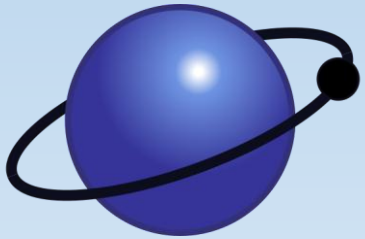


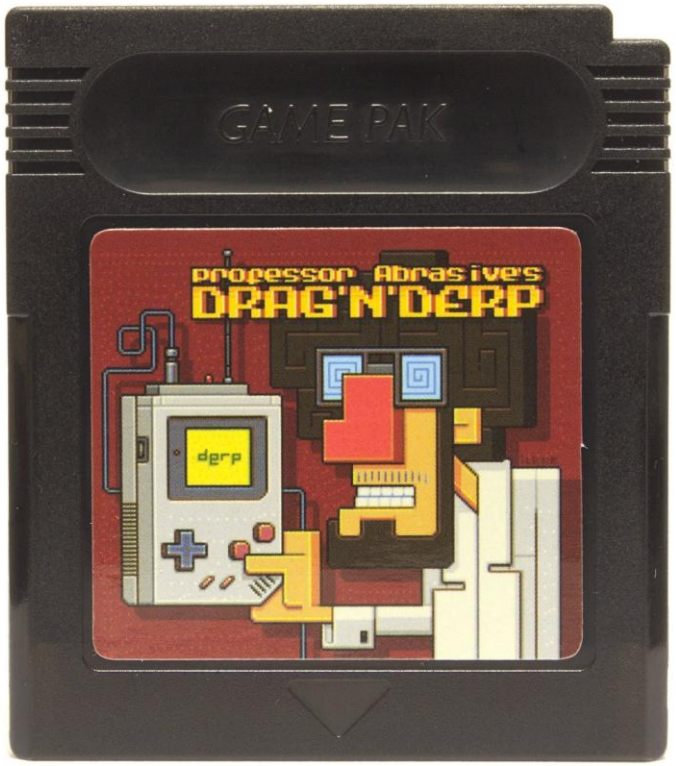
Orbiting the Saturn

Investigating a decades-old DRM system

James Laird-Wah

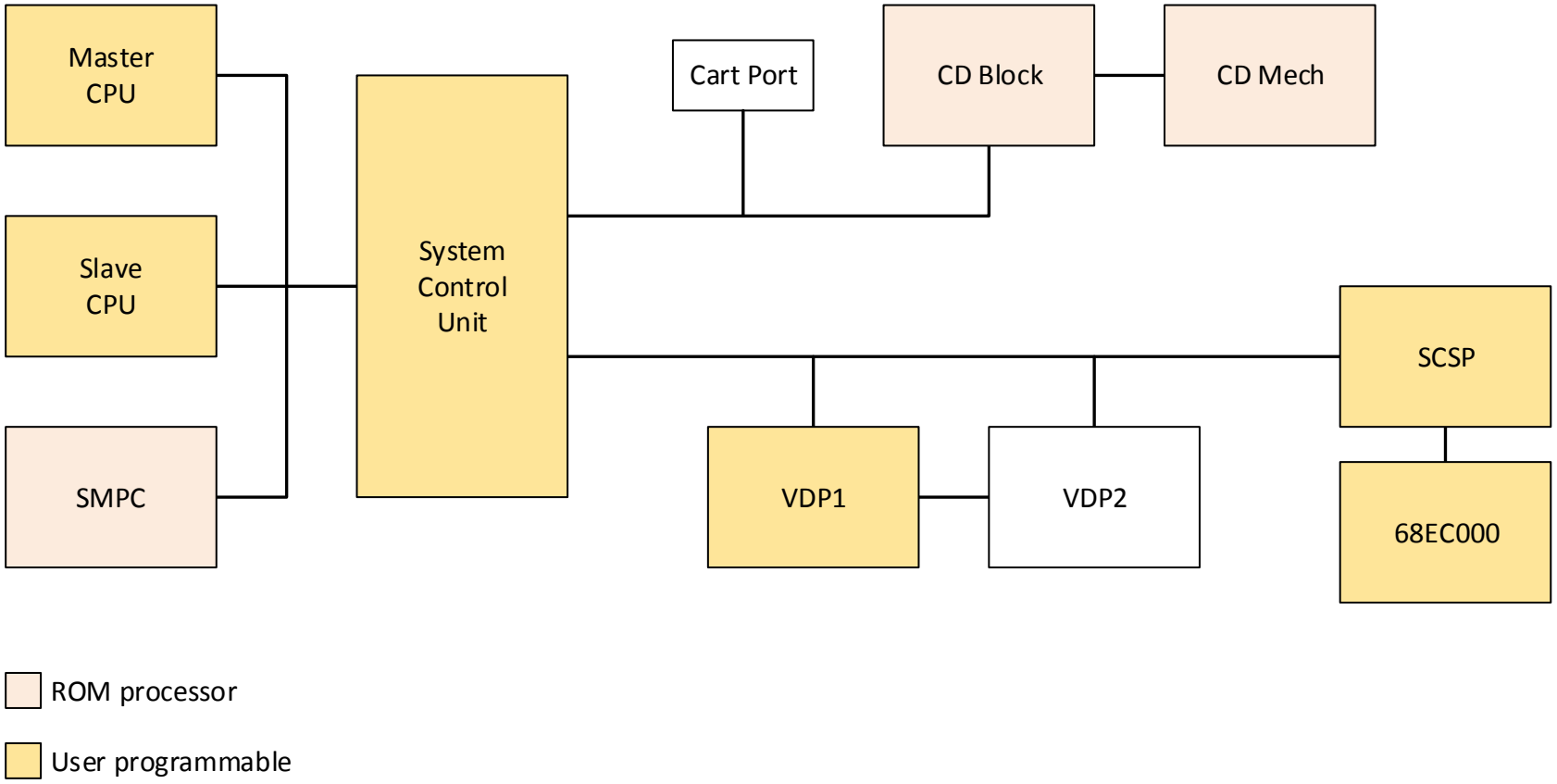
Chipmusic?

Drag'n'Derp: a USB cartridge

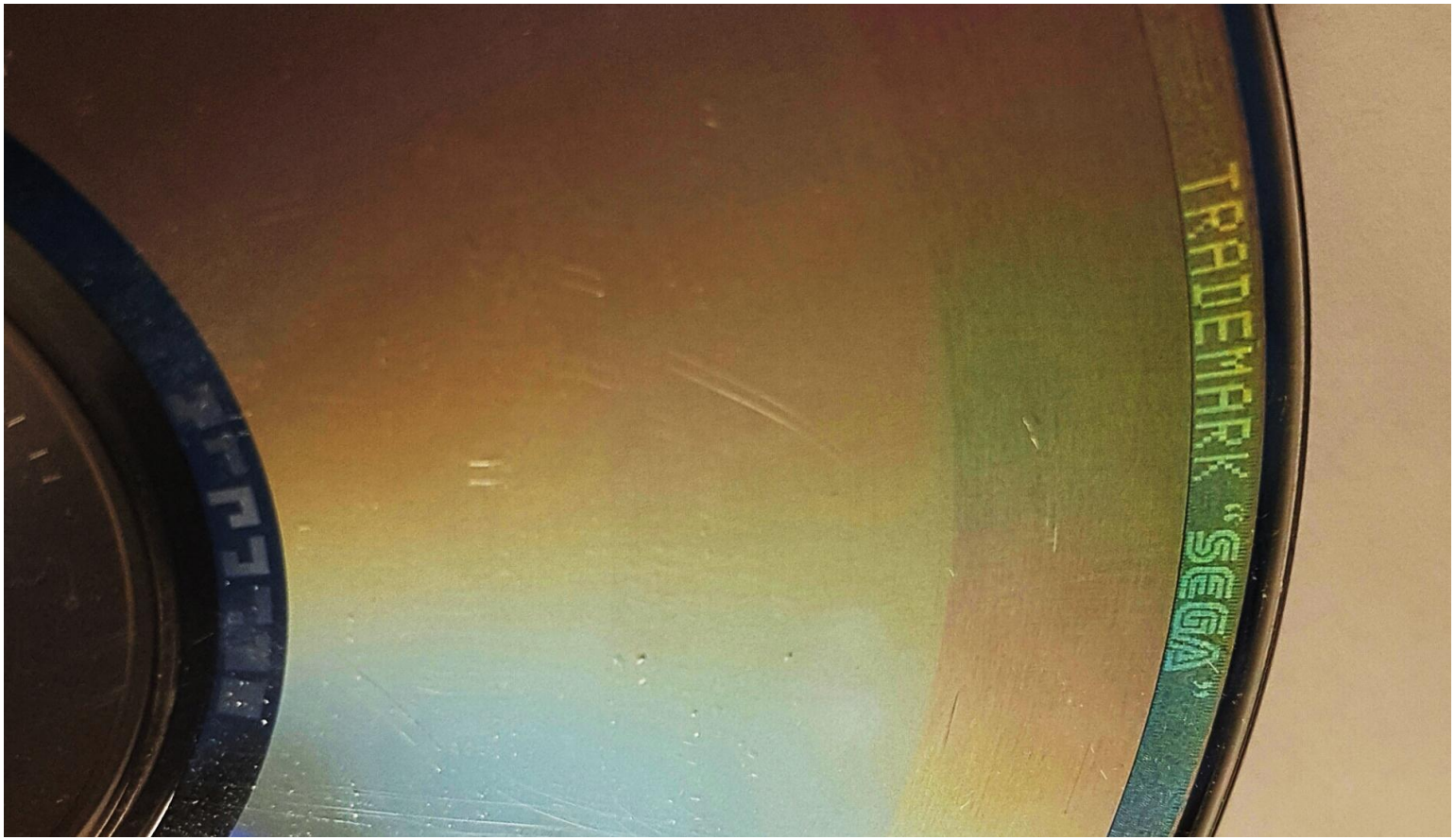


Why the Saturn?

