

Teletype Documentation

Contents

1	Modes	2
	Global key bindings	2
	Text editing	2
	Live mode	3
	Edit mode	3
	Pattern mode	3
	Preset read mode	4
	Preset write mode	4
	Help mode	5
2	OPs and MODs	6
	Variables	6
	Maths	6
3	Extended OP documentation	7
	ADD	7
	DRUNK	7
A	Missing documentation	8
B	Changelog	9
	v2.0	9
	v1.4.1	10
	v1.2.1	10
	v1.2	10
	v1.1	10
	v1.0	11

Chapter 1

Modes

Global key bindings

These bindings work everywhere.

Key	Action
<tab>	change modes, live to edit to pattern and back
<esc>	preset read mode, or return to last mode
alt-<esc>	preset write mode
win-<esc>	clear delays, stack and slews
<print screen>	help text, or return to last mode
<F1> to <F8>	run corresponding script
<F9>	run metro script
<F10>	run init script
alt-<F1> to alt-<F8>	edit corresponding script
alt-<F9>	edit metro script
alt-<F10>	edit init script
<numpad-1> to <numpad-8>	run corresponding script

Text editing

These bindings work when entering text or code.

In most cases, the clipboard is shared between *live*, *edit* and the 2 *preset* modes.

Key	Action
<left> / ctrl-b	move cursor left
<right> / ctrl-f	move cursor right

Key	Action
<home> / ctrl-a	move to beginning of line
<end> / ctrl-e	move to end of line
<backspace> / ctrl-h	backwards delete one character
<delete> / ctrl-d	forwards delete one character
shift-<backspace> / ctrl-u	delete from cursor to beginning
shift-<delete> / ctrl-e	delete from cursor to end
alt-<backspace> / ctrl-w	delete from cursor to beginning of word
ctrl-x / alt-x	cut to clipboard
ctrl-c / alt-c	copy to clipboard
ctrl-v / alt-v	paste to clipboard

Live mode

Key	Action
<down> / C-n	history next
<up> / C-p	history previous
<enter>	execute command
[/]	switch to edit mode

Edit mode

Key	Action
<down> / C-n	line down
<up> / C-p	line up
[previous script
]	next script
<enter>	enter command
shift-<enter>	insert command

Pattern mode

The pattern mode clipboard is independent of text and code clipboard.

Key	Action
<down>	move down
alt-<down>	move a page down

Key	Action
<up>	move up
alt-<up>	move a page up
<left>	move left
alt-<left>	move to the very left
<right>	move right
alt-<right>	move to the very right
[decrement by 1
]	increment by 1
<backspace>	delete a digit
shift-<backspace>	delete an entry, shift numbers up
<enter>	move down (increase length only if on the entry immediately after the current length)
shift-<enter>	duplicate entry and shift downwards (increase length as <enter>)
alt-x	cut value (n.b. ctrl-x not supported)
alt-c	copy value (n.b. ctrl-c not supported)
alt-v	paste value (n.b. ctrl-v not supported)
shift-alt-v	insert value
shift-l	set length to current position
alt-l	go to current length entry
shift-s	set start to current position
alt-s	go to start entry
shift-e	set end to current position
alt-e	go to end entry
-	negate value
<space>	toggle non-zero to zero, and zero to 1
0 to 9	numeric entry

Preset read mode

Key	Action
<down> / C-n	line down
<up> / C-p	line up
<left> / [preset down
<right> /]	preset up
<enter>	load preset

Preset write mode

Key	Action
<down> / C-n	line down
<up> / C-p	line up
[preset down
]	preset up
<enter>	enter text
alt-<enter>	save preset

Help mode

Key	Action
<down> / C-n	line down
<up> / C-p	line up
<left> / [previous page
<right> /]	next page

Chapter 2

OPs and MODs

Variables

OP	OP (<i>set</i>)	(<i>aliases</i>)	Description
A	A x		get / set the variable A, default value 1
B	B x		get / set the variable B, default value 2
C	C x		get / set the variable C, default value 3
D	D x		get / set the variable D, default value 4
DRUNK	DRUNK x		changes by -1, 0, or 1 upon each read saving its state, setting will give it a new value for the next read
DRUNK.MIN	DRUNK.MIN x		set the lower bound for DRUNK
FLIP	FLIP x		returns inverted state (0 or 1) on each read (also settable)

