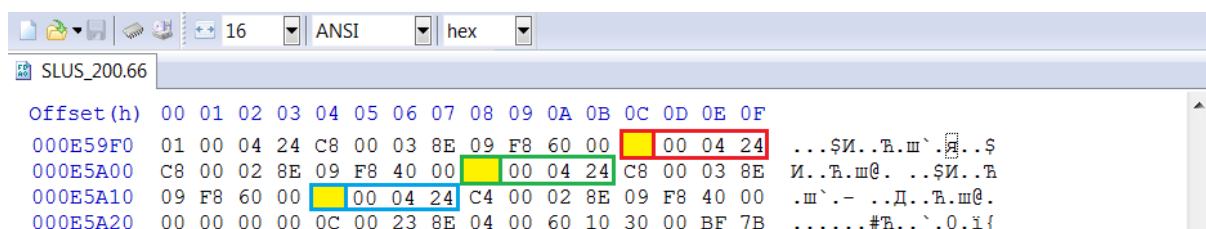


As you can see, changing red and green components is as simple as overriding operands. However, because blue component is equal to zero separate instruction for loading zero is used for it. So, to change blue component we need to change instruction to the one that is used for R and G.

Here is the same fragment, but inside an executable file (Offsets: 0xE59FC, 0xE5A08, 0xE5A14):



Here is how you change HUD colors:



First, you need to change last 3 bytes inside blue box from 0x20, 0x00, 0x00 to 0x00, 0x04, 0x24. Then all you need to do is just to input R, G and B components of desired color in yellow fields. That is it.

2 Enabling developer console

YouTube user GeckonCZ discovered that developer console is still present in PS2 Half-Life and can be accessed via a single byte hack. Link to his video:

<https://www.youtube.com/watch?v=TZpNdbHtSw8&t=1s>

These are known offsets that were published by him:

“HL PS2 USA ver 0.10: offset 0x00165428, change 01 to 00

HL PS2 USA ver 2.40: offset 0x00166988, change 01 to 00”

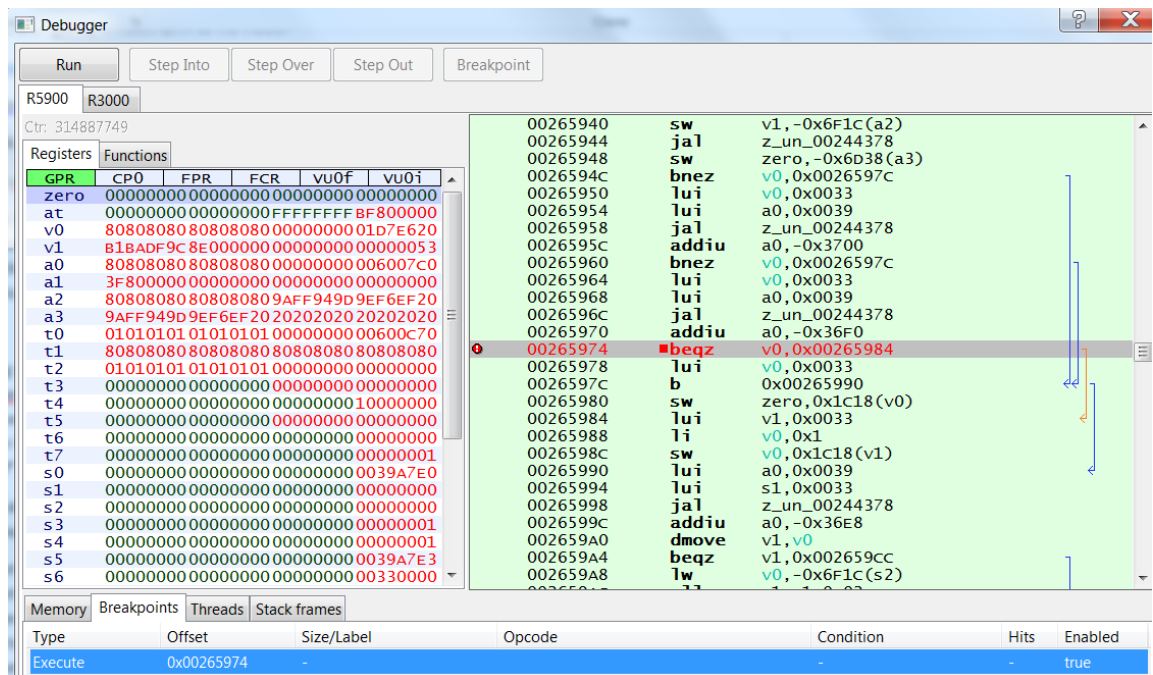
At the time I found this video he kept offsets in private. So I grinded my way to the console in PCSX2 debugger and I can add information on how this hack works in low level. Code that serves "toggleconsole" command checks for some flag and opens console only if this flag is set to 0. By default, this flag is set to 1 thus restricting console to show up. Here is a code that sets the flag:

00265974	beqz	v0,0x00265984
00265978	lui	v0,0x0033
0026597c	b	0x00265990
00265980	sw	zero,0x1c18(v0)
00265984	lui	v1,0x0033
00265988	li	v0,0x1
0026598c	sw	v0,0x1c18(v1)
00265990	lui	a0,0x0039
00265994	lui	s1,0x0033
00265998	jal	0x00244378

By changing this byte you effectively changing operand of selected instruction to 0 and thus setting the flag to 0. GeckonCZ later explained how it works on high level: “It just checks for the "-console" launch arguments as all the WON versions do. If that switch is found it sets the console CVAR to 1. And that's it... I'm not sure if you can easily pass launch arguments to PS2 executable so for me it was easier to patch the binary itself”.

What I can also add is the way to enable console in PCSX2 without changing files in hex editor using just built in debugger:

- 1) Start PCSX2.
- 2) Start PS2 Half-Life.
- 3) Open PCSX2 debugger (Debug->Open debug window ...)
- 4) Go to Breakpoints tab and add new execute breakpoint at 0x265974.
- 5) Reset the game.
- 6) Debugger should pop up and show this:

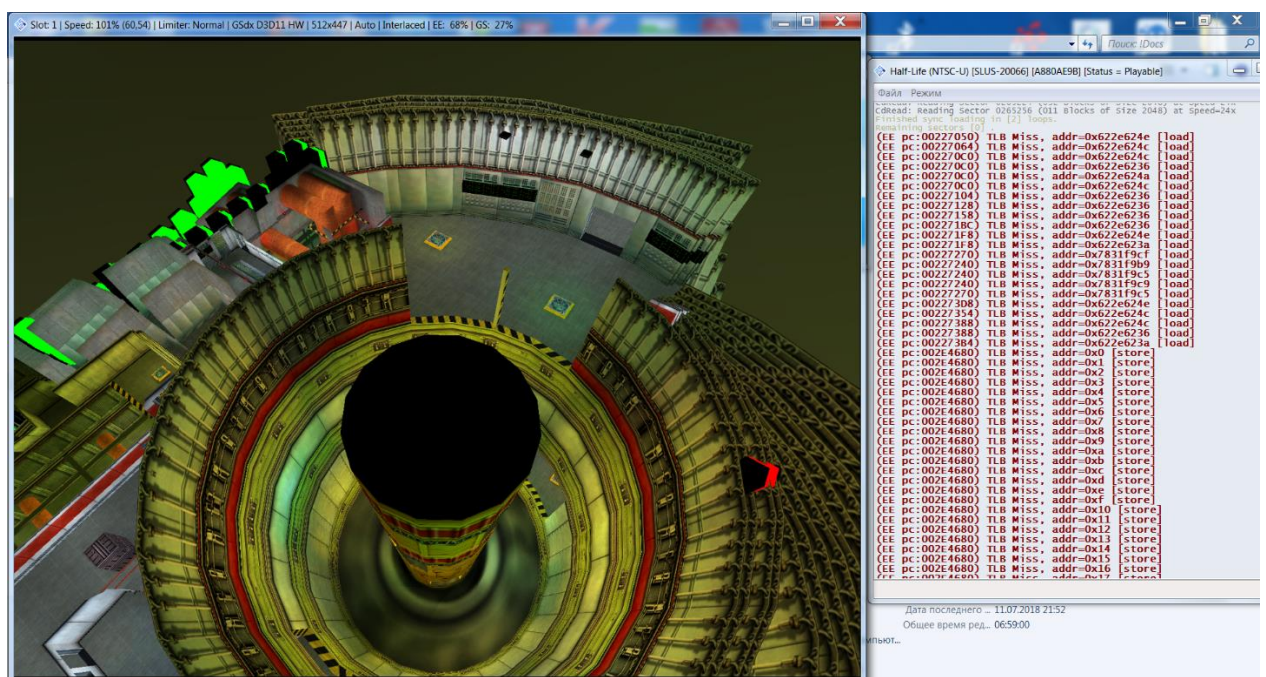


- 7) Press right mouse button on the next instruction after red line (lui v0,0x0033) and select "Jump to cursor" option.
- 8) Hit Run button. That is it, console should be enabled now.