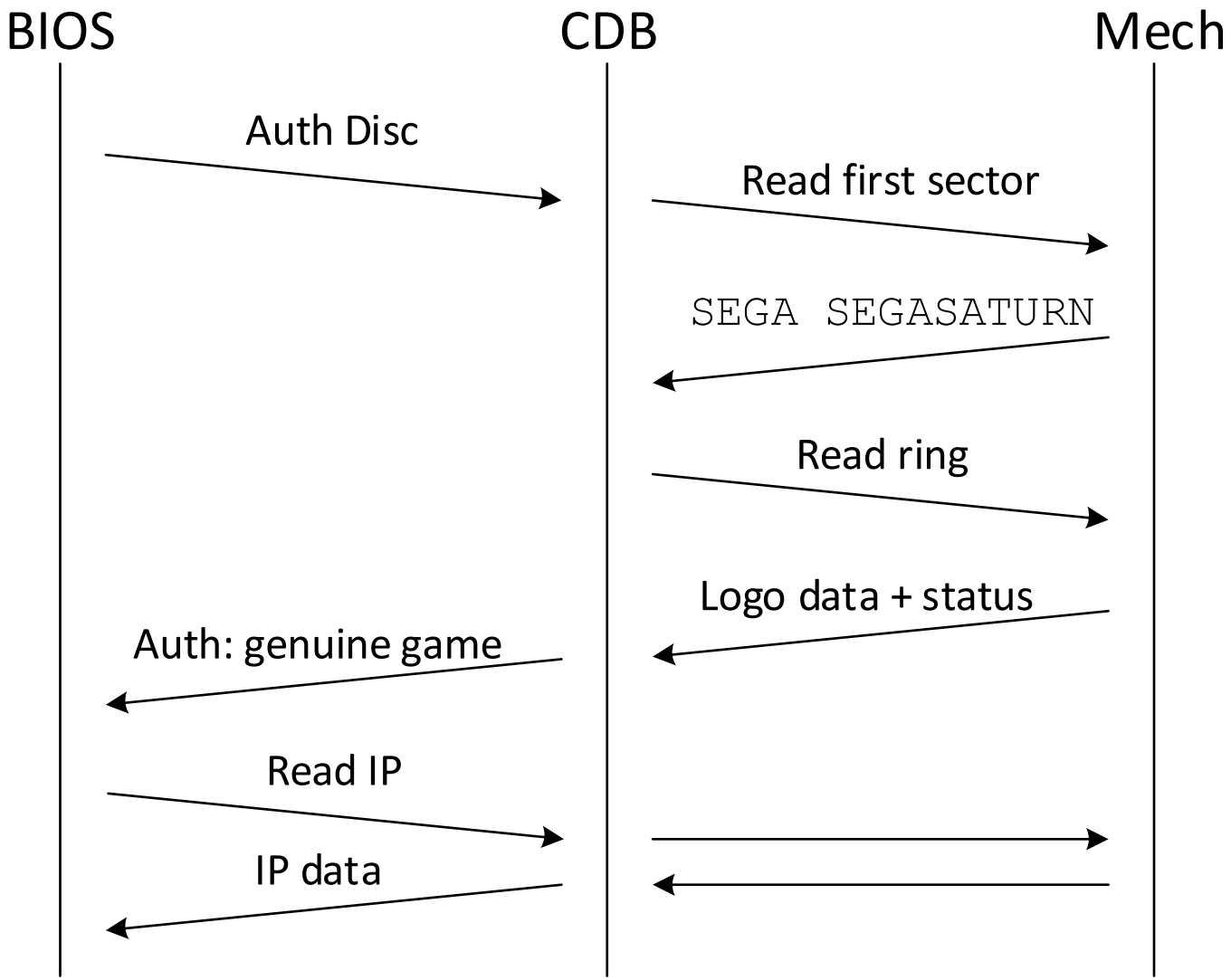
Architecture



Saturn discs



Boot sequence

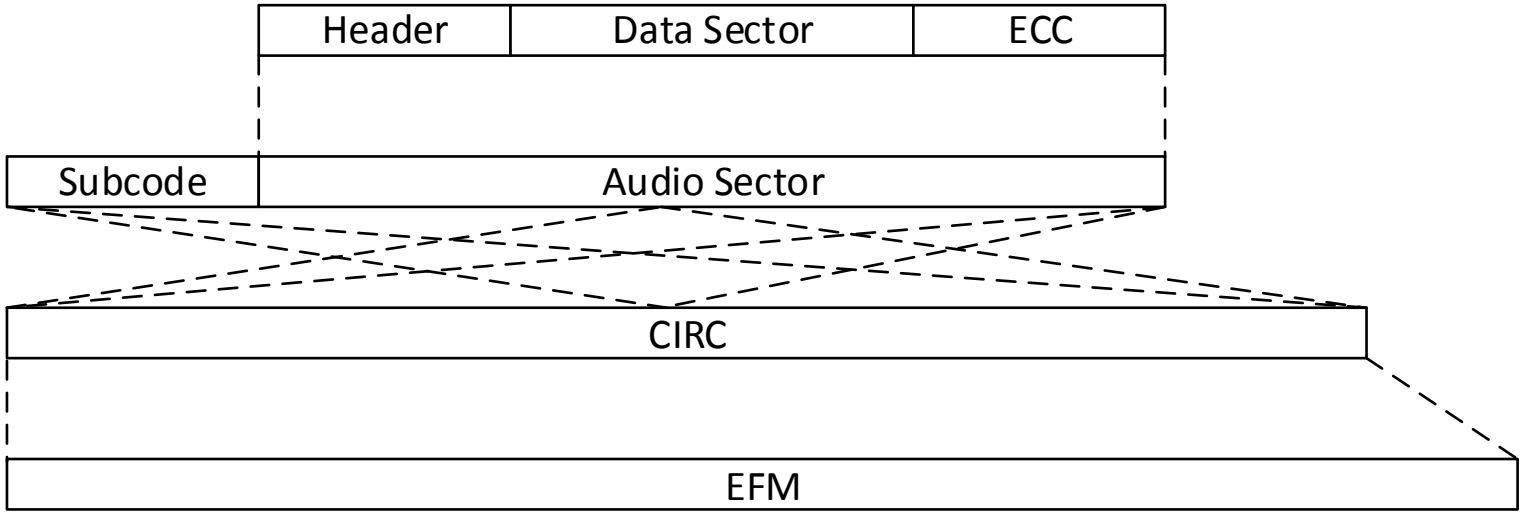


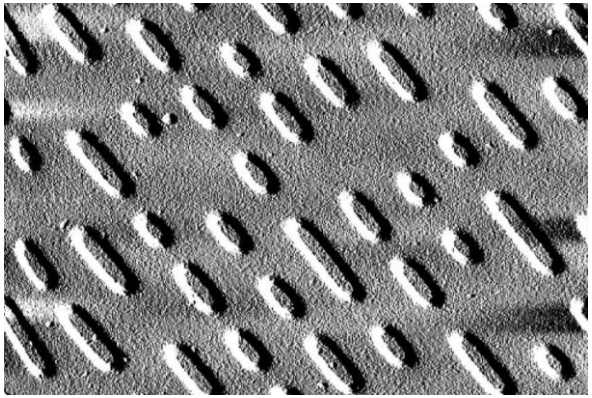
System Disc


SEGA SATURN
SYSTEM-DISC(KD01)
COLUMBIA
COMPACT
disc
610-5732-05
Part No.670-6481-05
MAKER ID=
SOAD19113I
SEGA

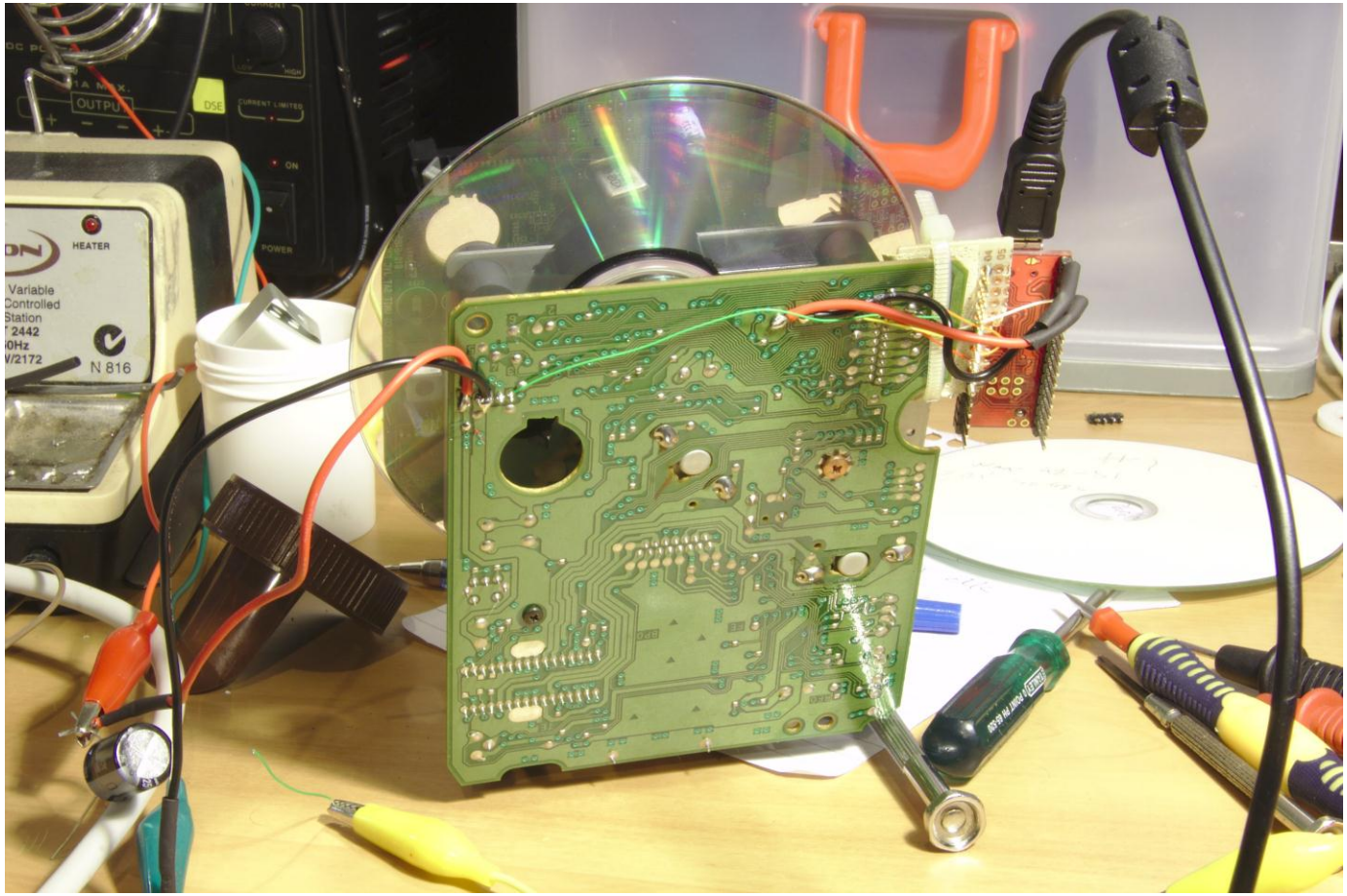
“Just cut a real disc…”