Maths

OP	OP (<i>set</i>)	(<i>aliases</i>)	Description
ADD x y		+	add x and y together

Chapter 3

Extended OP documentation

ADD

- **ADD** *x y*
- *alias*: +

Add 2 numbers together

Example 1:

ADD 2 3
= 5

Example 2:

+ 2 5
= 7

DRUNK

- **DRUNK** / **DRUNK** *x*

Changes by -1, 0, or 1 upon each read, saving its state. Setting DRUNK will give it a new value for the next read, and drunkenness will continue on from there with subsequent reads.

Appendix A

Missing documentation

!, !=, %, &&, *, -, /, <, <<, <=, ==, >, >=, >>, ABS, AND, ARP.DIV, ARP.ER, ARP.FILL, ARP.GATE, ARP.HOLD, ARP.RESET, ARP.ROT, ARP.RPT, ARP.SHIFT, ARP.SLEW, ARP.STYLE, AVG, CV, CV.OFF, CV.SET, CV.SLEW, CY.CV, CY.POS, CY.PRESET, CY.RESET, CY.REV, DEL, DEL.CLR, DIV, DRUNK.MAX, DRUNK.WRAP, ELIF, ELSE, EQ, ER, ES.CLOCK, ES.MAGIC, ES.MODE, ES.PATTERN, ES.PRESET, ES.RESET, ES.STOP, ES.TRANS, ES.TRIPLE, EXP, EZ, GT, I, IF, IN, JF.GOD, JF.MODE, JF.NOTE, JF.QT, JF.RMODE, JF.RUN, JF.SHIFT, JF.TICK, JF.TR, JF.TUNE, JF.VOX, JF.VTR, JI, KILL, KR.LOOP.LEN, KR.LOOP.ST, KR.PATTERN, KR.PERIOD, KR.POS, KR.PRESET, KR.RESET, KR.SCALE, L, LIM, LSH, LT, LV.CV, LV.L.DIR, LV.L.LEN, LV.L.ST, LV.POS, LV.PRESET, LV.RESET, M, M.ACT, M.RESET, MAX, MID.SHIFT, MID.SLEW, MIN, MOD, MP.FREEZE, MP.MUTE, MP.PERIOD, MP.PRESET, MP.PRESET1, MP.RESET, MP.RESET1, MP.SCALE, MP.STOP, MP.STOP1, MP.SYNC, MP.UNFREEZE, MP.UNMUTE, MUL, MUTE, N, NE, NZ, O, O.INC, O.MAX, O.MIN, O.WRAP, OR, OR.BANK, OR.CLK, OR.CVA, OR.CVB, OR.DIV, OR.GRST, OR.MUTE, OR.PHASE, OR.PRESET, OR.RELOAD, OR.ROTS, OR.ROTW, OR.RST, OR.SCALE, OR.TRK, OR.WGT, P, P.END, P.HERE, P.I, P.INS, P.L, P.N, P.NEXT, P.POP, P.PREV, P.PUSH, P.RM, P.START, P.WRAP, PARAM, PN, PN.END, PN.HERE, PN.I, PN.INS, PN.L, PN.NEXT, PN.POP, PN.PREV, PN.PUSH, PN.RM, PN.START, PN.WRAP, PRM, PROB, Q, Q.AVG, Q.N, QT, RAND, RRAND, RSH, S, S.ALL, S.CLR, S.L, S.POP, SCALE, SCENE, SCRIPT, STATE, SUB, T, TI.IN, TI.IN.CALIB, TI.IN.MAP, TI.IN.N, TI.IN.QT, TI.IN.SCALE, TI.PARAM, TI.PARAM.CALIB, TI.PARAM.MAP, TI.PARAM.N, TI.PARAM.QT, TI.PARAM.SCALE, TI.RESET, TI.STORE, TIME, TIME.ACT, TO.CV, TO.CV.N, TO.CV.N.SET, TO.CV.OFF, TO.CV.QT, TO.CV.QT.SET, TO.CV.SCALE, TO.CV.SET, TO.CV.SLEW, TO.CV.SLEW.M, TO.CV.SLEW.S, TO.ENV.ACT, TO.ENV.ATT, TO.ENV.ATT.M, TO.ENV.ATT.S, TO.ENV.DEC, TO.ENV.DEC.M, TO.ENV.DEC.S, TO.ENV.TRIG, TO.KILL, TO.OSC, TO.OSC.CYC, TO.OSC.CYC.M, TO.OSC.CYC.M.SET, TO.OSC.CYC.S, TO.OSC.CYC.S.SET, TO.OSC.CYC.SET, TO.OSC.FQ, TO.OSC.FQ.SET, TO.OSC.LF0, TO.OSC.LF0.SET, TO.OSC.N, TO.OSC.N.SET, TO.OSC.PHASE, TO.OSC.QT, TO.OSC.QT.SET, TO.OSC.RECT, TO.OSC.SCALE, TO.OSC.SET, TO.OSC.SLEW, TO.OSC.SLEW.M, TO.OSC.SLEW.S, TO.OSC.SYNC, TO.OSC.WAVE, TO.OSC.WIDTH, TO.TR, TO.TR.M, TO.TR.M.ACT, TO.TR.M.BPM, TO.TR.M.COUNT, TO.TR.M.M, TO.TR.M.S, TO.TR.M.SYNC, TO.TR.POL, TO.TR.PULSE, TO.TR.PULSE.DIV, TO.TR.TIME, TO.TR.TIME.M, TO.TR.TIME.S, TO.TR.TOG, TO.TR.WIDTH, TOSS, TR, TR.P, TR.POL, TR.PULSE, TR.TIME, TR.TOG, V, VV, WRAP, WW.END, WW.MUTE1, WW.MUTE2, WW.MUTE3, WW.MUTE4, WW.MUTEA, WW.MUTEB, WW.PATTERN, WW.PMODE, WW.POS, WW.PRESET, WW.QPATTERN, WW.START, WW.SYNC, X, XOR, Y, Z, ||