3 Improving stability by increasing file reference buffer size

There is section in the RAM where a reference to every used map, model and sprite file is stored. Handling of this section is broken: references to files that are no longer used are remaining forever and not cleaned, thus causing this section to grow indefinitely towards SYSTEM.PAK section with each new file used. When it reaches SYSTEM.PAK section, it can either cause spawning out of bounds or game crash. And the only way to clean out this thing is to hit reset button on PS2.

I had successfully recreated this bug on vanilla PS2 HL game: I played in one sitting through hazard course and campaign, and then in the "Lambda Reactor" chapter I got out of bounds spawn and crash after I went through Portal 7:



Here is what RAM looked like right before the crash:

```

Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
21D16920 00 00 00 00 00 00 00 00 02 00 00 00 00 6D 61 70 73 2F 63 33 61 32 65 2E 62 73 32 00 00 00 00 00 00
21D16940 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16960 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16980 00 00 40 43 00 00 10 C3 00 00 00 00 00 00 3F 45 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D169A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D169C0 F2 10 00 00 50 EB F9 00 15 02 00 00 00 FD FC 00 32 14 00 00 60 A7 FF 00 2E 22 00 00 00 00 10 FE 00
21D169E0 FB 06 00 00 40 3E FB 00 2C 0E 00 00 00 FA 01 01 A5 43 00 00 C0 98 FE 00 A6 14 00 00 00 22 01 01
21D16A00 DC 11 00 00 B0 D3 09 01 80 EA 00 01 50 EB F9 00 00 00 00 FA 06 00 00 00 00 00 00 00 00 00 00 00
21D16A20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 60 22 01 01 50 EB F9 00 00 00 00 00 A5 14 00 00
21D16A40 00 00 80 C1 00 00 80 C1 00 00 10 C2 00 00 80 41 00 00 80 41 00 00 10 42 60 22 01 01 50 EB F9 00
21D16A60 8A 04 00 00 A5 14 00 00 00 00 C2 00 00 00 C2 00 00 00 C2 00 00 00 42 00 00 00 42 00 00 00 42
21D16A80 60 22 01 01 50 EB F9 00 BA 08 00 00 A5 14 00 00 00 80 C1 00 00 80 C1 00 00 90 C1 00 00 80 41
21D16AA0 00 00 80 41 00 00 90 41 72 00 00 00 20 2F 11 01 20 1B 0A 01 70 61 0A 01 50 7F 23 01 00 00 00 00
21D16AC0 00 00 00 00 02 00 00 6D 61 70 73 2F 63 33 61 32 2E 62 73 32 00 00 00 00 00 00 00 00 00 00 00
