Bootloader and Linux kernel debugging on ARM board with OpenOCD

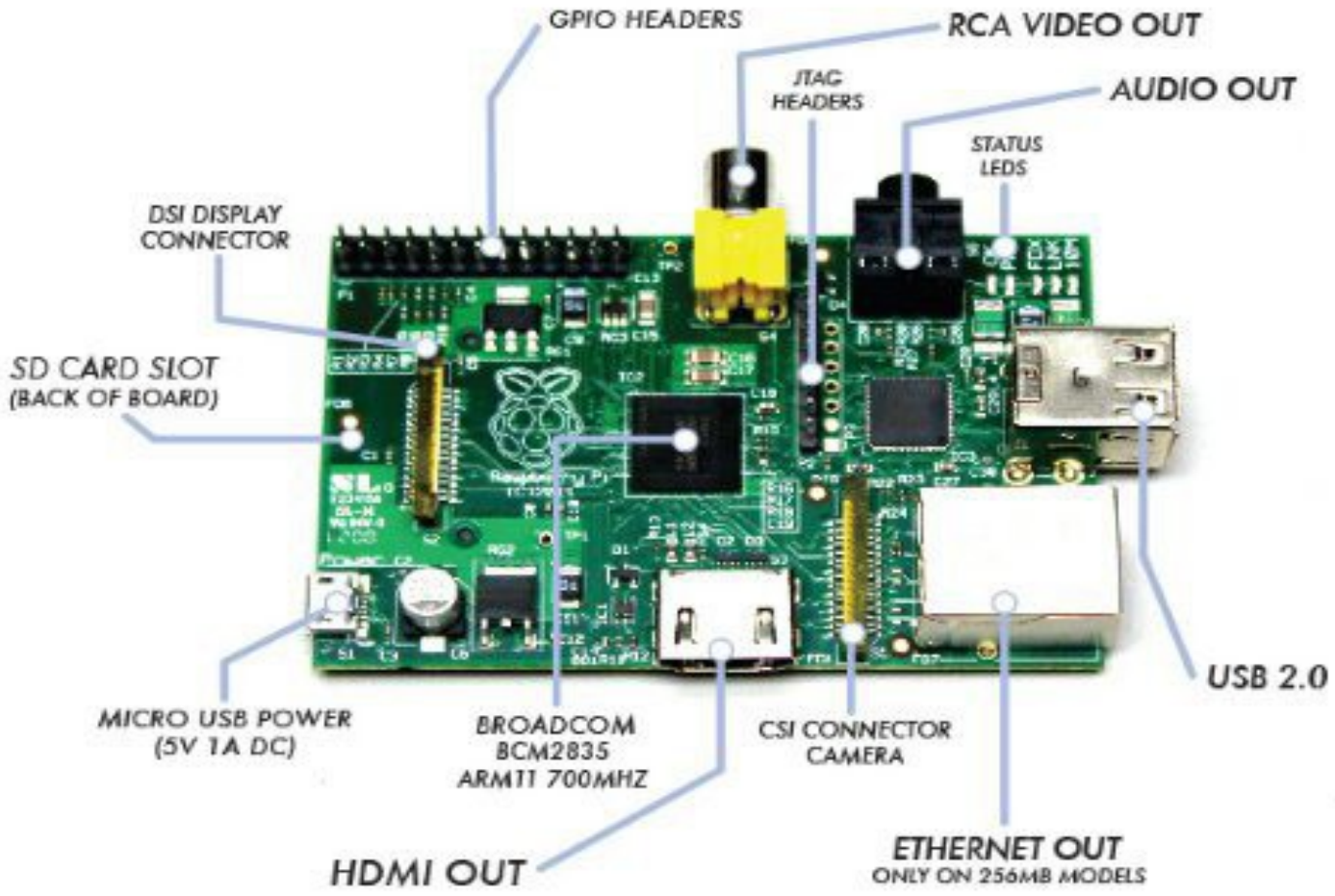
LVEE 2014
Vladimir Zapolskiy
vladimir_zapolskiy@mentor.com

mentor embedded

mentor.com/automotive

Android is a trademark of Google Inc. Use of this trademark is subject to Google Permissions.
Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.
Qt is a registered trade mark of Digia Plc and/or its subsidiaries. All other trademarks mentioned in this document are trademarks of

ARM + Linux


GPIO HEADERS
RCA VIDEO OUT
JTAG HEADERS
AUDIO OUT
STATUS LEDS
DSI DISPLAY CONNECTOR
SD CARD SLOT
(BACK OF BOARD)
USB 2.0
MICRO USB POWER
(5V 1A DC)
BROADCOM
BCM2835
ARM11 700MHZ
CSI CONNECTOR
CAMERA
HDMI OUT
ETHERNET OUT
ONLY ON 256MB MODELS

Development Tasks

Development Tasks

- port bootloader and Linux kernel to a new ARM SoC

Development Tasks

- port bootloader and Linux kernel to a new ARM SoC
- port bootloader and Linux kernel to a new ARM powered board

Development Tasks

- port bootloader and Linux kernel to a new ARM SoC
- port bootloader and Linux kernel to a new ARM powered board
- add new features of arbitrary nature into bootloader or kernel

Development Tasks

- port bootloader and Linux kernel to a new ARM SoC
- port bootloader and Linux kernel to a new ARM powered board
- add new features of arbitrary nature into bootloader or kernel
- fix a bug in bootloader or kernel

Development Tasks

- port bootloader and Linux kernel to a new ARM SoC
- port bootloader and Linux kernel to a new ARM powered board
- add new features of arbitrary nature into bootloader or kernel
- fix a bug in bootloader or kernel
- get understanding how bootloader or kernel works in runtime

Debugging Tools and Methods

Debugging Tools and Methods

- add change, update firmware, test and repeat

Debugging Tools and Methods

- add change, update firmware, test and repeat
- full or partial hardware virtualization

Debugging Tools and Methods

- add change, update firmware, test and repeat
- full or partial hardware virtualization
- use special software debugging frameworks
  - blinking leds
  - KGDB over serial line
  - etc.

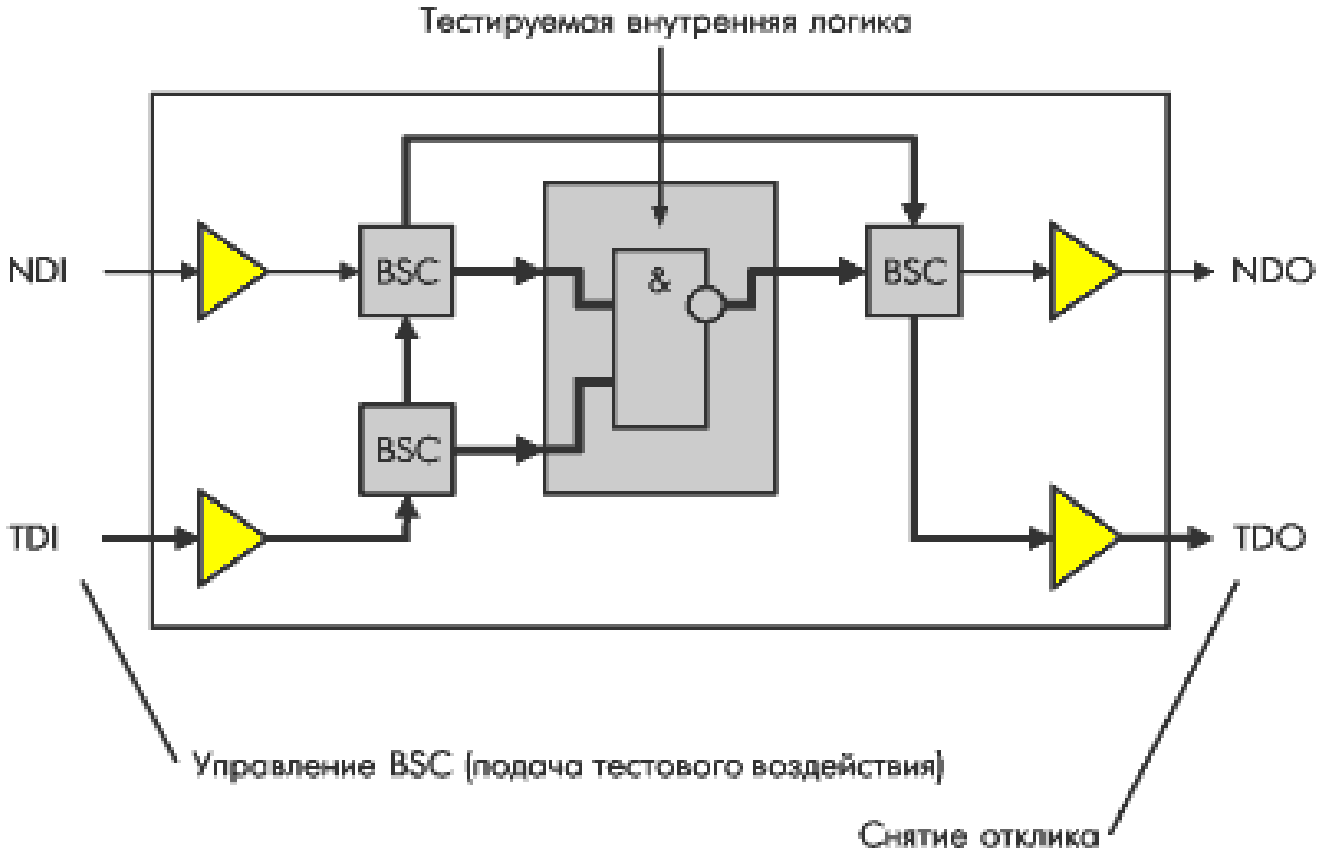
Debugging Tools and Methods

- add change, update firmware, test and repeat
- full or partial hardware virtualization
- use special software debugging frameworks
  - blinking leds
  - KGDB over serial line
  - etc.
- software or hardware dumps for connectivity and protocol debugging

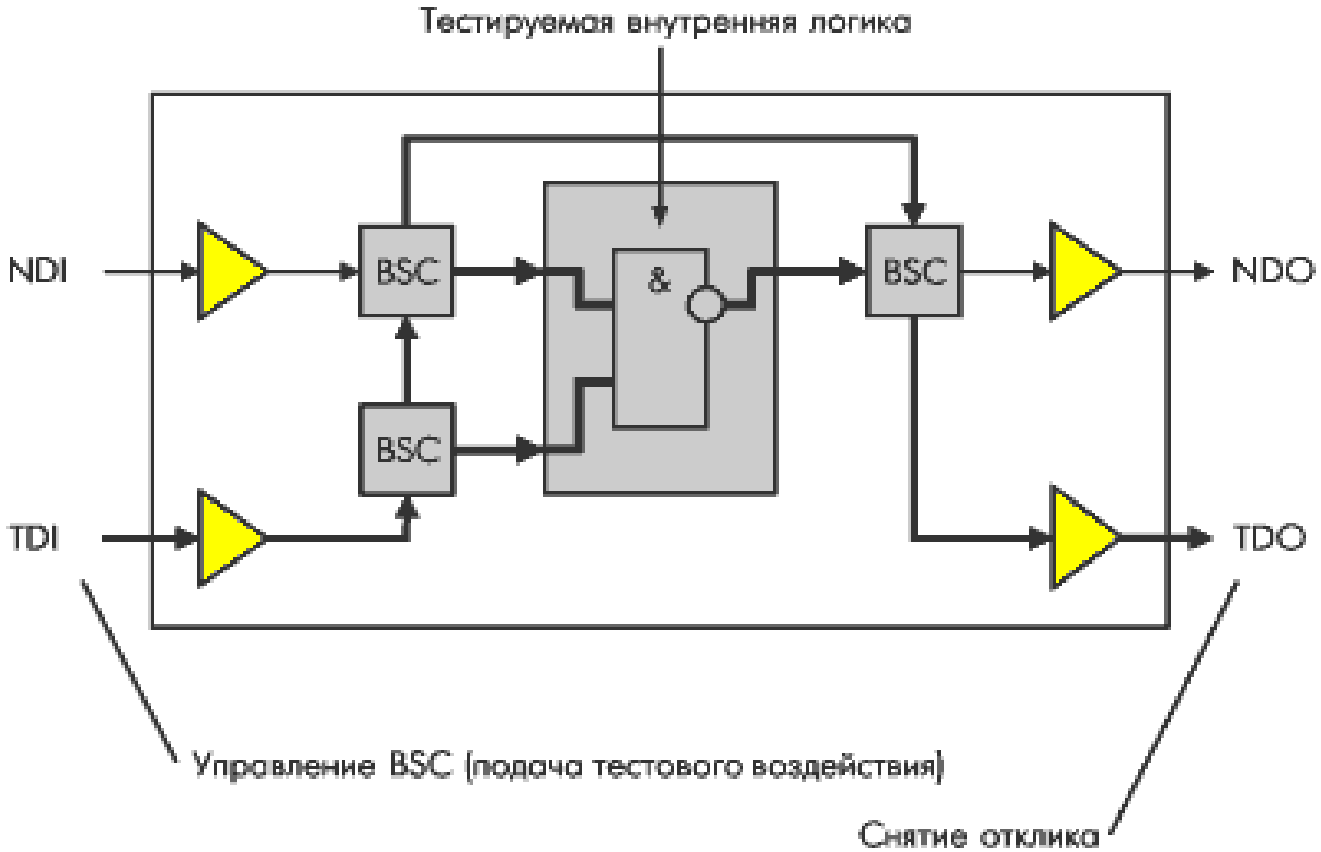
Debugging Tools and Methods

- add change, update firmware, test and repeat
- full or partial hardware virtualization
- use special software debugging frameworks
  - blinking leds
  - KGDB over serial line
  - etc.
- software or hardware dumps for connectivity and protocol debugging
- boundary scan testing of integrated circuits / IEEE 1149

