**J-LINK 命令详解**

J-Link>?

Available commands are:

----------------------

f Firmware info

h halt

g go

Sleep Waits the given time (in milliseconds). Syntax: Sleep <delay>

s Single step the target chip

st Show hardware status

hwinfo Show hardware info

mem Read memory. Syntax: mem <Addr>, <NumBytes> (hex)

mem8 Read 8-bit items. Syntax: mem8 <Addr>, <NumBytes> (hex)

mem16 Read 16-bit items. Syntax: mem16 <Addr>, <NumItems> (hex)

mem32 Read 32-bit items. Syntax: mem32 <Addr>, <NumItems> (hex)

w1 Write 8-bit items. Syntax: w1 <Addr>, <Data> (hex)

w2 Write 16-bit items. Syntax: w2 <Addr>, <Data> (hex)

w4 Write 32-bit items. Syntax: w4 <Addr>, <Data> (hex)

wm Write test words. Syntax: wm <NumWords>

is Identify length of scan chain select register

ms Measure length of scan chain. Syntax: ms <Scan chain>

mr Measure RTCK react time. Syntax: mr

q Quit

qc Close JLink connection and quit

r Reset target (RESET)

rx Reset target (RESET). Syntax: rx <DelayAfterReset>

RSetType Set the current reset type. Syntax: RSetType <type>

Regs Display contents of registers

wreg Write register. Syntax: wreg <RegName>, <Value>

SetBP Set breakpoint. Syntax: SetBP <addr> [A/T] [S/H]

SetWP Set Watchpoint. Syntax: <Addr> [R/W] [<Data> [<D-Mask>] [A-Mask]]

ClrBP Clear breakpoint. Syntax: ClrBP <BP\_Handle>

ClrWP Clear watchpoint. Syntax: ClrWP <WP\_Handle>

VCatch Write vector catch. Syntax: VCatch <Value>

loadbin Load binary file into target memory.

Syntax: loadbin <filename>, <addr>

savebin Saves target memory into binary file.

Syntax: savebin <filename>, <addr>, <NumBytes>

SetPC Set the PC to specified value. Syntax: SetPC <Addr>

le Change to little endian mode

be Change to big endian mode

log Enables log to file. Syntax: log <filename>

unlock Unlocks a device. Syntax: unlock <DeviceName>

Type unlock without <DeviceName> to get a list

of supported device names.

nRESET has to be connected

---- CP15 ------------

rce Read CP15. Syntax: rce <Op1>, <CRn>, <CRm>, <Op2>

wce Write CP15. Syntax: wce <Op1>, <CRn>, <CRm>, <Op2>, <Data>

---- ICE -------------

Ice Show state of the embedded ice macrocell (ICE breaker)

ri Read Ice reg. Syntax: ri <RegIndex>(hex)

wi Write Ice reg. Syntax: wi <RegIndex>, <Data>(hex)

---- ETM -------------

etm Show ETM status

re Read ETM reg. Syntax: re <RegIndex>

we Write ETM reg. Syntax: we <RegIndex>, <Data>(hex)

es Start trace

---- ETB -------------

etb Show ETB status

rb Read ETB register. Syntax: rb <RegIndex>

wb Write ETB register. Syntax: wb <RegIndex>, <Data>(hex)

---- TRACE -----------

TAddBranch TRACE - Add branch instruction to trace buffer. Paras:<Addr>,<BAddr>

TAddInst TRACE - Add (non-branch) instruction to trace buffer. Syntax: <Addr>

TClear TRACE - Clear buffer

TSetSize TRACE - Set Size of trace buffer

TSetFormat TRACE - SetFormat

TSR TRACE - Show Regions (and analyze trace buffer)

TStart TRACE - Start

TStop TRACE - Stop

---- SWO -------------

SWOSpeed SWO - Show supported speeds

SWOStart SWO - Start

SWOStop SWO - Stop

SWOStat SWO - Display SWO status

SWORead SWO - Read and display SWO data

SWOShow SWO - Read and analyze SWO data

SWOFlush SWO - Flush data

---- File I/O --------

fwrite Write file to emulator

fread Read file from emulator

fshow Read and display file from emulator

fdelete Delete file on emulator

fsize Display size of file on emulator

---- Test ------------

TestHaltGo Run go/halt 1000 times

TestStep Run step 1000 times

TestCSpeed Measure CPU speed.

Parameters: [<RAMAddr>]

TestWSpeed Measure download speed into target memory.

Parameters: [<Addr> [<Size>]]

TestRSpeed Measure upload speed from target memory.

Parameters: [<Addr> [<Size>] [<NumBlocks>]]

TestNWSpeed Measure network download speed.

Parameters: [<NumBytes> [<NumReps>]]

TestNRSpeed Measure network upload speed.

Parameters: [<NumBytes> [<NumReps>]]

---- JTAG ------------

Config Set number of IR/DR bits before ARM device.

Syntax: Config <IRpre>, <DRpre>

speed Set JTAG speed. Syntax: speed <freq>|auto|adaptive, e.g. speed 2000,

speed a

i Read JTAG Id (Host CPU)

wjc Write JTAG command (IR). Syntax: wjc <Data>(hex)

wjd Write JTAG data (DR). Syntax: wjd <Data32>(hex), <NumBits>(dec)

RTAP Reset TAP Controller using state machine (111110)

wjraw Write Raw JTAG data. Syntax: wjraw <NumBits(dec)>, <tms>, <tdi>

rt Reset TAP Controller (nTRST)

---- JTAG-Hardware ---

c00 Create clock with TDI = TMS = 0

c Clock

tck0 Clear TCK

tck1 Set TCK

0 Clear TDI

1 Set TDI

t0 Clear TMS

t1 Set TMS

trst0 Clear TRST

trst1 Set TRST

r0 Clear RESET

r1 Set RESET

---- Connection ------

usb Connect to J-Link via USB. Syntax: usb <port>, where port is 0..3

ip Connect to J-Link ARM Pro or J-Link TCP/IP Server via TCP/IP.

Syntax: ip <ip\_addr>

---- Configuration ---

si Select target interface. Syntax: si <Interface>,

where 0=JTAG and 1=SWD.

power Switch power supply for target. Syntax: power <State> [perm],

where State is either On or Off. Example: power on perm

wconf Write configuration byte. Syntax: wconf <offset>, <data>

rconf Read configuration bytes. Syntax: rconf

usbaddr Assign usb address to the connected J-Link: Syntax: usbaddr = <addr>

ipaddr Show/Assign IP address and subnetmask of/to the connected J-Link.

gwaddr Show/Assign network gateway address of/to the connected J-Link.

dnsaddr Show/Assign network DNS server address of/to the connected J-Link.

conf Show configuration of the connected J-Link.

ecp Enable the J-Link control panel.

calibrate Calibrate the target current measurement.

selemu Select a emulator to communicate with,

from a list of all emulators which are connected to the host

The interfaces to search on, can be specified

Syntax: selemu [<Interface0> <Interface1> ...]

ShowEmuList Shows a list of all emulators which are connected to the host.

The interfaces to search on, can be specified.

Syntax: ShowEmuList [<Interface0> <Interface1> ...]

----------------------

NOTE: Specifying a filename in command line

will start J-Link Commander in script mode.