CD sector format





Isolating the mech



Mech status bytes

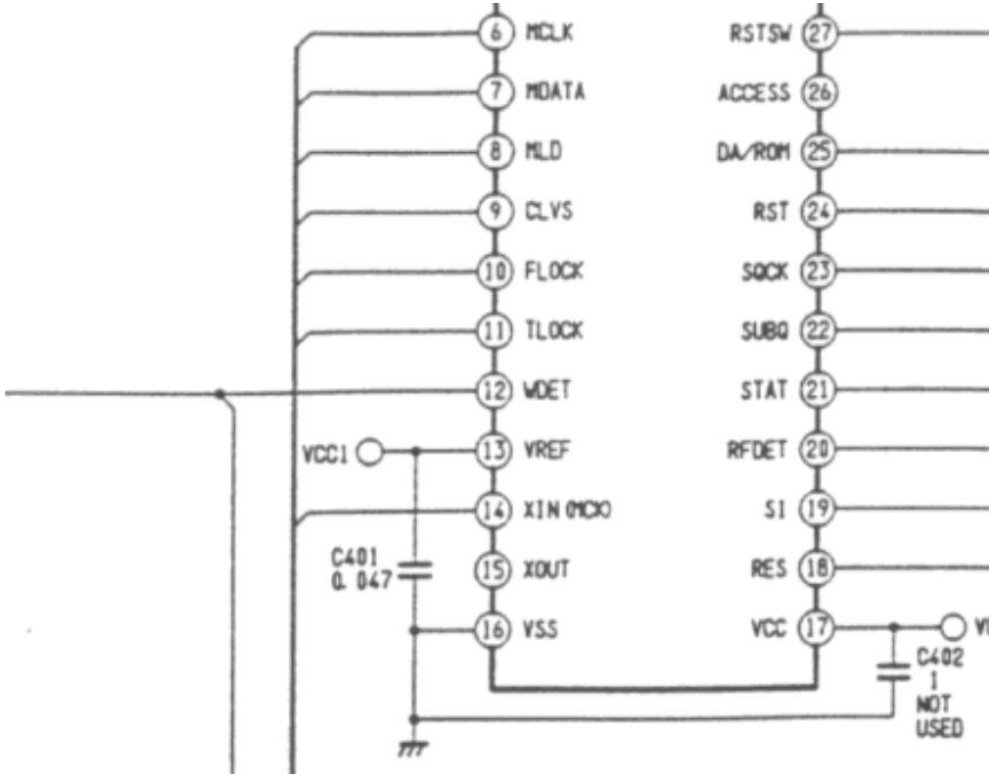
Regular data area

```
b2 00 00 03 ad 3f 00 00 00 00 00 5e 01
b2 00 00 03 ad 3f 01 00 00 00 00 5d 01
b2 00 00 03 ad 3f 01 00 01 01 00 5b 01
b2 00 00 03 ad 3f 02 00 01 01 00 5a 01
b2 00 00 03 ad 3f 02 00 02 02 00 58 01
b2 00 00 03 ad 3f 03 00 02 02 00 57 01
b2 00 00 03 ad 3f 03 00 03 03 00 55 01
b2 00 00 03 ad 3f 04 00 03 03 00 54 01
```

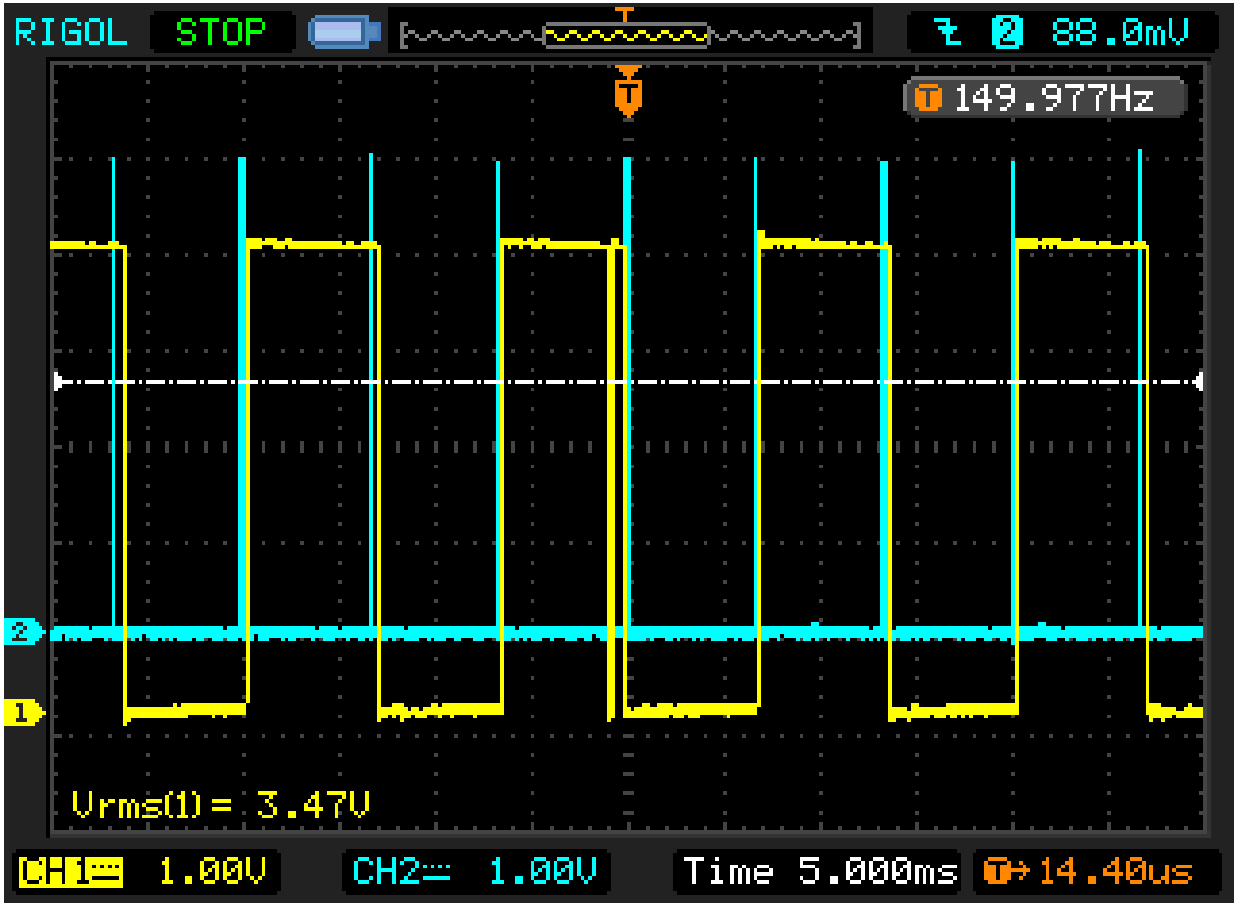
Ring area

```
b2 00 00 04 ad 3f 00 00 00 00 00 5d 01
b2 00 00 04 ad 3f 01 01 00 00 00 5b 01
b2 00 00 04 ad 3f 01 01 01 01 00 59 01
b2 00 00 04 ad 3f 02 02 01 01 00 57 01
b2 00 00 04 ad 3f 02 02 02 02 00 55 01
b2 00 00 04 ad 3f 03 03 02 02 00 53 01
b2 00 00 04 ad 3f 03 03 03 03 00 51 01
b2 00 00 04 ad 3f 04 04 03 03 00 4f 01
```

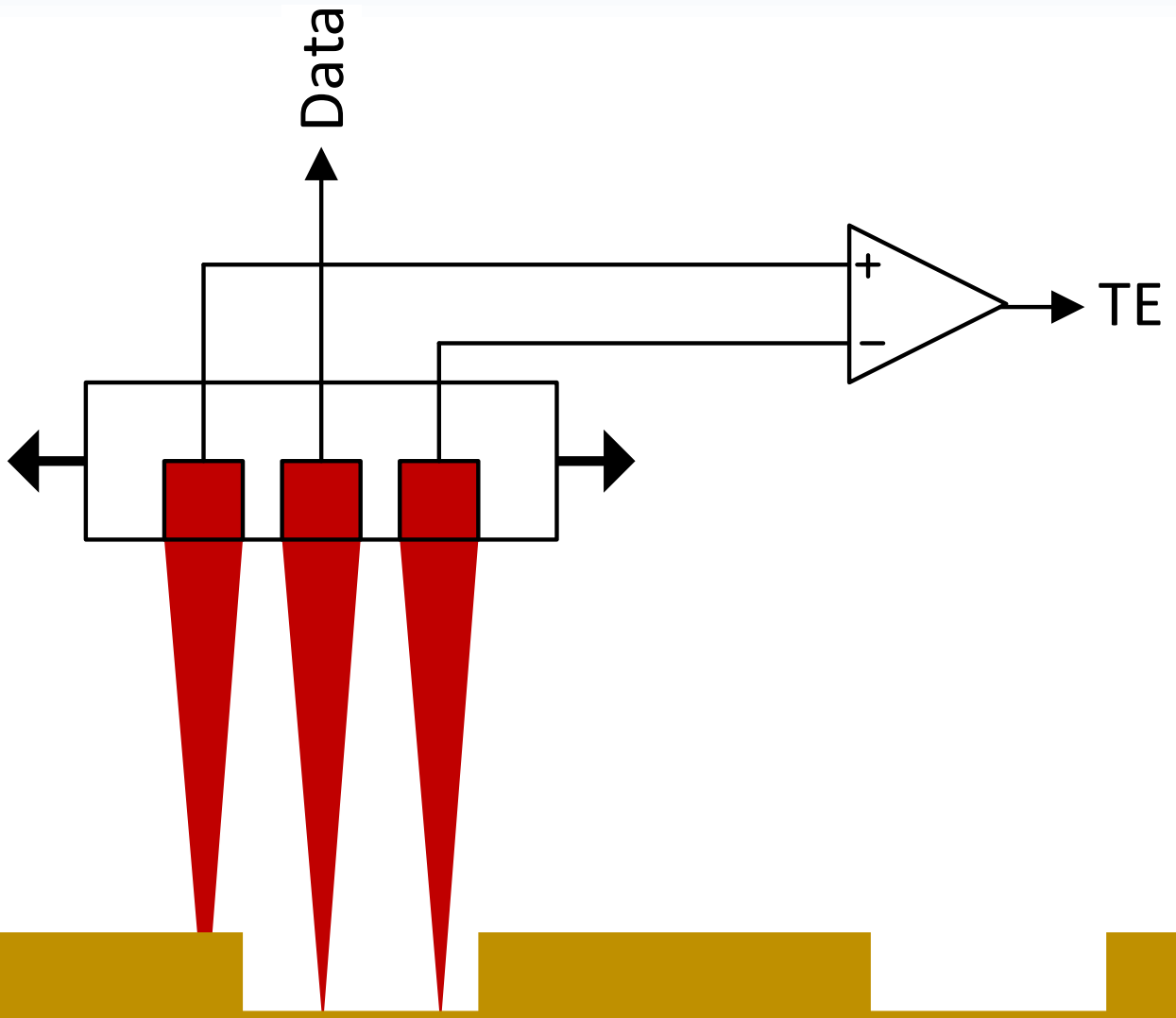
Mech's MCU


6 MCLK
7 MDATA
8 MLD
9 CLVS
10 FLOCK
11 TLOCK
12 WDET
13 VREF
14 XIN (MCK)
15 XOUT
16 VSS
VCC1
C401
0.047
RSTSW 27
ACCESS 26
DA/ROM 25
RST 24
SQCK 23
SUBQ 22
STAT 21
RFDET 20
SI 19
RES 18
VCC 17
C402
1
NOT
USED

WDET?

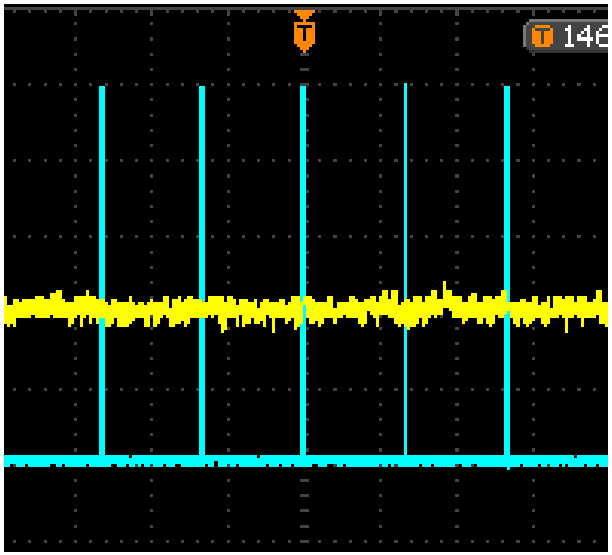

RIGOL
STOP
88.0mV
149.977Hz
Vrms(1)= 3.47V
CH1 1.00V
CH2 1.00V
Time 5.000ms
14.40us

Tracking

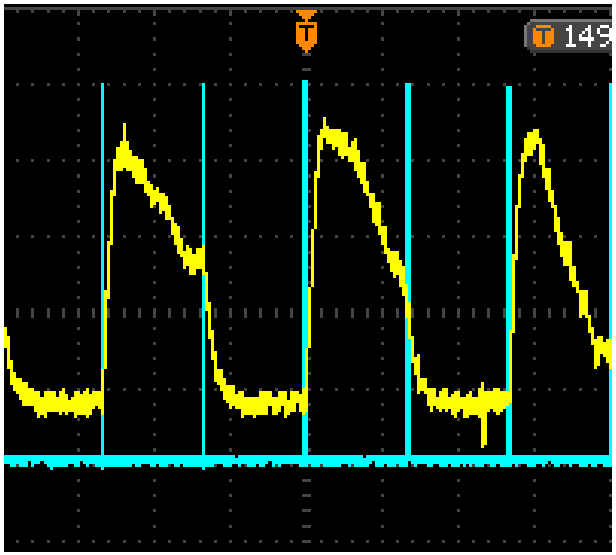


Tracking Error

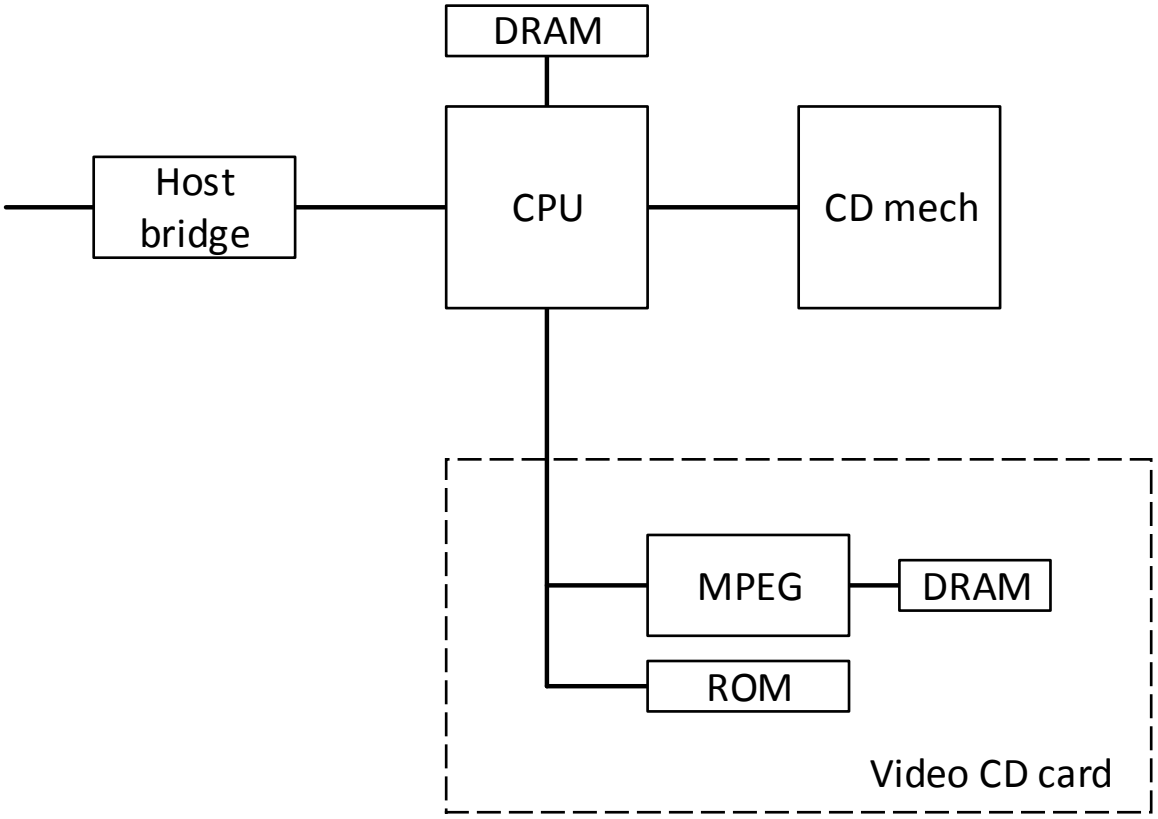
Regular data area



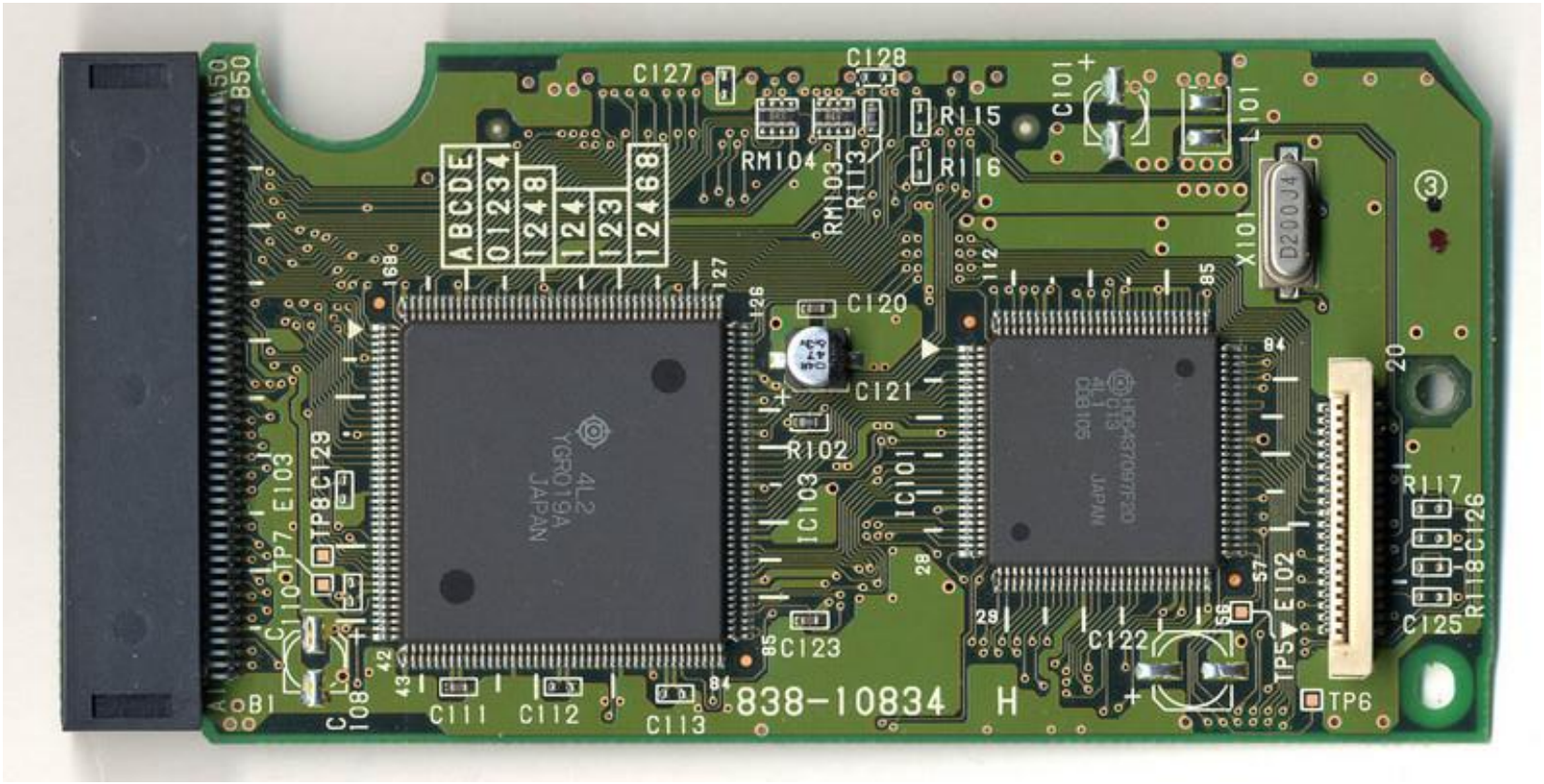
Ring area



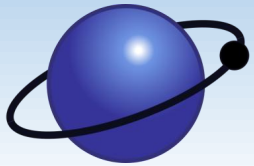
CDB architecture



CDB module

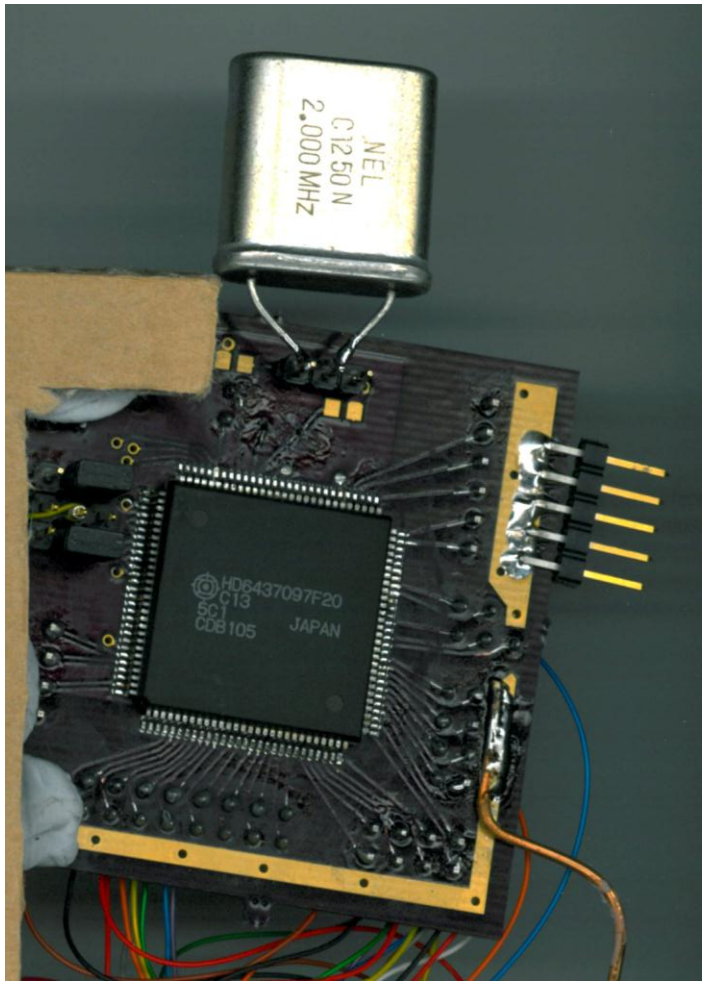

C127
C128
C101
R115
R116
RM104
RM103
R113
L101
X101
D200J4
C120
C121
R102
IC103
IC101
C123
C122
E102
TP5
TP6
C125
R117
R118
C126
C129
TP8
TP7
E103
C110
C108
C111
C112
C113
838-10834 H
ABCDE
01234
1248
124
123
12468

SH-1 boot modes

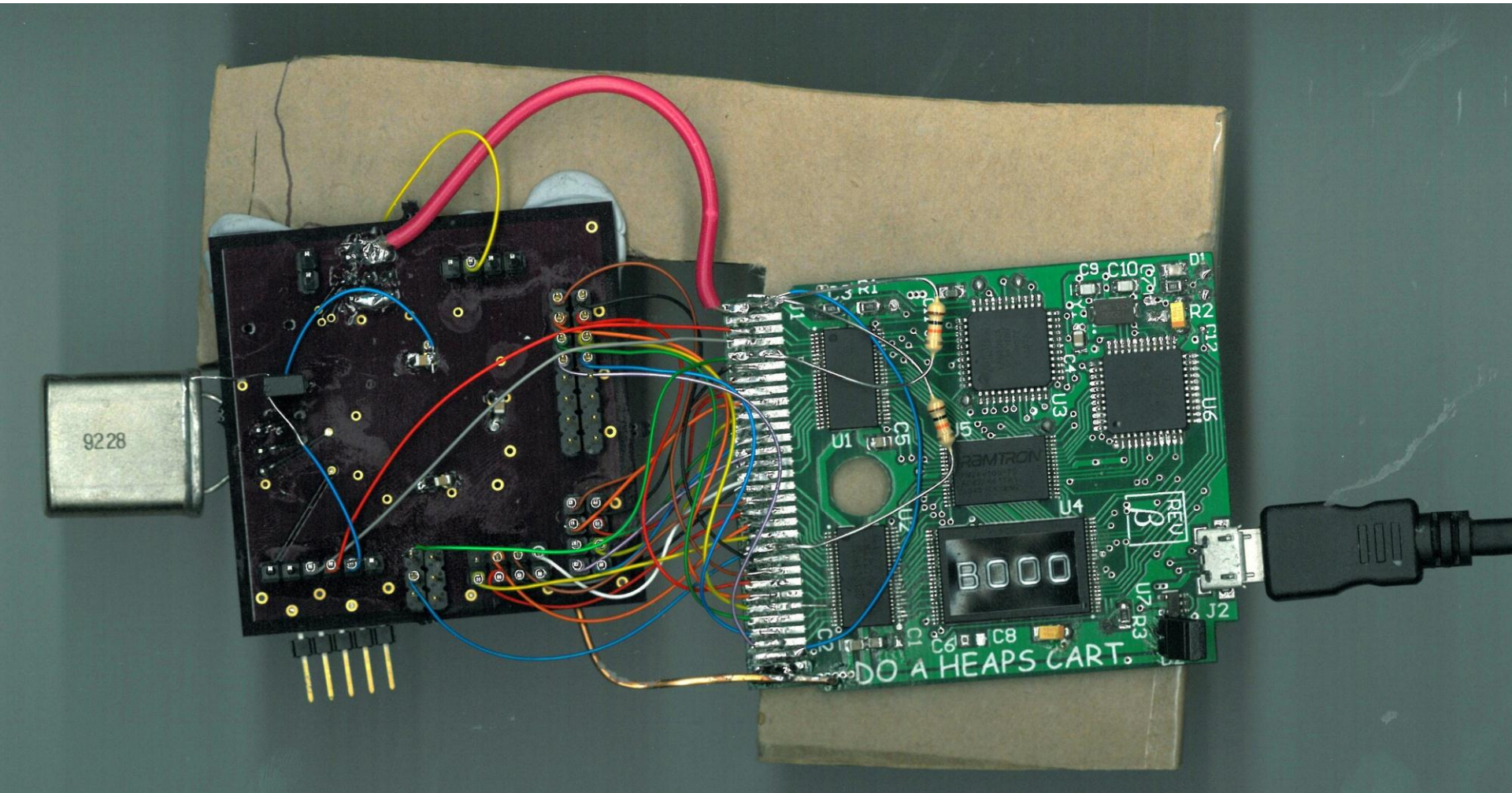


MD2	MD1	MD0	Operating Mode	On-Chip ROM	Bus Size in Area 0
0	0	0	MCU mode	Disabled	8 bits
0	0	1			16 bits
0	1	0		Enabled*1	
0	1	1	(Reserved)		
1	0	0			
1	0	1			
1	1	0			
1	1	1	PROM mode*2		

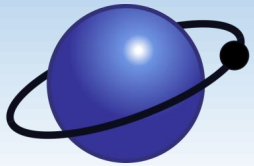
Dumping the SH-1



Dumping the SH-1

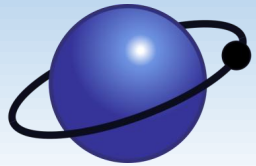


Victory!



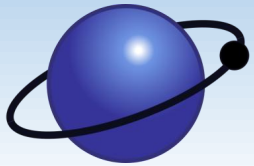
```
03d0: 0000 09d2 0000 09d2 0000 09d2 0000 09d2  ................
03e0: 0000 09d2 0000 09d2 0000 09d2 0000 09d2  ................
03f0: 0000 09d2 0000 09d2 0000 09d2 0000 09d2  ................
0400: 2243 6f70 7972 6967 6874 2028 4329 2048  "Copyright (C) H
0410: 6974 6163 6869 2c20 4c74 642e 2031 3939  itachi, Ltd. 199
0420: 3322 0009 e2e0 9ba6 d039 400e 3ee8 4e2e  3".......9@.>.N.
0430: b080 0009 d0e9 401e b088 0009 b094 0009  ......@.........
0440: b0b2 0009 b0c0 0009 b094 0009 b0d5 0009  ................
0450: d030 401e b0dc 0009 b0e8 0009 d02e 401e  .0@...........@.
0460: b0ea 0009 b12a 0009 d0dc 401e b0b0 0009  .....*....@.....
```

Where to start?



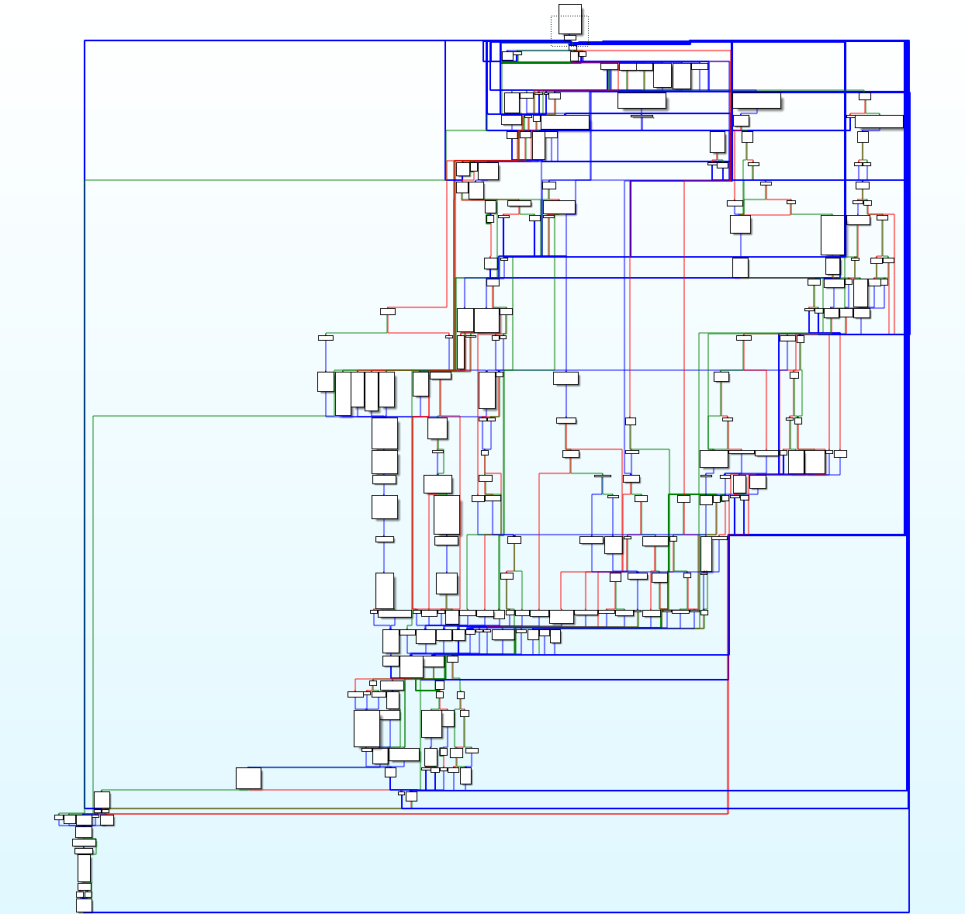
```
ROM:00000004 off_4:          .data.l init_sp          ; DATA XREF: ROM:00005FA0↓o
ROM:00000008 off_8:          .data.l resetvect        ; DATA XREF: ROM:00007BED↓o
ROM:0000000C                 .data.l init_sp
ROM:00000010                 .data.l vect_illegalinsn
ROM:00000014                 .data.l vect_rsvd
ROM:00000018 off_18:         .data.l vect_illegalslotinsn
ROM:0000001C                 .data.l vect_rsvd
ROM:00000020 off_20:         .data.l vect_rsvd
ROM:00000024                 .data.l vect_cpuaddr_error
ROM:00000028                 .data.l vect_dmaadrr_err
ROM:0000002C                 .data.l vect_nmi
ROM:00000030                 .data.l vect_ubrk
ROM:00000034                 .data.l vect_rsvd
```

More vectors

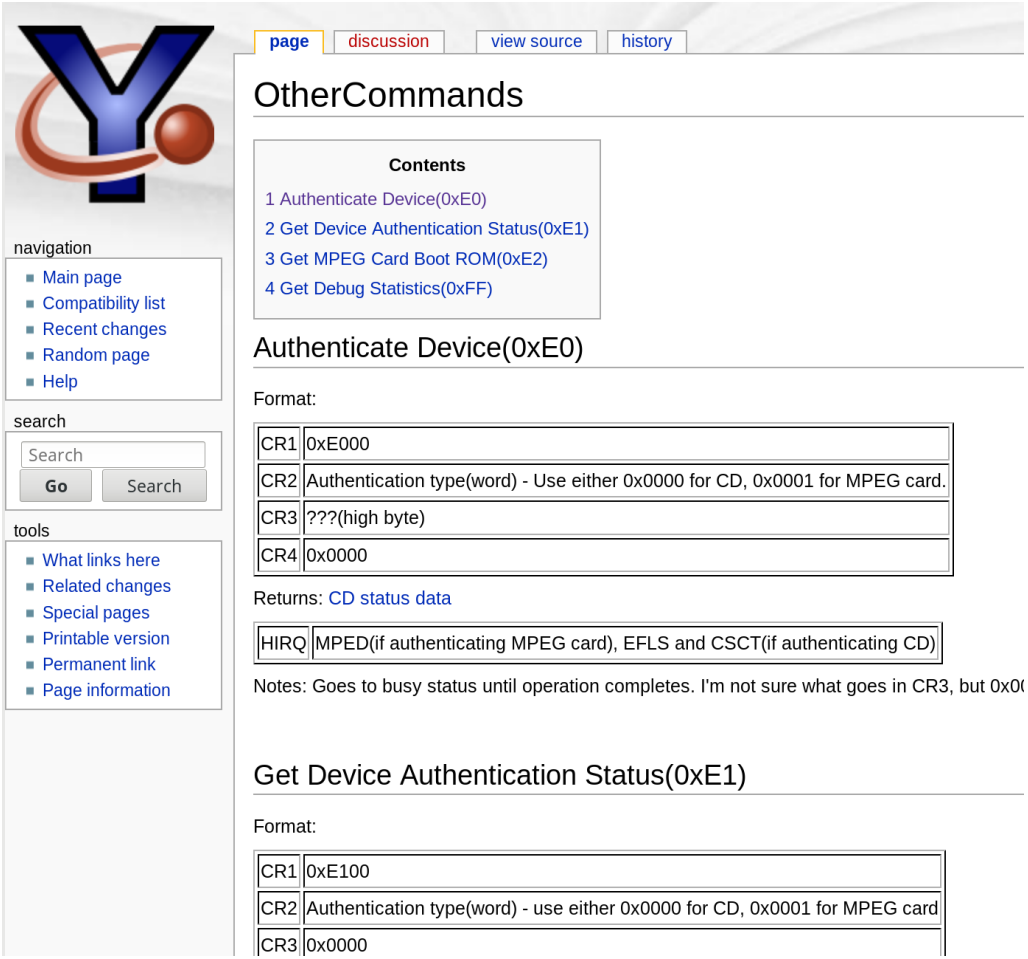


```
ROM:00000100                                            ; ROM:off_7374↓o
ROM:00000104                    .data.l irqGen
ROM:00000108                    .data.l irqGen
ROM:0000010C                    .data.l irqGen
ROM:00000110                    .data.l irqGen
ROM:00000114                    .data.l irqGen
ROM:00000118 off_118:           .data.l irq6            ; DATA XREF: ROM:off_675C↓o
ROM:0000011C                    .data.l irq7
ROM:00000120                    .data.l irqDMAC0DEI0
ROM:00000124                    .data.l vect_rsvd
ROM:00000128                    .data.l irqDMAC1DEI1
ROM:0000012C                    .data.l vect_rsvd
ROM:00000130                    .data.l irqDMAC2DEI2
ROM:00000134                    .data.l vect_rsvd
ROM:00000138                    .data.l irqDMAC3DEI3
ROM:0000013C                    .data.l vect_rsvd
ROM:00000140                    .data.l irqGen
ROM:00000144                    .data.l irqGen
ROM:00000148                    .data.l irqGen
ROM:0000014C                    .data.l vect_rsvd
ROM:00000150                    .data.l irqGen
ROM:00000154                    .data.l irqGen
ROM:00000158                    .data.l irqGen
ROM:0000015C                    .data.l vect_rsvd
ROM:00000160                    .data.l irqITU2IMIA2    ; mpeg audio
ROM:00000164                    .data.l irqITU2IMIB2    ; mpeg video
ROM:00000168                    .data.l irqGen
```

Spaghetti Central



The key to everything



page | discussion | view source | history

OtherCommands

Contents

navigation
- Main page
- Compatibility list
- Recent changes
- Random page
- Help

search
Search
Go | Search

tools
- What links here
- Related changes
- Special pages
- Printable version
- Permanent link
- Page information

Authenticate Device(0xE0)

Format:

CR1	0xE000
CR2	Authentication type(word) - Use either 0x0000 for CD, 0x0001 for MPEG card.
CR3	???(high byte)
CR4	0x0000

Returns: CD status data

HIRQ	MPED(if authenticating MPEG card), EFLS and CSCT(if authenticating CD)

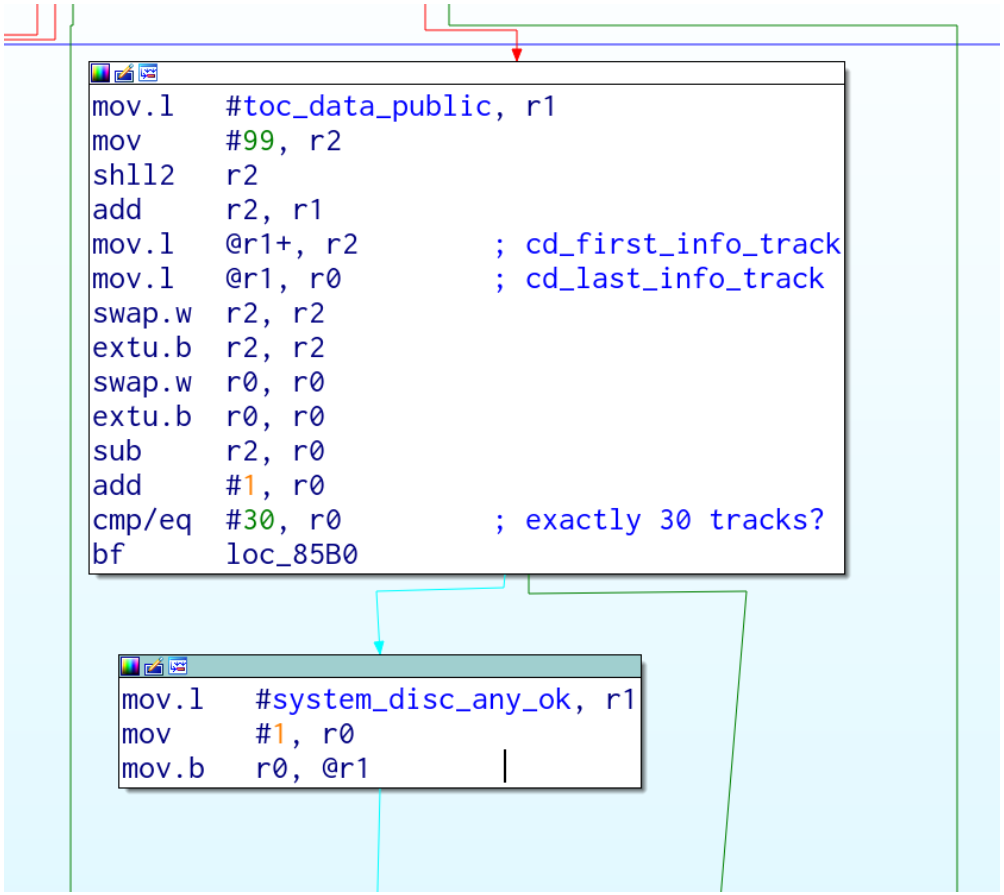
Notes: Goes to busy status until operation completes. I'm not sure what goes in CR3, but 0x0(

Get Device Authentication Status(0xE1)

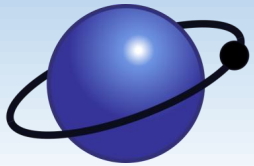
Format:

CR1	0xE100
CR2	Authentication type(word) - use either 0x0000 for CD, 0x0001 for MPEG card
CR3	0x0000

The secret backdoor!



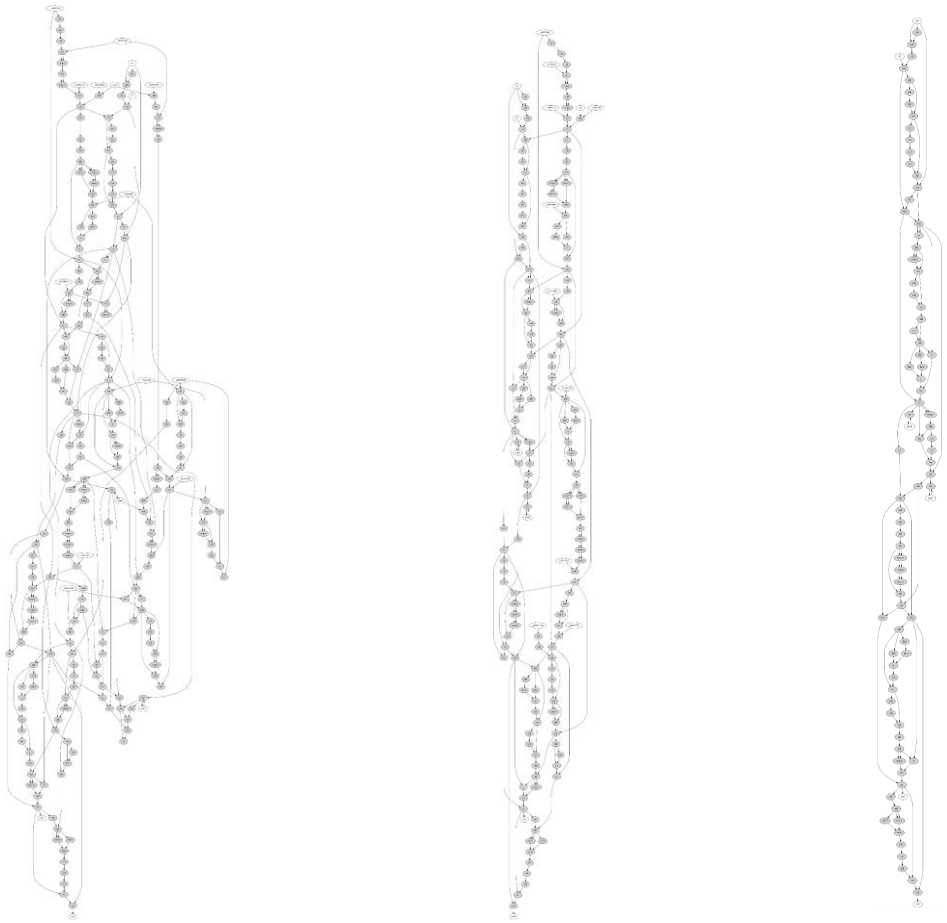
???



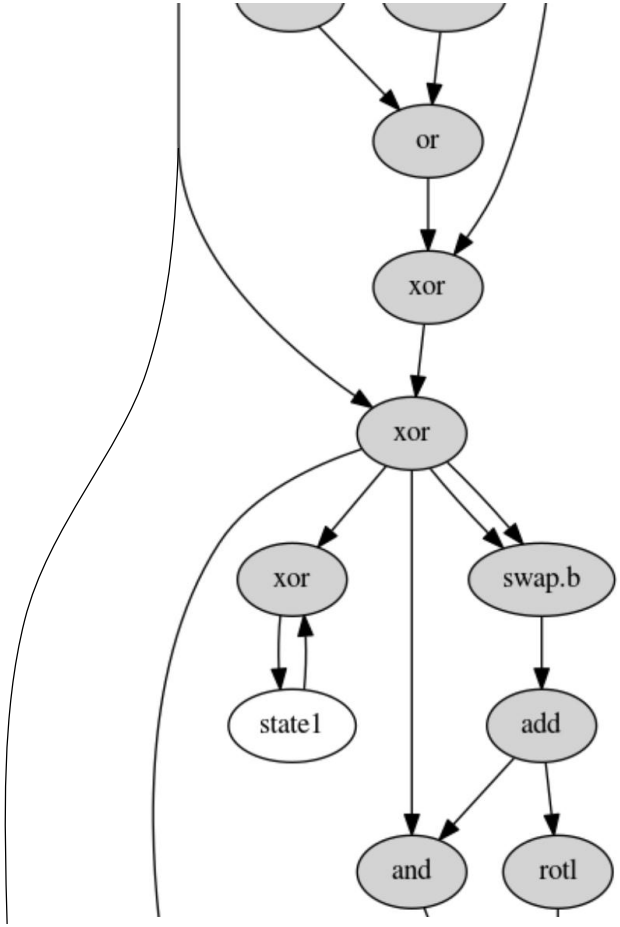
```
; ---------------------------------------------------------------------------
off_99DC:       .data.l strPKC          ; DATA XREF: pk_prep_one+2↑r
off_99E0:       .data.l strHitachi      ; DATA XREF: pk_prep_one+6↑r
strHitachi:     .data.b 'H, 'i, 't, 'a, 'c, 'h, 'i, '.
                                        ; DATA XREF: pk_prep_one+6↑o
                                        ; ROM:off_99E0↑o
strPKC:         .data.b 'P, 'u, 'b, 'l, 'i, 'c, 'K, 'e, 'y, 'C, 'i, 'p, 'h, 'e, 'r, '.
                                        ; DATA XREF: pk_prep_one+2↑o
                                        ; ROM:off_99DC↑o

; =============== S U B R O U T I N E =======================================
```

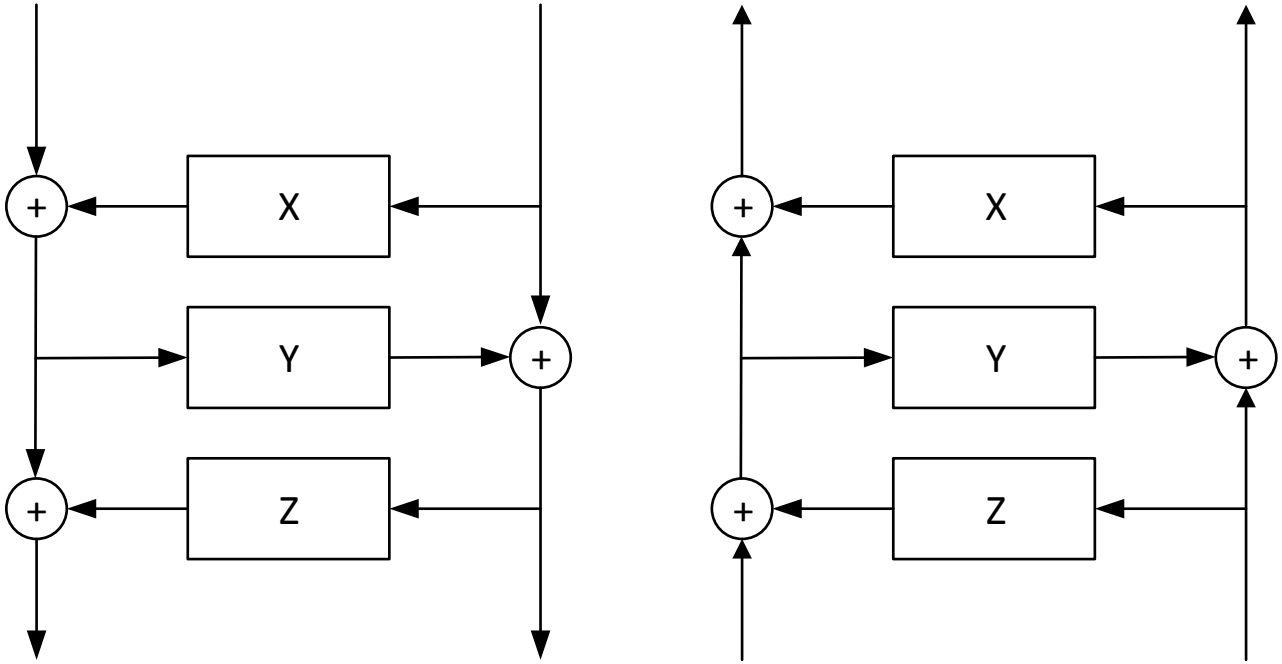
Eyeballing the round function



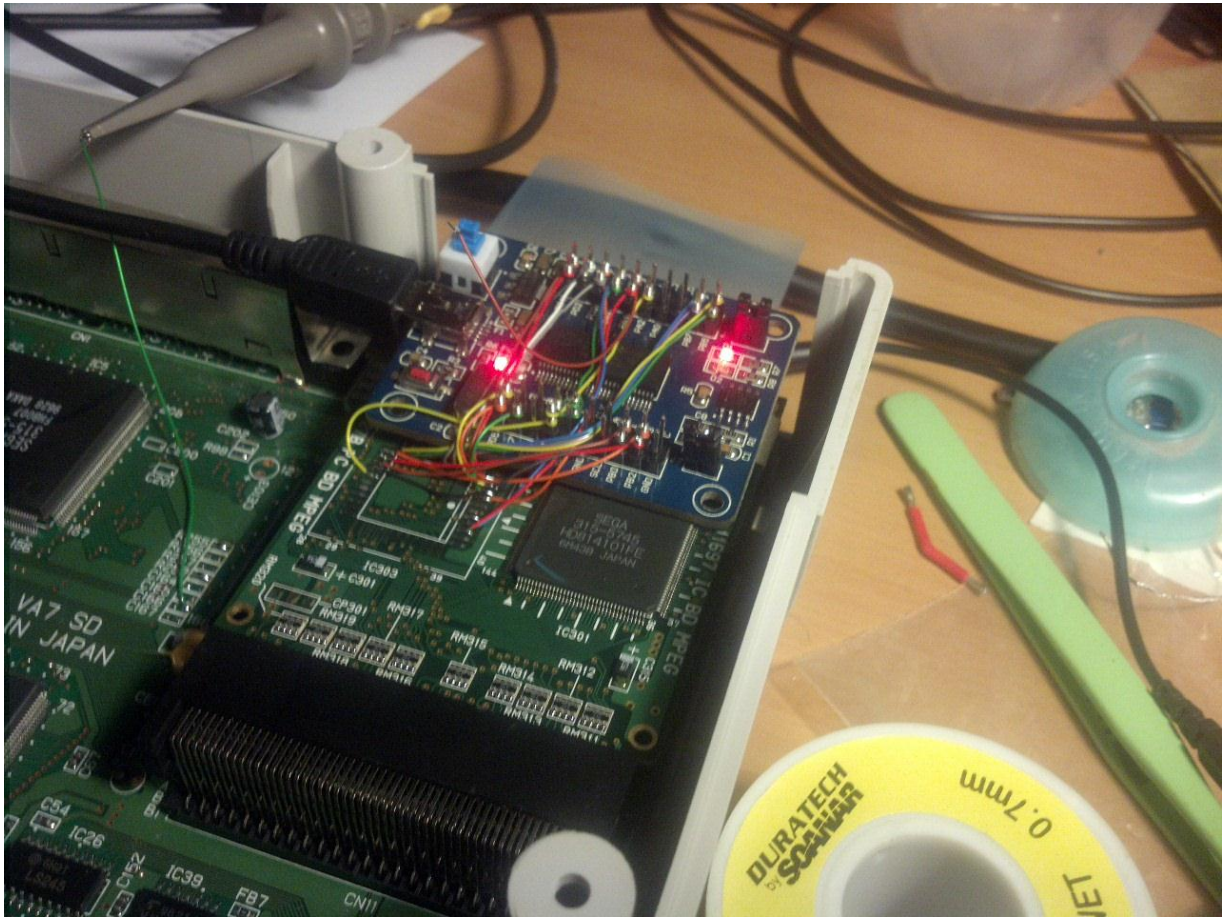
A pattern



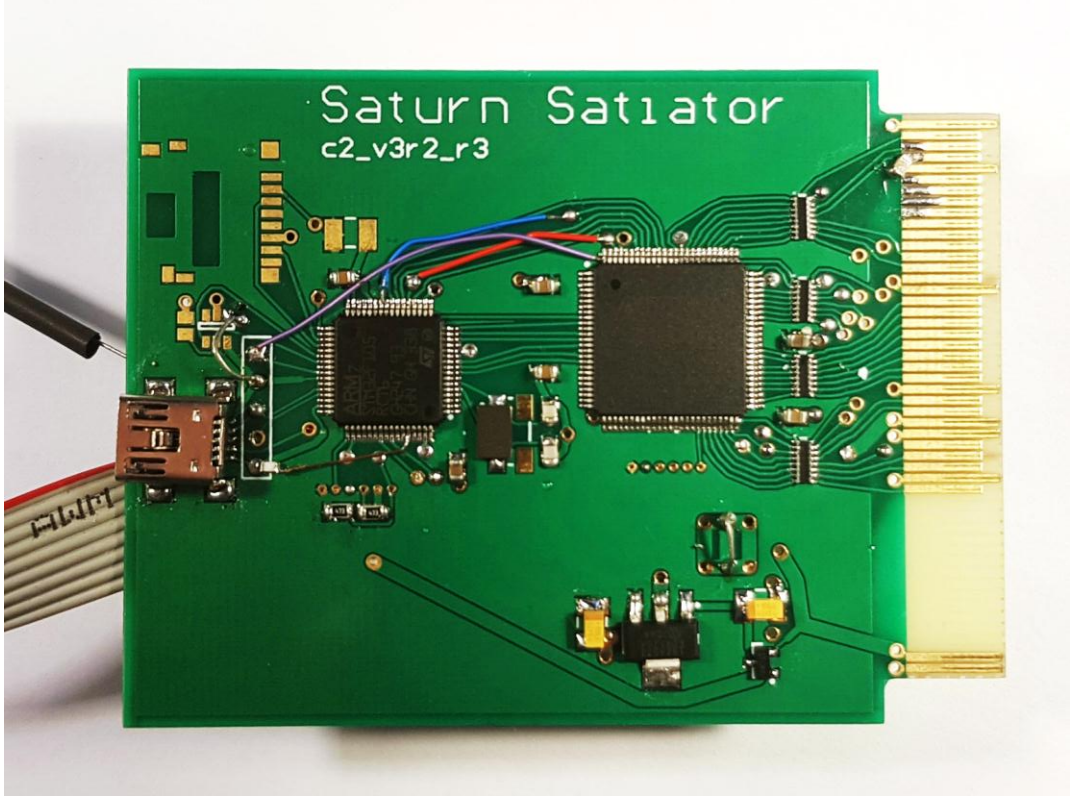
Feistel networks



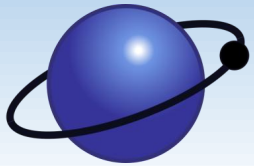
Code injection!



A new tool



Thanks



Lazerbeat

zyrobs

the Yabause team

cTrix

Estee Laird-Wah