JTAG / IEEE 1149.x

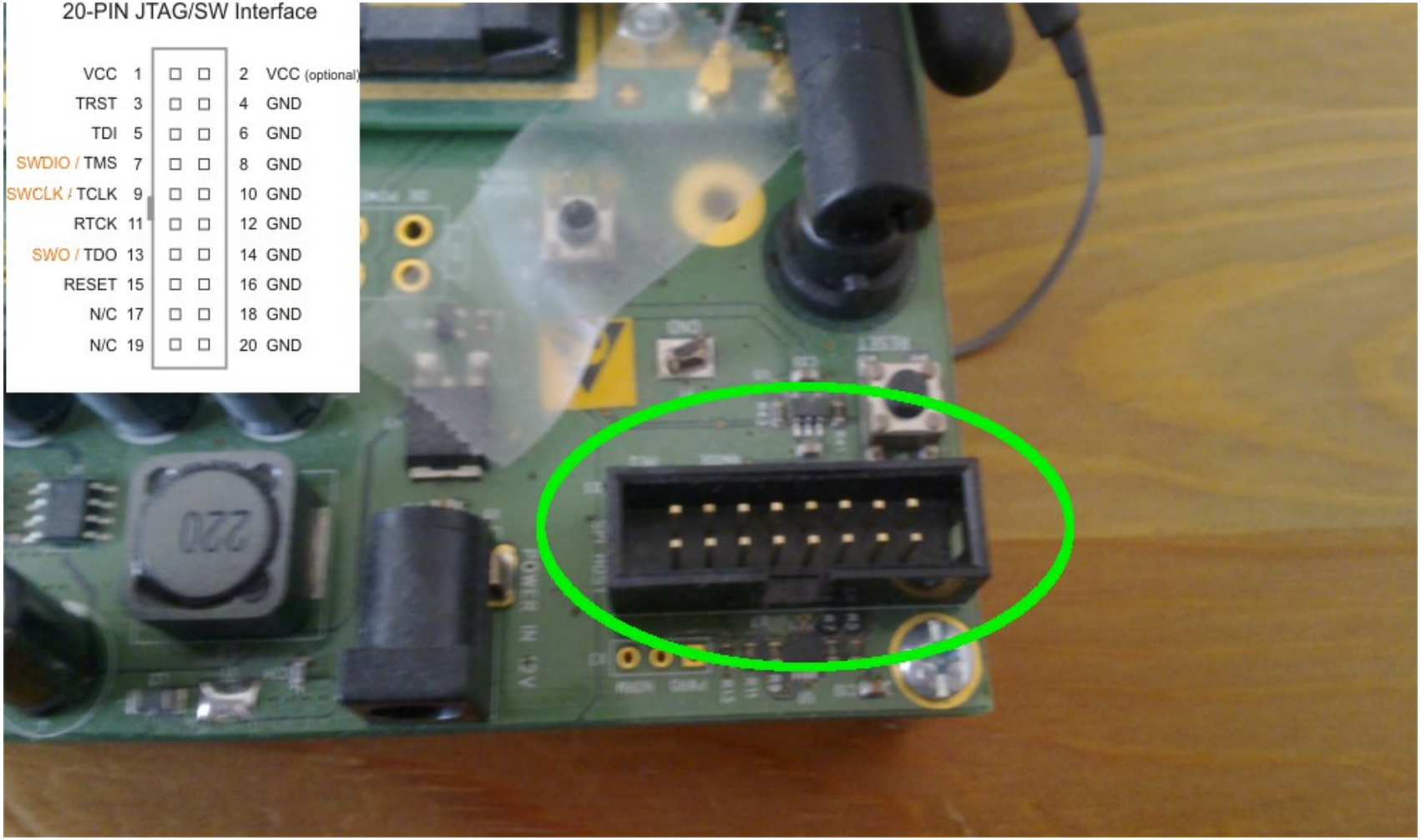


JTAG / IEEE 1149.x

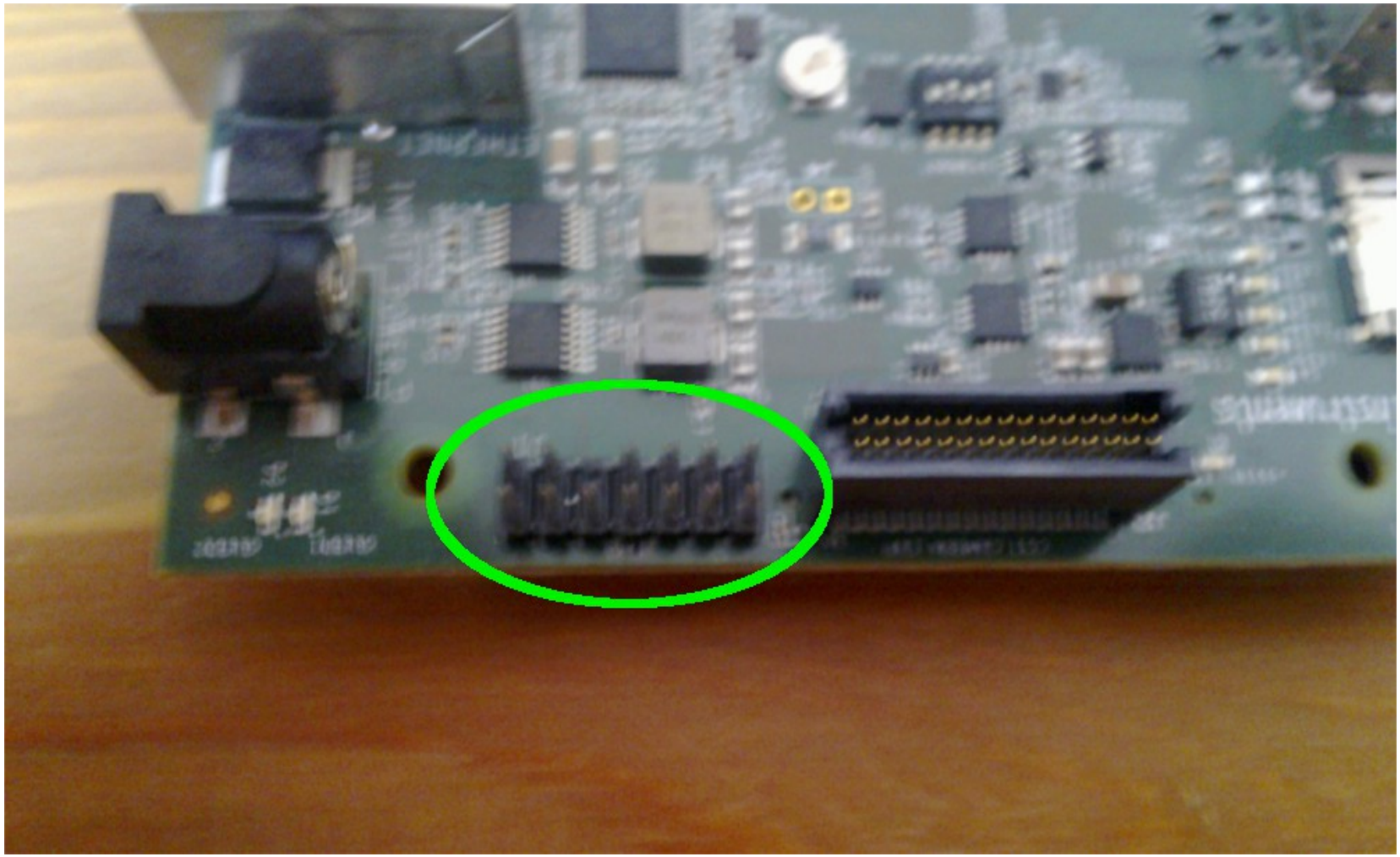


Too complicated for software developer!

JTAG connector 20 pin

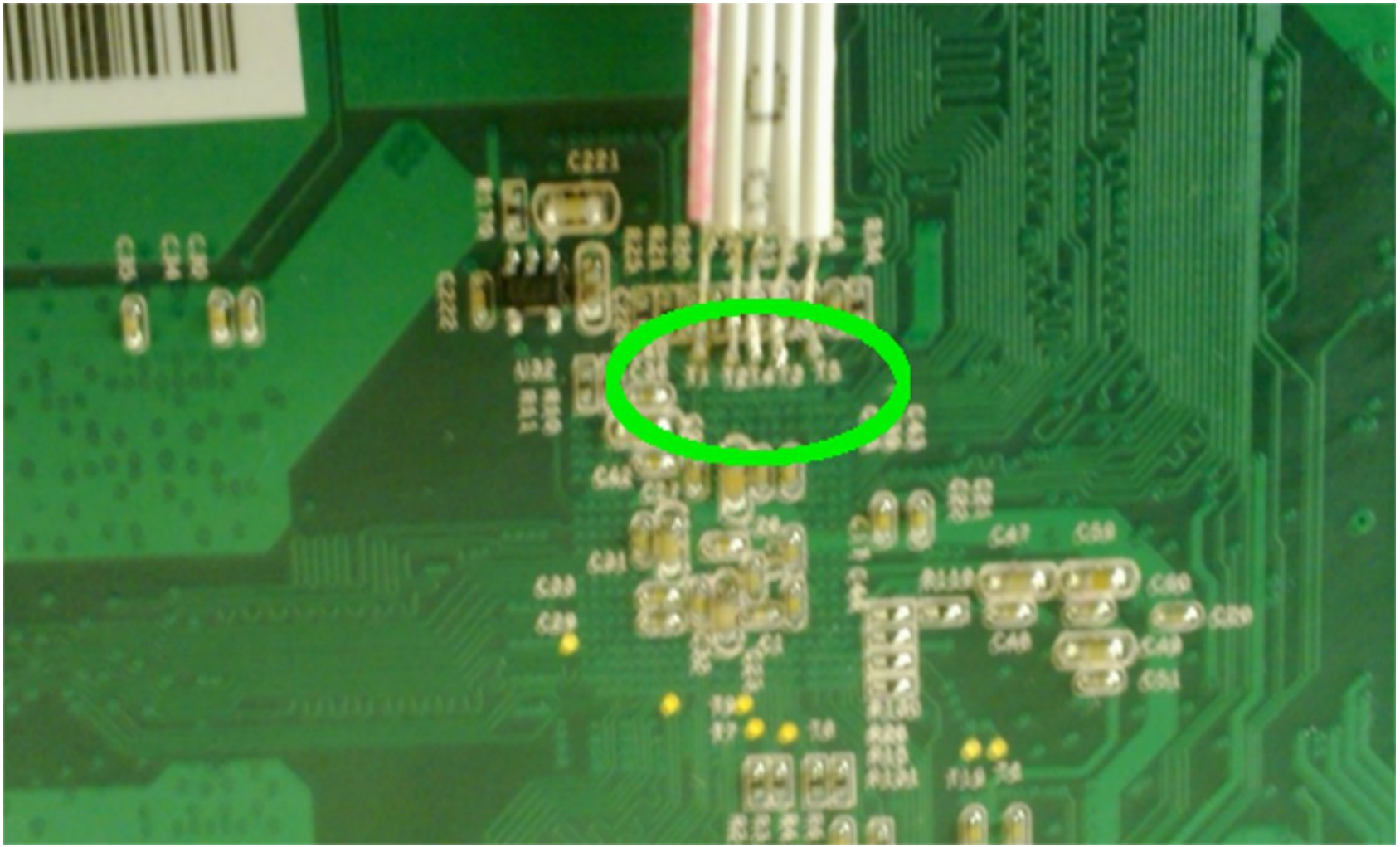


JTAG connector 14 pin

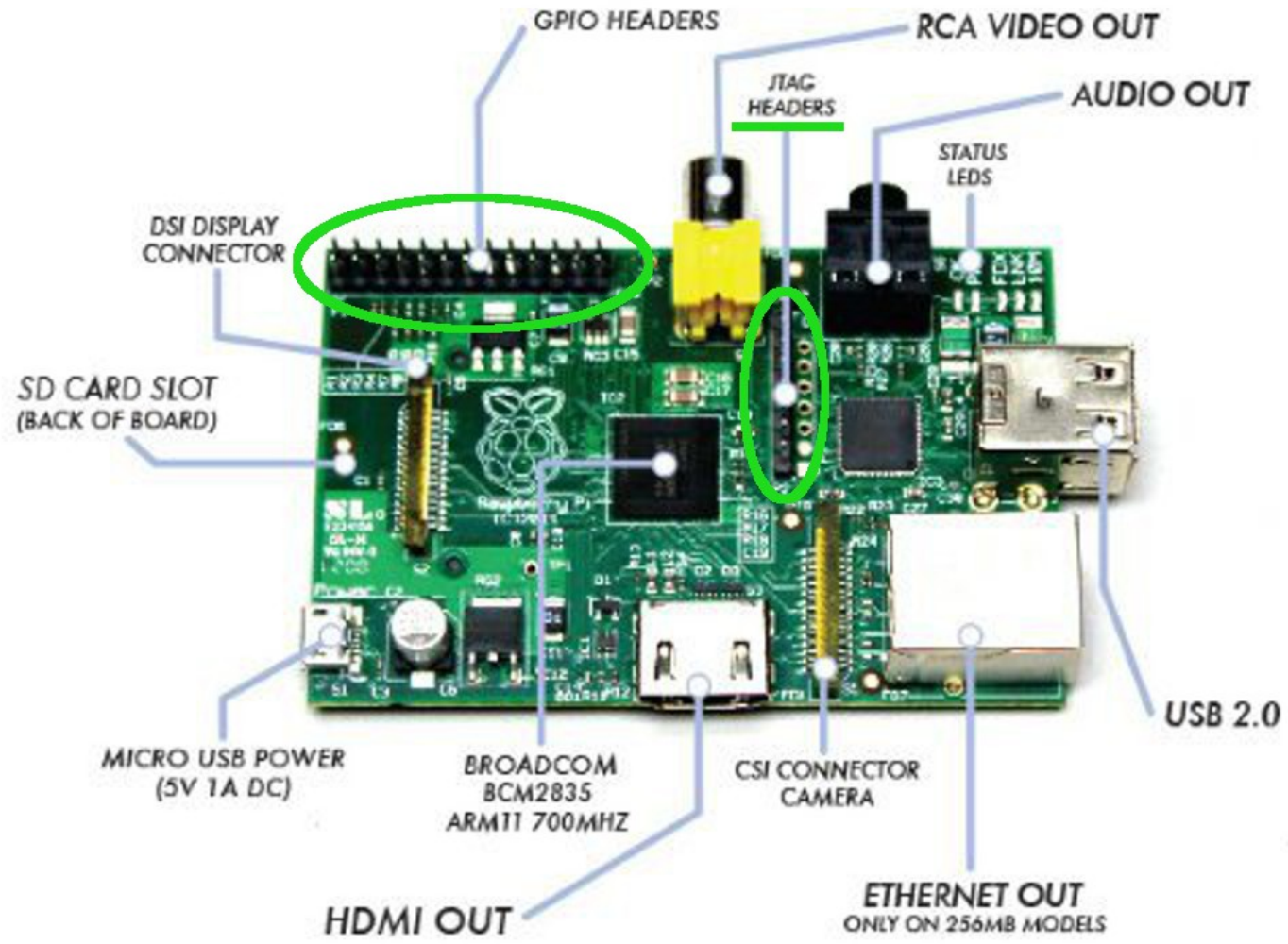


JTAG connector 10 pin

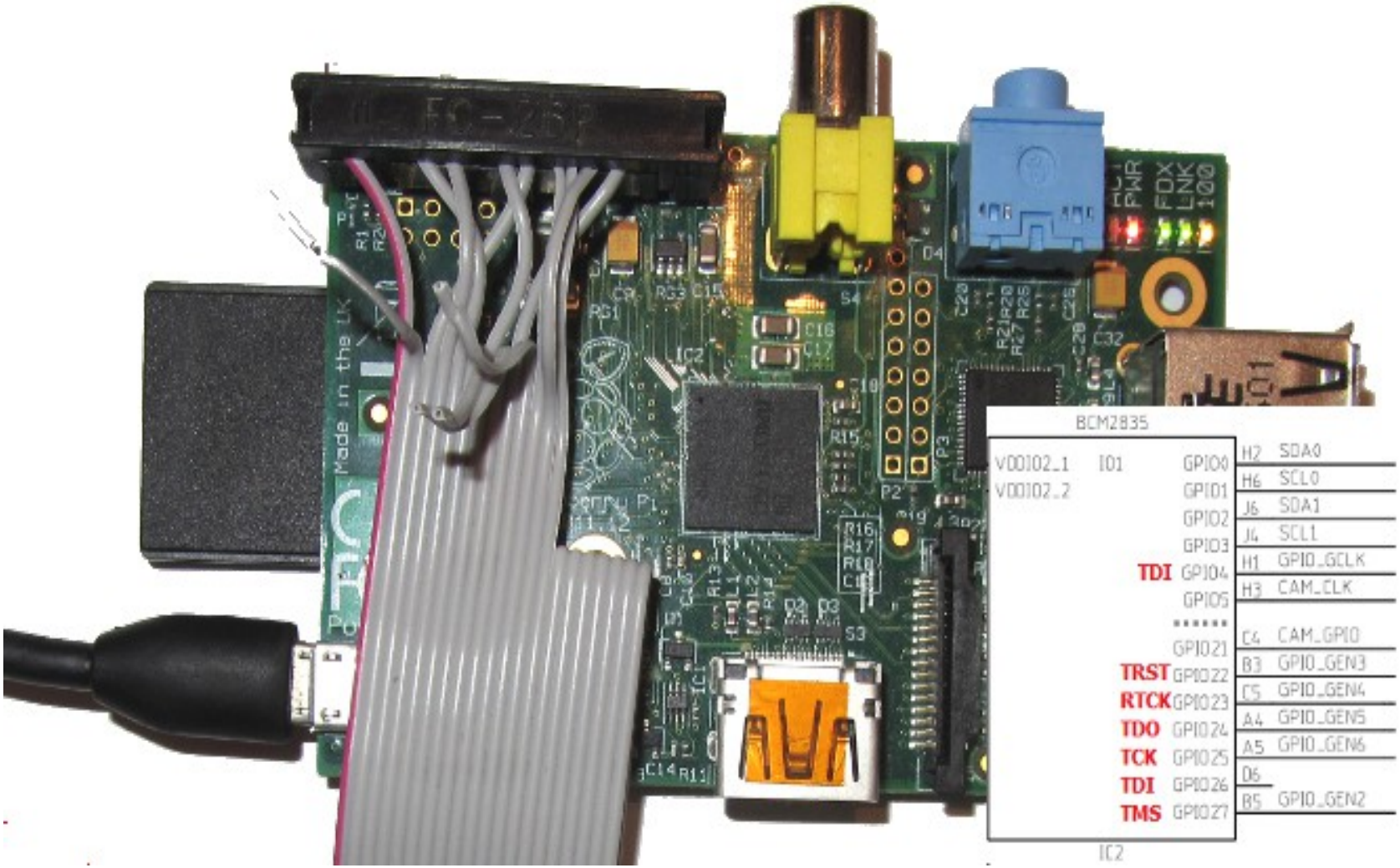
No JTAG connector!



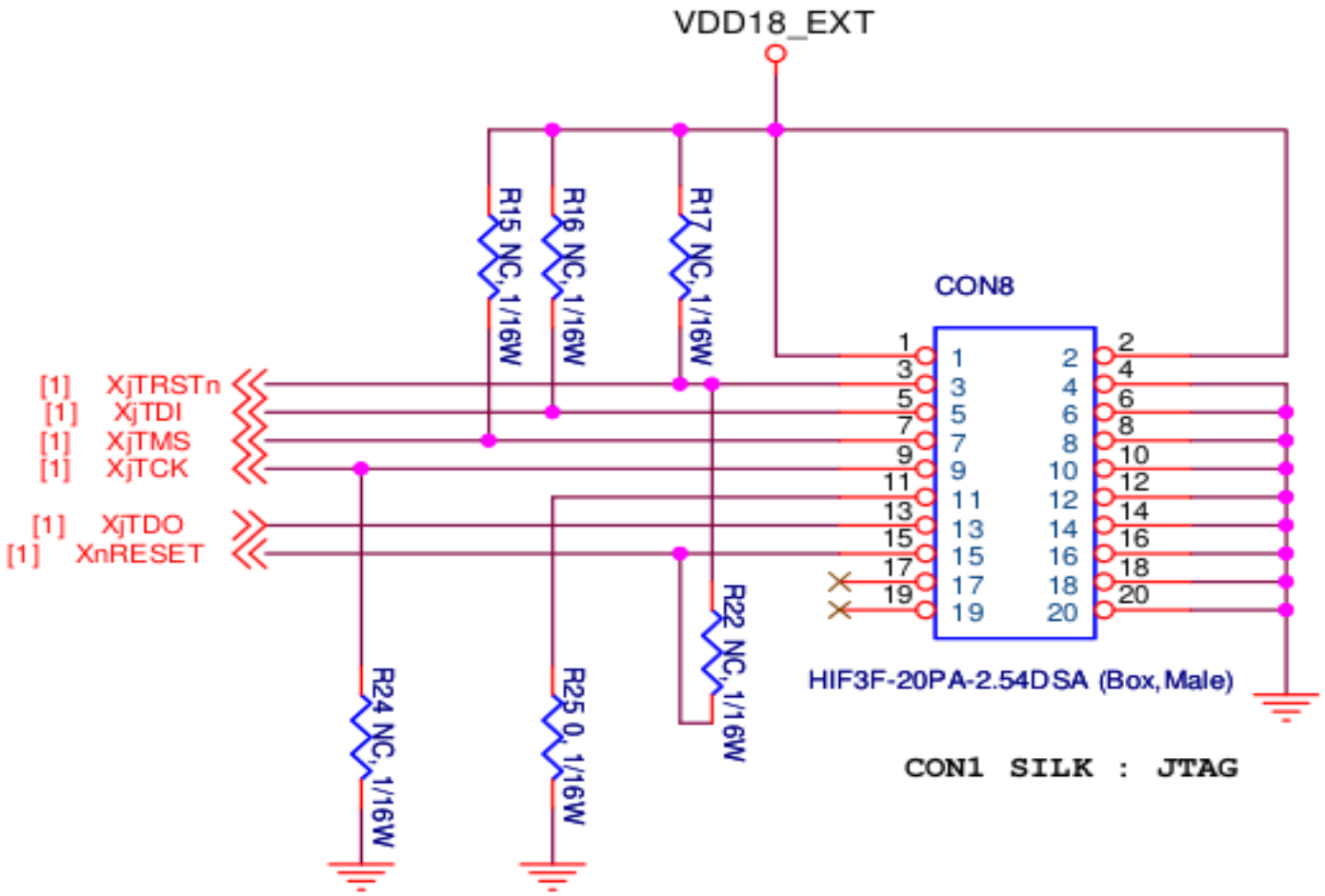
No JTAG connector, alternatives?



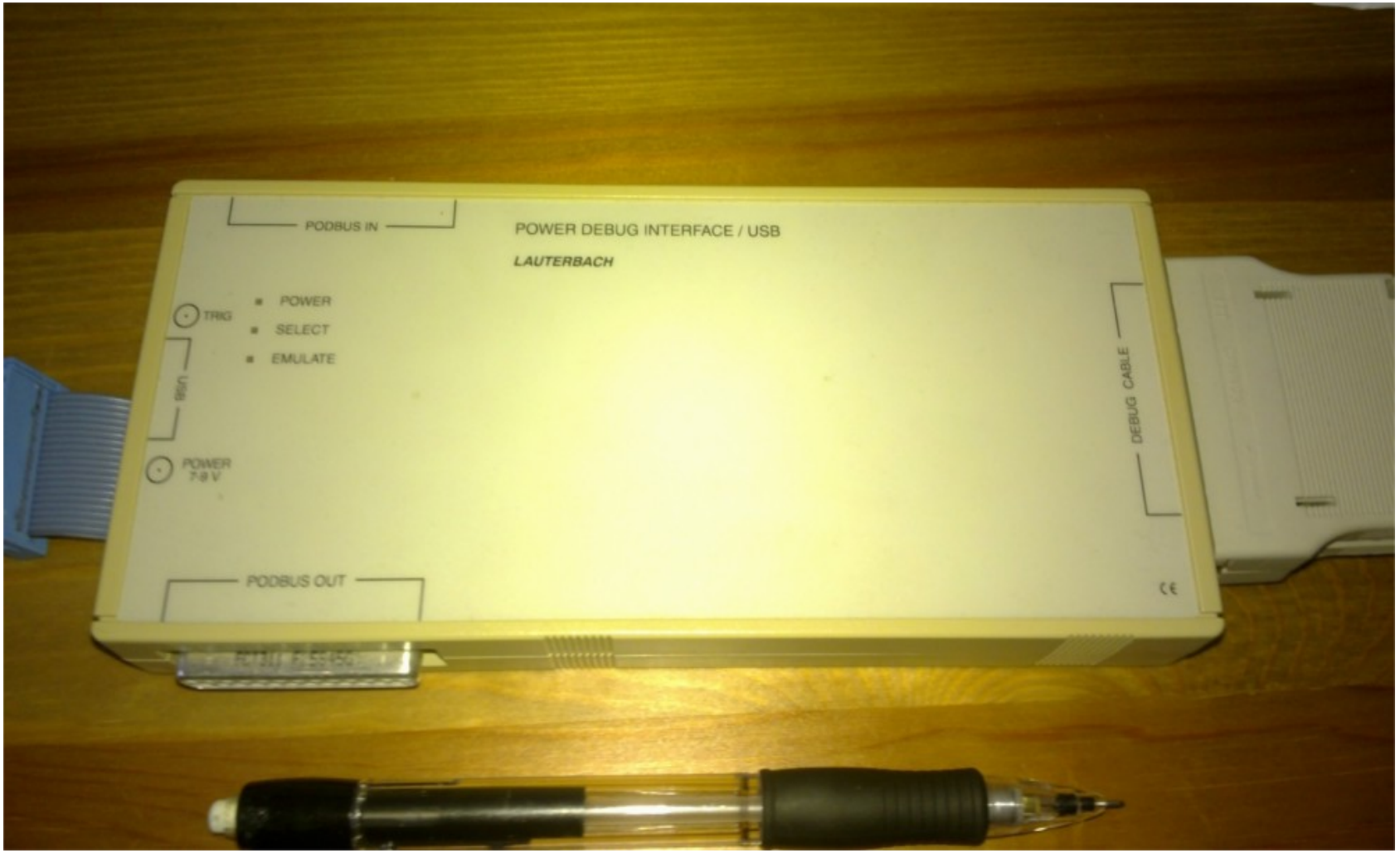
No JTAG connector, alternatives?



JTAG circuits

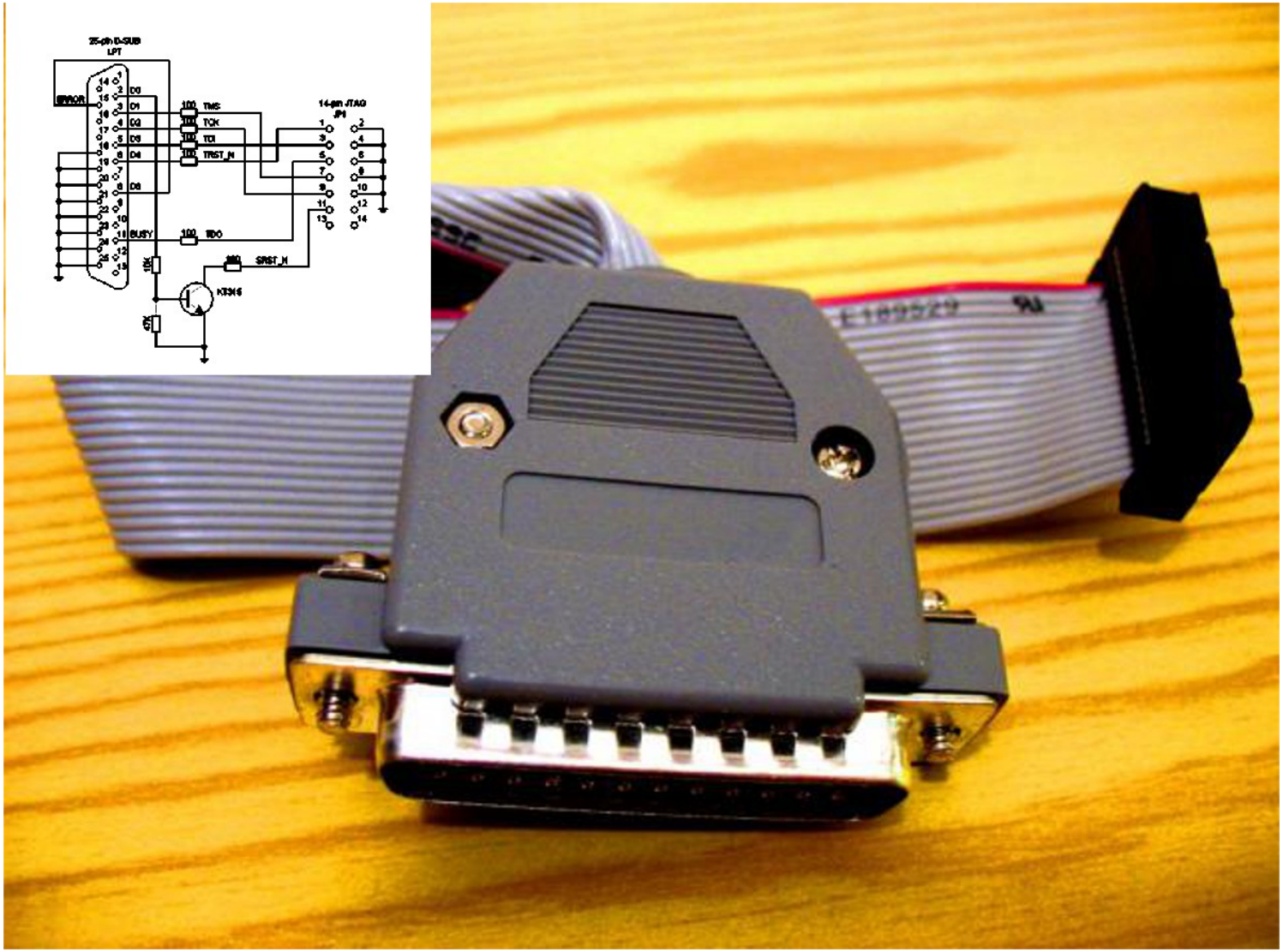


Lauterbach Trace32

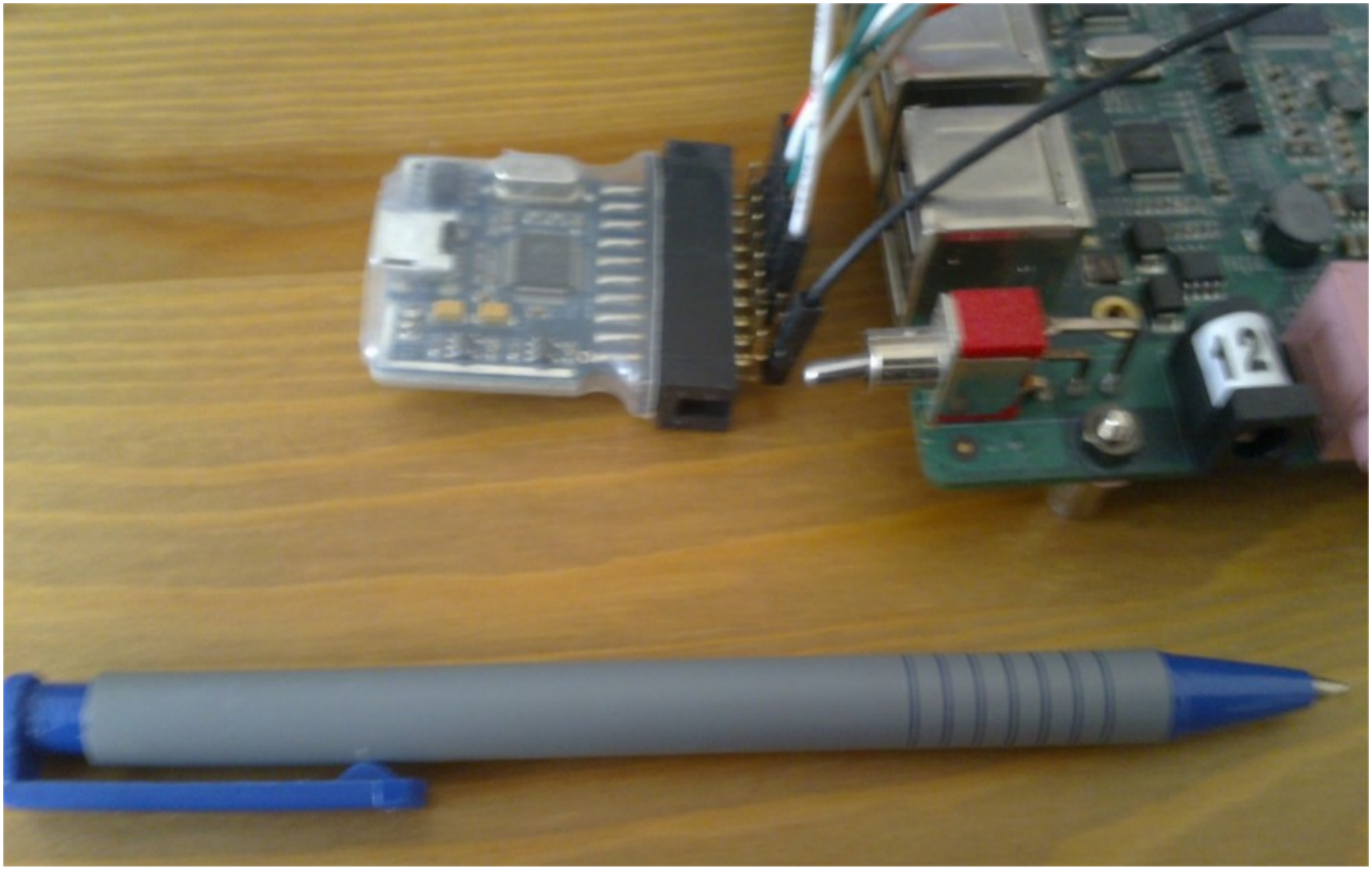


Abatron BDI2000/BDI3000

Parallel Port Interface



USB JTAG Dongle



My Favourite USB JTAG Dongle :)

Comparison


Price
Usability
Speed
ParPort
SoftICE
Lauterbach
BDI3000
USB Dongle

OpenOCD

Host side software, which provides debugging, in-system programming and boundary-scan testing for embedded devices.

Started as a master diploma, distributed under GPLv2 licence.

Supported interfaces:

- parallel port
- FTDI FT2232
- GPIO over sysfs
- many other less popular interfaces

Supported cores by OpenOCD

- ARM7 (ARM7TDMI, ARM720t)
- ARM9 (ARM9TDMI, ARM920t, ARM922t, ARM926ejs, ARM966)
- ARM11 (ARM1136, ARM1156, ARM1176)
- Intel Xscale (PXA25x, PXA26x, PXA27x, IXP42x, IXP45x, IXP46x)
- ARM Cortex-A8
- ARM Cortex-A9
- ARM Cortex-M3
- MIPS m4k

OpenOCD configuration

```
adapter_khz 1000
adapter_nsrst_delay 400
reset_config none

if { [info exists CHIPNAME] } {
  set _CHIPNAME $CHIPNAME
} else {
  set _CHIPNAME rspi
}

if { [info exists CPU_TAPID] } {
  set _CPU_TAPID $CPU_TAPID
} else {
  set _CPU_TAPID 0x07b7617f
}

jtag newtap $_CHIPNAME arm -irlen 5 -expected-id $_CPU_TAPID

set _TARGETNAME $_CHIPNAME.arm
target create $_TARGETNAME arm11 -chain-position $_TARGETNAME
rspi.arm configure -event gdb-attach { halt }
```

OpenOCD runtime

```
animist@meadow:~$ openocd -f /opt/backups/old/imx6.cfg
Open On-Chip Debugger 0.7.0 (2013-08-04-10:13)
Licensed under GNU GPL v2
For bug reports, read
        http://openocd.sourceforge.net/doc/doxygen/bugs.html
Info : only one transport option; autoselect 'jtag'
RCLK - adaptive
Warn : imx6q.sdma: nonstandard IR value
RCLK - adaptive
trst_and_srst separate srst_gates_jtag trst_push_pull srst_open_drain connect_deassert_srst
force hard breakpoints
Info : RCLK (adaptive clock speed) not supported - fallback to 2000 kHz
Polling target imx6q.cpu.0 failed, GDB will be halted. Polling again in 100ms
Polling target imx6q.cpu.1 failed, GDB will be halted. Polling again in 100ms
Polling target imx6q.cpu.0 failed, GDB will be halted. Polling again in 300ms
Info : JTAG tap: imx6q.dap tap/device found: 0x4ba00477 (mfg: 0x23b, part: 0xba00, ver: 0x4)
Info : TAP imx6q.sdma does not have IDCODE
Info : JTAG tap: imx6q.sjc tap/device found: 0x0191c01d (mfg: 0x00e, part: 0x191c, ver: 0x0)
Info : imx6q.cpu.0: hardware has 6 breakpoints, 4 watchpoints
Info : imx6q.cpu.1: hardware has 6 breakpoints, 4 watchpoints
Info : imx6q.cpu.2: hardware has 6 breakpoints, 4 watchpoints
Info : imx6q.cpu.3: hardware has 6 breakpoints, 4 watchpoints
Polling target imx6q.cpu.1 succeeded again
Polling target imx6q.cpu.0 succeeded again
```

```
animist@meadow:~$ telnet localhost 4444
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Open On-Chip Debugger
> scan_chain
   TapName             Enabled  IdCode     Expected   IrLen IrCap IrMask
-- ------------------- -------- ---------- ---------- ----- ----- ------
 0 imx6q.dap               Y     0x4ba00477 0x4ba00477     4 0x01  0x0f
 1 imx6q.sdma              Y     0x00000000 0x00000000     4 0x00  0x0f
 2 imx6q.sjc               Y     0x0191c01d 0x*191c01d     5 0x01  0x1f
> 
```

OpenOCD GDB session

```
animist@meadow:~$ PATH=${PATH}:/opt/projects/mentor/codebench/sourceryg-2013.11-32-arm/bin arm-none-linux-gn
ueabi-gdb
GNU gdb (Sourcery CodeBench 2013.11-32) 7.6.50.20130726-cvs
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "--host=i686-pc-linux-gnu --target=arm-none-linux-gnueabi".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://sourcery.mentor.com/GNUToolchain/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb) dir ~/linux.source
Source directories searched: /home/animist/linux.source:$cdir:$cwd
(gdb) file ~/linux.build/vmlinux
Reading symbols from  ~/linux.build/vmlinux...done.
(gdb) target remote localhost:3333
Remote debugging using localhost:3333
cpu_v7_do_idle () at ~/linux.source/arch/arm/mm/proc-v7.S:74
74              mov     pc, lr
(gdb) b do_sys_open
Breakpoint 1 at 0x800f3ee4: file ~/linux.source/fs/open.c, line 964.
(gdb) c
Continuing.

Breakpoint 1, do sys open (dfd=-100, filename=0x4a00e7bc "", flags=524288, mode=1)
    at ~/linux.source/fs/open.c:964
964     {
(gdb) n
966             int fd = build_open_flags(flags, mode, &op);
```

Problems

- only ARM SoCs are supported well, what about other RISCs?
- not all target ARM SoCs supported
- board support requires additional extensions to a target script
- ARM JTAG protocol implementation is not 100% sustainable
- … no more obvious problems from user's point of view

Thank You!