The hitchhacker’s guide to iPhone Lightning & JTAG hacking
by stacksmashing
About me: stacksmashing

• Security researcher

• youtube.com/stacksmashing

• @ghidraninja
Thanks!
As always: Standing on the shoulders of giants

- Carlo Maragno - @CarloMara1
- Jiska Classen
- Fabian Freyer
- Caro Gross
- Lily - @bendycatus
- John - @nyan_satan
- LambdaConcept
- Elliot
Lightning?
Charging
Charging

Data transfer
DCSD Cable for iPhone/iPad/iPod Engineering & Exploit DCSD USB Cable for WL 64Bit Mijing HDD Test Fixture Engineering Cable

⭐⭐⭐⭐ - 4.0 - 1 Review 2 orders

US $17.99 - 30.29

Get coupons

Color:

Number of Pieces:
1pcs

Quantity:
1 + 540 Pieces available

Ships to United States
Kanzi Cable

The Kanzi Cable is a JTAG/SWD Cable capable of debugging CPFM 00 or 01 devices (EVT and DVT devices) which have the Lightning port, using software called Astris. It can be connected to another SWD debugger, using the SWD port, and it can also do UART/Serial. They can be purchased from obscure markets. There are two known types of the Kanzi cable. The normal version and a prototype version with PROTO etched to it.

Uses

Dumping the SecureROM

One use of the cable is dumping the SecureROM from devices. This can be done using commands such as this one.

This hardware article is a "stub", an incomplete page. Please add more content to this article and remove this tag.

Categories: Article stubs | Cables
BONobo JTAG/SWD Debug Cable

OUT-OF-STOCK

€749.00
Tax excluded

iPhone debugging requires proper tools.

The Bonobo cable connects to your target through Lightning and allows CPU debugging through JTAG/SWD using OpenOCD + AArch64 GDB. Among others, you can: access all CPUs and registers, single step, put hardware breakpoints, dump memory, etc... Perfect for security research.

The target serial console can be accessed on the control PC through Minicom (Boot prompt), as well as Lightning USB (For DFU, USB exploitation, demote, etc.)

More Here

Demonstration:
BONOBO JTAG/SWD DEBUG CABLE

€749.00

Tax excluded

iPhone debugging requires proper tools.

The Bonobo cable connects to your target through Lightning and allows CPU debugging through JTAG/SWD using OpenOCD + AArch64 GDB. Among others, you can: access all CPUs and registers, single step, put hardware breakpoints, dump memory, etc... Perfect for security research.

The target serial console can be accessed on the control PC through Minicom (Boot prompt), as well as Lightning USB (For DFU, USB exploitation, demote, etc.)

More Here

Demonstration:
Let’s build our own!
SDQ or IDBUS
SDQ or IDBUS
Tristar
UART
USB
JTAG
What do you want?
What do you want?

Speak USB plz
SDQAnalyzer

This is a simple analyzer for the SDQ protocol that is used on a lot of Apple products, for example:

- Lightning
- MagSafe
- iPhone/iPad battery

Documentation for the Saleae Logic Analyzer SDK can be found here:
http://github.com/saleae/SampleAnalyzer
What’re SDQ and IDBUS?

These 2 terms are often referred as kind of synonyms. For convenience, I’ll only use term IDBUS from now on, as it seems more correct to me (and that’s how this technology called in the THS7383 datasheet)

So, IDBUS - is a digital protocol used for negotiations between Tristar and HiFive. Very similar to Onewire protocol

Now we can play

Let’s sniff the negotiations between Tristar and HiFive. Take a logic analyzer, a Lightning male-to-female passthrough breakout board, some accessory (normal Lightning to USB cable would fit just fine) and of course some device with Lightning port

First connect logic analyzer’s channels to both ID lines of the breakout (pins 4 and 8) and connect the
SDQ or IDBUS

Simple one-wire protocol
SDQ or IDBUS

Simple one-wire protocol
SDQ or IDBUS
SDQ or IDBUS
SDQ or IDBUS

0x0F
SDQ or IDBUS
SDQ or IDBUS

0x0F Break
SDQ or IDBUS

0x0F  Break  0x79
• Runs at 3.3V
• Can easily implement fast protocols using PIO
• Is actually available 😴
But we need a Lightning connector...
OKCS Originals Extension Cable 1 Metre Compatible with iPhone 11, 11 Pro, 11 Max, XR, XS, XS Max, X, 8, 8 Plus, 7, 7 Plus etc. - White

Colour Name: 8 pin extender cable.
Black - GND
White - L0p
Green - L0n
Yellow - ID0
Red - 5V
Purple - L1n
Orange - L1p
Blue - ID1
Building a simple DCSD cable
Who’s there?

74 00 07 20
Who’s there? 74 00 07 20

Please speak USB & UART 75 20 00 00 00 00 00 00 AA
JTAG...
JTAG...
... actually SWD
The plan!
The plan!

Set Lightning to JTAG via SDQ

It doesn't work

It doesn't work
Who’s there?

Please speak USB, UART & SWD!
The plan!

Set Lightning to JTAG via SDQ

It doesn't work.

It doesn't work.
The plan!

Set Lightning to JTAG via SDQ

Connect debug probe with bonobo configs

It doesn't work

It doesn't work
The plan!

Set Lightning to JTAG via SDQ

Connect debug probe with bonobo configs
The plan!

Set Lightning to JTAG via SDQ

Connect debug probe with bonobo configs

It doesn’t really work
The plan!

Set Lightning to JTAG via SDQ

Connect debug probe with bonobo configs

It doesn’t really work

It doesn’t really work
[J-Link>connect
Device "CORTEX-A12" selected.

Connecting to target via SWD
Found SW-DP with ID 0x4BA02477
<table>
<thead>
<tr>
<th>TargetName</th>
<th>Type</th>
<th>Endian</th>
<th>TapName</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>mem_ap</td>
<td>little</td>
<td>iphone.cpu</td>
<td>running</td>
</tr>
<tr>
<td>1</td>
<td>mem_ap</td>
<td>little</td>
<td>iphone.cpu</td>
<td>running</td>
</tr>
<tr>
<td>2</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
<tr>
<td>3</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
<tr>
<td>4</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
<tr>
<td>5</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
<tr>
<td>6</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
<tr>
<td>7</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
<tr>
<td>8</td>
<td>aarch64</td>
<td>little</td>
<td>iphone.cpu</td>
<td>unknown</td>
</tr>
</tbody>
</table>
How do you debug this?
You read the SWD spec
You logic analyze it
You fix the logic analyzer
You find OpenOCD does not handle SWD WAITs
You implement SWD by hand
You find a single SWD bit was wrong
You find a single SWD bit was wrong 😐
nc 127.0.0.1 4444
Open On-Chip Debugger
nc 127.0.0.1 4444

Open On-Chip Debugger

~
gdb

GNU gdb (GDB) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-apple-darwin21.5.0".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
Find the GDB manual and other documentation resources online at:

For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb) target remote 127.0.0.1:3333
```bash
nc 127.0.0.1 4444
```

```bash
~ $ gdb
GNU gdb (GDB) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show libraries" for more information.
This GDB was configured as "x86_64-unknown-linux-gnu".
Type "show configuration" for more details.
For bug reporting instructions, read the bug reporting guide
Find the GDB manual and other information online at:
For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb) target remote 127.0.0.1:3333
```
• SDQ bridge
• Supporting DCSD & JTAG
• iPhone compatible debug probe
• Fully open-source
Kanzi
Kanzi

Bonobo
Kanzi

Kong

Chimp

Bonobo

Tamarin Cable
Good morning!

1: JTAG mode
2: DCSO mode
3: Reset device
4: Reset and enter DFU mode
R: Reset Tamarin cable
Releases

• You can find everything on github.com/stacksmashing
  • Tamarin Cable Firmware
  • Forked OpenOCD
  • Forked swd-analyzer
  • SDQAnalyzer
Thank you!

• youtube.com/stacksmashing
• @ghidraninja
• contact@stacksmashing.net