OpenOCD cheat sheet

General usage: **`openocd -f path/to/file-with-cmds.cfg -c "command in args"`**
default file path is **`/usr/share/openocd/scripts/`**
OpenOCD starts a GDB server by default on port 3333, can be configured with **`gdb_port`**
It also starts a Telnet server for running commands at runtime, on port 4444 (**`telnet_port`**)

Select interface ('JTAG probe'): **`-f interface/`*`probe`*`.cfg`** eg. **`buspirate`**
Select CPU/MCU: **`-f target/`*`mpu`*`.cfg`** eg. **`zynq_7000`**
Select devboard (interface+target): **`-f board/`*`boardname`*`.cfg`** eg. **`stm32f0discovery`**

General commands:

`help` ***`cmd`***	get help for a command
`exit`	close telnet, OpenOCD stays
`shutdown`	stop OpenOCD

Adapter/interface config:

`adapter driver` ***`name`***	select itf
`transport select` ***`xport`***	jtag/swd/...
`adapter speed` ***`khz`***	select phys. speed

Debugging:

`init`	config → runtime mode (cfg: specify target/itf/..., rt: do device ops)
`scan_chain`	do a scan of all available JTAG devices
`reg` ***`(name) (value)`***	get/set a register (default: get all)
`halt`	stop target cpu for debugging
`resume|step` ***`(addr)`***	resume/single-step cpu, opt. at address
`reset (run|halt|init)`	hard reset
`m[dw][bhwd]` ***`addr (val) (num)`***	memory **D**isplay/**W**rite, 8(**b**)/16(**h**)/32(**w**)/64(**d**)-bit
`dump_image` ***`file addr size`***	dump memory to file
`load_image` ***`file (addr) (size)`***	load file into memory

Flash commands:

`flash banks`	list available flash on target
`flash info` ***`bankno`***	show info (size, lock, ...) of bank
`flash read_bank` ***`bankno file (addr) (len)`***	dump flash to file
`program` ***`file`*** **`(actions|`**`addr`**`)`**	program flash

Note: specific flash drivers can have extra lock/protect/... options, see docs. Not covered: erasing flash (**`flash erase_sector`** etc).

NAND commands:

`nand list`, **`nand probe`** ***`num`***	like **`flash banks`**, **`flash info`**. **`probe`** is reqd for ↓!
`nand dump` ***`num file off len`***	dump NAND
`nand write` ***`num file off`***	write NAND

Not covered: erasing NAND (**`nand erase`**), checking integrity (**`check_bad_blocks`** etc).

TL;DR and examples:

"*I want to load a program into RAM and debug it with gdb*"

`openocd -f board/`*`mydevbrd`*`.cfg -c init -c "load_image` ***`myprog`***`.elf" -c "reset halt"`**

→ connect gdb to server at **`localhost:3333`**. sequence of **`-c`** args can be put in its own script file and used with **`-f`** to avoid having to retype everything

"*Tell me what this device is*":

`openocd -f interface/`*`probe`*`.cfg -c "transport select jtag" -c init -c scan_chain -c shutdown`

"*I want to dump the flash of this MCU I found*"

`openocd -f interface/`*`probe`*`.cfg -f target/`*`mcu`*`.cfg -c init -c "flash banks" -c "flash info 0" -c shutdown`

`openocd -f interface/`*`probe`*`.cfg -f target/`*`mcu`*`.cfg -c init -c "flash read_bank 0` ***`flash.bin`***`" -c shutdown`**