21D16AE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16B00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16B20 00 00 64 C4 00 00 00 00 00 00 2C 45 00 00 27 45 00 00 14 44 00 00 00 19 6B 76 45 00 00 00 00
21D16B40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 D3 04 00 00 34 00 00 60 47 FE 00 E8 0B 00 00 00 00
21D16B60 50 EB F9 00 AF 00 00 00 B0 C1 FB 00 87 0A 00 00 00 2E FD 00 F7 11 00 00 30 57 FC 00 A1 03 00 00
21D16B80 70 D9 FA 00 6D 07 00 00 A0 57 FE 00 BC 23 00 00 10 9F FC 00 1C 0A 00 00 80 F6 FD 00 84 08 00 00
21D16BA0 F0 84 02 01 70 D9 FD 00 50 EB F9 00 00 00 00 A0 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16BC0 00 00 00 00 00 00 00 00 00 00 80 F6 FD 00 50 EB F9 00 00 00 00 00 1B 0A 00 00 00 80 C1
21D16BE0 00 00 80 C1 00 00 10 C2 00 00 80 41 00 00 80 41 00 00 10 42 80 F6 FD 00 50 EB F9 00 8B 01 00 00
21D16C00 1B 0A 00 00 00 00 C2 00 00 C2 00 00 C2 00 00 00 42 00 00 42 00 00 42 00 00 42 00 F6 FD 00
21D16C20 50 EB F9 00 08 03 00 00 1B 0A 00 00 00 80 C1 00 00 80 C1 00 00 90 C1 00 00 80 41 00 00 80 41
21D16C40 00 00 90 41 5F 00 00 00 20 75 06 01 00 A7 02 01 D0 B1 02 01 E0 A6 18 01 00 00 00 00 00 00 00
21D16C60 02 00 00 00 6D 61 70 73 2F 63 33 61 32 61 2E 62 73 32 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16C80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16CA0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16CC0 00 00 00 00 00 00 D8 43 00 00 2E 45 00 00 A2 44 00 00 00 D7 F0 84 45 00 00 00 00 00 00 00 00
21D16CE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16D00 12 06 00 00 50 A6 00 01 E8 29 00 00 10 BC 06 01 D4 46 00 00 90 6D 03 01 27 10 00 00 90 9C FC 00
21D16D20 9A 1D 00 00 80 76 0B 01 C9 8C 00 00 00 E0 88 04 01 F7 2C 00 00 D0 DB 09 01 D2 24 00 00 20 1D 1C 01
21D16D40 90 5A 09 01 50 EB F9 00 00 00 00 00 00 26 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16D60 00 00 00 00 00 00 00 00 D0 DB 09 01 50 EB F9 00 00 00 00 F6 2C 00 00 00 80 C1 00 00 80 C1
21D16D80 00 00 10 C2 00 00 80 41 00 00 80 41 00 00 10 42 D0 DB 09 01 50 EB F9 00 F2 09 00 00 F6 2C 00
21D16DA0 00 00 00 C2 00 00 00 00 C2 00 00 00 00 00 00 42 00 00 42 00 00 42 00 DB 09 01 50 EB F9 00
21D16DC0 20 13 00 00 F6 2C 00 00 00 80 C1 00 00 80 C1 00 00 00 C1 00 00 80 41 00 00 80 41 00 00 90 41
21D16DE0 79 00 00 00 B0 DB 2C 01 70 80 1C 01 00 78 1D 01 B0 4A 3F 01 00 00 00 00 00 00 00 02 00 00 00