Appendix B

Changelog

v2.0

- **BREAKING:** remove II op. Ops that required it will now work with out it. (e.g. II MP.PRESET 1 will become just MP.PRESET 1)
- **BREAKING:** merge the MUTE and UNMUTE ops. Now MUTE x will return the mute status for trigger x (0 is unmuted, 1 is muted), and MUTE x y will set the mute for trigger x (y = 0 to unmute, y = 1 to mute)
- **NEW:** sub commands, use a ; separator to run multiple commands on a single line, e.g. X 1; Y 2
- **NEW:** key bindings rewritten, see docs
- **NEW:** multiple commands on each line separated by ‘:’
- **NEW:** aliases: + for ADD, – for SUB, * for MUL, / for DIV, % for MOD, << for LSH, >> for RSH, == for EQ, != for NE, < for LT, > for GT, <= for LTE, >= for GTE, ! for EZ, && for AND, || for OR, PRM for PARAM, TR.P for TR.PULSE
- **NEW:** new ops: LTE (less than or equal), and GTE (greater than or equal)
- **NEW:** new pattern ops: PN.L, PN.WRAP, PN.START, PN.END, PN.I, PN.HERE, PN.NEXT, PN.PREV, PN.INS, PN.RM, PN.PUSH and PN.POP
- **NEW:** USB disk loading and saving works at any time
- **IMP:** new Ragel parser backend
- **IMP:** script recursion enhanced, maximum recursion depth is 8, and self recursion is allowed
- **IMP:** removed the need to prefix : and ; with a space, e.g. IF X : TR.PULSE 1 becomes IF X: TR.PULSE
- **FIX:** divide by zero errors now explicitly return a 0 (e.g. DIV 5 0 now returns 0 instead of -1), previously the behaviour was undefined and would crash the simulator
- **FIX:** numerous crashing bugs with text entry
- **FIX:** i2c bus crashes under high M times with external triggers

v1.4.1

- **NEW:** added Ansible remote commands LV.CV and CY.CV
- **NEW:** Added TELEX Modules Support for the TXi and the TXo
- **NEW:** 75 New Operators Across the Two Modules
- **NEW:** Supports all basic Teletype functions (add TI and T0 to the commands you already know)
- **NEW:** Extended functionality allows for additional capabilities for existing functions
- **NEW:** Experimental input operators add capabilities such as input range mapping and quantization
- **NEW:** Experimental output operators add oscillators, envelopes, independent metronomes, pulse dividing, etc.
- **NEW:** Full List of Methods Found and Maintained Here

v1.2.1

- **NEW:** Just Friends ops: JF.GOD, JF.MODE, JF.NOTE, JF.RMODE, JF.RUN, JF.SHIFT, JF.TICK, JF.TR, JF.TUNE, JF.VOX, JF.VTR

v1.2

- **NEW:** Ansible support added to ops: CV, CV.OFF, CV.SET, CV.SLEW, STATE, TR, TR.POL, TR.PULSE, TR.TIME, TR.TOG
- **NEW:** P.RM will also return the value removed
- **NEW:** ER op
- **IMP:** a TR.TIME of 0 will disable the pulse
- **IMP:** 0.DIR renamed to 0.INC, it's the value by which 0 is *incremented* when it is accessed
- **IMP:** IF, ELIF, ELSE status is reset on each script run
- **IMP:** key repeat now works for all keypresses
- **FIX:** FLIP won't interfere with the value of 0
- **FIX:** the 0 op now returns it's set value *before* updating itself
- **FIX:** the DRUNK op now returns it's set value *before* updating itself
- **FIX:** P.START and P.END were set to 1 when set with too large values, now are set to 63
- **FIX:** CV.SLEW is correctly initialised to 1 for all outputs
- **FIX:** several bugs where pattern length wasn't updated in track mode
- **FIX:** fixed [and] not updating values in track mode

v1.1

- **NEW:** USB flash drive read/write
- **NEW:** SCRIPT op for scripted execution of other scripts!
- **NEW:** MUTE and UNMUTE ops for disabling trigger input

- **NEW:** hotkeys for MUTE toggle per input (meta-shift-number)
- **NEW:** screen indication in live mode for MUTE status
- **NEW:** SCALE op for scaling number from one range to another
- **NEW:** JI op just intonation helper
- **NEW:** STATE op to read current state of input triggers 1-8 (low/high = 0/1)
- **NEW:** keypad executes scripts (works for standalone USB keypads and full-sized keyboards)
- **NEW:** KILL op clears delays, stack, CV slews, pulses
- **NEW:** hotkey meta+ESC executes KILL
- **NEW:** ABS op absolute value, single argument
- **NEW:** FLIP op variable which changes state (0/1) on each read
- **NEW:** logic ops: AND, OR, XOR
- **NEW:** O ops: O.MIN, O.MAX, O.WRAP, O.DIR for counter range control
- **NEW:** DRUNK ops: DRUNK.MIN, DRUNK.MAX, DRUNK.WRAP for range control
- **NEW:** TR.POL specifies the polarity of TR.PULSE
- **NEW:** if powered down in tracker mode, will power up in tracker mode
- **IMP:** TR.PULSE retrigger behaviour now predictable
- **IMP:** mode switch keys more consistent (not constantly resetting to live mode)
- **FIX:** bug in command history in live mode
- **FIX:** EXP op now exists
- **FIX:** P and PN parse error
- **FIX:** possible crash on excess length line entry
- **FIX:** CV wrapping with negative CV.OFF values
- **FIX:** INIT script executed now on keyboardless scene recall
- **FIX:** Q.AVG overflow no more
- **FIX:** P.PUSH will fully fill a pattern
- **FIX:** CV.SET followed by slewed CV in one command works
- **FIX:** DEL 0 no longer voids command

v1.0

- Initial release