21D16E00 D2 62 2C FE 10 2E 02 00 30 9C D3 01 E0 35 CE 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16E20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16E40 50 41 43 1E D0 2C 02 00 00 01 00 00 00 00 00 19 00 00 00 00 00 00 00 00 00 00 00 00 00 00
21D16E60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

maps/c3a2s.bs2.....
.....D
..@C...F.....FE...E.....TE
.....>.....hS
...Pmm.....as.2.....Sa.....B
...@Ma.....pB...FC...A...
...Y...E...Epm.....S
.....Pmm.....I.....
..SE...SE...B...BA...B...Pmm
...I.....B...B...B...B...B
...Pmm.e...I.....SE...hB...BA
...hA...hA.../.....BS.a.#.....
.....maps/c3a2s.bs2.....
.....Bh...@C
...dL.....E...E...D...kvE...
.....Y...4.....Gm...
Pmm.I....."Bm...Y...4...Om...Y...
pBb.m...Wm.j#...ub...Tm...
p...pBb.Pmm.....
...Bm.Pmm.....
..B...B...BA...BA...Bm...
...B...B...B...B...B...Bm...
Pmm.....B...B...B...hB...BA...BA
..hA...u...S...P...a...
.....maps/c3a2s.bs2.....
.....E...E...
.....SE...E...
.....W...J...J...C...V...Pmm
...P...n)...j...F...Pm...
a...Pv...Pm...e...4...Pm...TS...
hZ...Pmm.....
...Pm.Pmm.....u...B...B
...B...BA...Bm...Pmm...u...
...B...B...B...B...B...Pm...
...u...SE...SE...hB...BA...BA
Y...H...p...Ax...J?..
Tb...OmY.a50...
BACHP

File references

Adding one more reference would lead to crash

SYSTEM.PAK

This is quite good endurance for original game, but in my Blue Shift mod this bug caused crash on the beginning of “Focal point” chapter of one sitting playthrough.

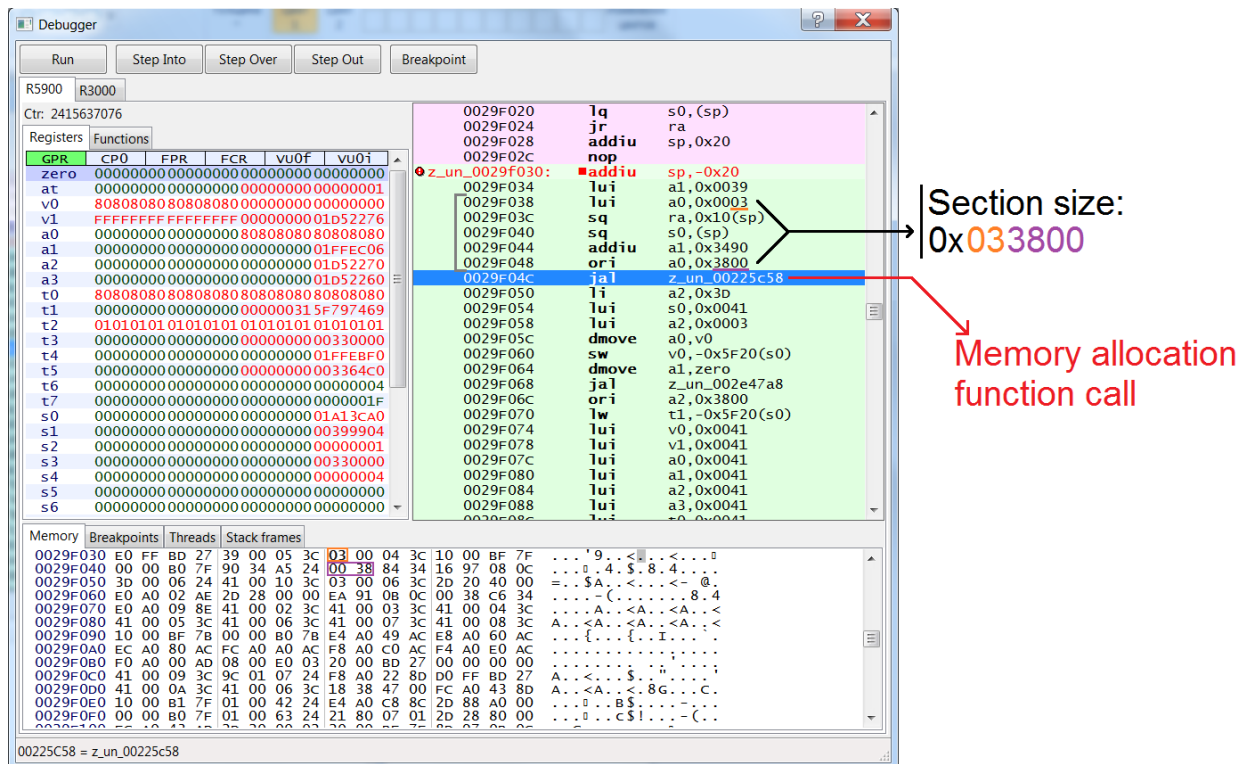
The good thing is that this section is allocated dynamically, which I determined by presence of this header that is placed right before it:

???	Current section size	Ptr. to next section	Ptr. to prev. section
82 8E 2E FE	00 38 03 00	80 70 D1 01	00 E2 AE 00

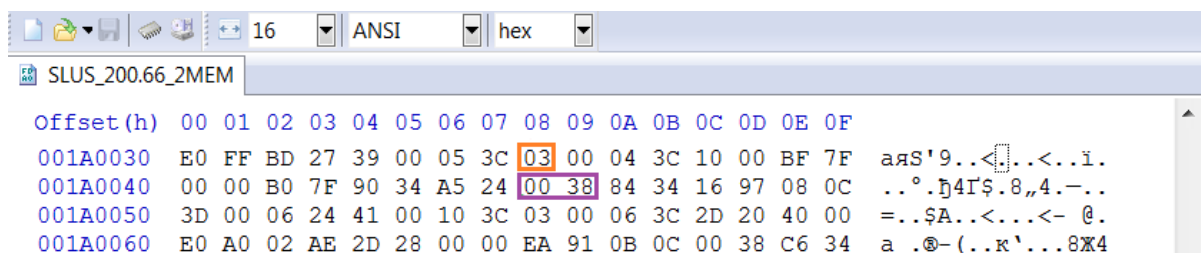
So a size of this section is 0x33800 bytes. Each file reference takes 0x19C bytes. So:

$0x33800 / 0x19C = 0x200 = 512$ – is maximum amount of files that can be used during one playthrough.

Because I had found exact location of section size field I used PCSX2 debugger to find place where this value came from:



Then I found location of these operands in executable file (offsets: 0x1A0038, 0x1A0048):



Then I rewrote them to allocate 0x80000 bytes (512 KiB) and It helped to stabilize Blue Shift so it was possible to finish it in one sitting:

