

# duckyScript Cheatsheet: Key Combos

11 Jan 2026

duckyPad Quick Ref 1/6

## Latest Resources

- [duckyPad Homepage](#)
- [duckyScript Full Doc](#)
- [Autoswitcher](#)
- [Syntax Highlighter](#)
- [Discord Chatroom](#)

## Available Special Keys

CTRL / RCTRL	(media keys)
SHIFT / RSHIFT	MK_VOLUP
ALT / RALT	MK_VOLDOWN
WINDOWS / RWINDOWS	MK_MUTE
COMMAND / RCOMMAND	MK_PREV
OPTION / ROPTION	MK_NEXT
ESC	MK_PP (play/pause)
ENTER	MK_STOP
UP/DOWN/LEFT/RIGHT	
SPACE	(numpad keys)
BACKSPACE	NUMLOCK
TAB	KP_SLASH
CAPSLOCK	KP_ASTERISK
PRINTSCREEN	KP_MINUS
SCROLLLOCK	KP_PLUS
PAUSE	KP_ENTER
BREAK	KP_0 - KP_9
INSERT	KP_DOT
HOME	KP_EQUAL
PAGEUP / PAGEDOWN	
DELETE	(Japanese IME)
END	ZENKAKUHANKAKU
MENU	HENKAN
POWER	MUHENKAN
F1 - F24	KATAKANAHIRAGANA

## Key Combos

- Easiest to Write
- For Shortcuts & Hotkeys
- Any Combo of:
- Special Keys / Letters / Numbers

ENTER

CTRL S

COMMAND SHIFT 4

Type Key Name in ALL CAPS

# duckyScript Commands

duckyPad Quick Ref 2/6

Typing	
STRING <code>text</code>	Type text AS-IS
STRINGLN <code>text</code>	Type text AS-IS Press ENTER at end
STRING_BLOCK END_STRING	Type text block AS-IS <b>No new lines</b>
STRINGLN_BLOCK END_STRINGLN	Same as above Press ENTER after <b>each line</b>

OLED	
OLED_CLEAR	Clear Screen
OLED_CURSOR <code>x</code> <code>y</code>	Set Cursor <code>x</code> <code>y</code> : 0 to 127 (0,0) = Top Left
OLED_PRINT <code>text</code>	Print Text at Current Cursor
OLED_CPRINT <code>text</code>	Print Text <b>Center-Aligned</b>
OLED_CIRCLE <code>x</code> <code>y</code> <code>radius</code> <code>fill</code>	<code>x</code> <code>y</code> : Position <code>radius</code> : In Pixels <code>fill</code> : 0 or 1
OLED_LINE <code>x1</code> <code>y1</code> <code>x2</code> <code>y2</code>	<code>x1</code> <code>y1</code> : Start <code>x2</code> <code>y2</code> : End
OLED_RECT <code>x1</code> <code>y1</code> <code>x2</code> <code>y2</code> <code>fill</code>	<code>x1</code> <code>y1</code> : Start <code>x2</code> <code>y2</code> : End <code>fill</code> : 0 or 1
OLED_UPDATE	<b>Commit Changes to Screen</b>
OLED_RESTORE	Show Default Screen

Pressing Keys	
Key Combos	See Page 1
KEYDOWN <code>key</code>	Hold key
KEYUP <code>key</code>	Release Key
<code>key</code> can be letter, number, or special key.	

Profile Switching	
PREV_PROFILE	NEXT_PROFILE
GOTO_PROFILE <code>name</code>	Case sensitive
Also check out <a href="#">duckyPad Autoswitcher</a>	

Miscellaneous	
DP_SLEEP	RGB & Screen OFF Halts Execution
HALT	Halt Execution
REPEAT <code>n</code>	Repeat <b>line above</b> <code>n</code> times

Comments	
//	
REM_BLOCK	ENDREM

Random Letters	
Type a random character	
RANDOM_LOWER_CASELETTER	
RANDOM_NUMBER	
RANDOM_UPPERCASELETTER	
RANDOM_SPECIAL	
RANDOM LETTER	
RANDOM CHAR	

Mouse	
LMOUSE MMOUSE RMOUSE FMOUSE BMOUSE	Left / Middle / Right Forward / Backward Mouse Button Click
MOUSE_MOVE <code>x</code> <code>y</code>	Move Mouse in Pixels <code>x</code> : + Right - Left <code>y</code> : + Up - Down
MOUSE_SCROLL <code>h</code> <code>v</code>	Scroll horizontal <code>h</code> lines vertical <code>v</code> lines

RGB Backlight	
SWC_SET <code>n</code> <code>r</code> <code>g</code> <code>b</code>	Change RGB Colour <code>n</code> : Key ID (0 for current key) <code>r</code> <code>g</code> <code>b</code> : 0 to 255
SWC_FILL <code>r</code> <code>g</code> <code>b</code>	Change ALL RGB <code>r</code> <code>g</code> <code>b</code> : 0 to 255
SWC_RESET <code>n</code>	Reset RGB to Default <code>n</code> : Key ID 0 = Current Key 99 = All keys

# duckyScript: Advanced Usage

duckyPad Quick Ref 3/6

## Constants

DEFINE	Define a Constant
	Replaced AS-IS during preprocessing
DEFINE EMAIL	bob@me.com

## Variables

VAR	Declare Signed 32-bit Variable
	VAR foo = 10

## Persistent Global Variables

\_GV0 to \_GV31

- Available across all profiles
- Persists over reboots

## Operators (Signed)

= Assign	&& Logical AND
== Equal	Logical OR
!= Not equal	! Logical NOT
> Greater than	& Bitwise AND
< Less than	Bitwise OR
>= GTE	^ Bitwise XOR
<= LTE	<< Left Shift
	>> Right Shift

Augmented Assignments: +=, \*=, etc

## Operators (Unsigned)

ULT(lhs, rhs)	UGTE(lhs, rhs)
ULTE(lhs, rhs)	UDIV(val, n)
UGT(lhs, rhs)	UMOD(val, n)
LSR(val, n)	

## IF Statement

Code inside is executed If expression is non-zero	
IF expression	code
END_IF	
Additional Checks	ELSE IF
	ELSE

## WHILE loop

Code inside is repeated If expression is non-zero	
WHILE expression	code
END WHILE	
Jump to start	CONTINUE
Exit immediately	LBREAK

## Functions

FUN my_func(args)	code=
	END_FUN
my_func() // call it	
Optional args & returns	

VARs declared inside have local scope

## Randomisation

VAR foo = RANDINT(lower, upper)
Unsigned: RANDUINT(lower, upper)
Range is inclusive

## Reading Keys

Blocking	VAR key = _BLOCKING_READKEY Wait until any keypress, returns KeyID
Non-Blocking	VAR key = _READKEY 0 if no key pressed, KeyID otherwise
Bitfield	VAR key = _SW_BITFIELD Each bit: 1 = Pressed, 0 = Released

## Real-time Clock

Automatically set when using <a href="#">Autoswitcher</a>	
Check _RTC_IS_VALID first.	
Do not proceed if 0.	
_RTC_YEAR	_RTC_MINUTE
_RTC_MONTH	_RTC_SECOND
_RTC_DAY	_RTC_WDAY
_RTC_HOUR	_RTC_YDAY

## VAR Print & Formatting

VAR foo = 10  
STRING Value is \$foo

Format Specifiers:  
Add **immediately**  
**after** var name

%d: Signed Decimal  
%u: Unsigned Decimal  
%x: Hex Lower  
%X: Hex Upper

Zero-Pad	%02d, %04x, etc
Space-Pad	%2d, %4x, etc

# duckyScript: Advanced Usage

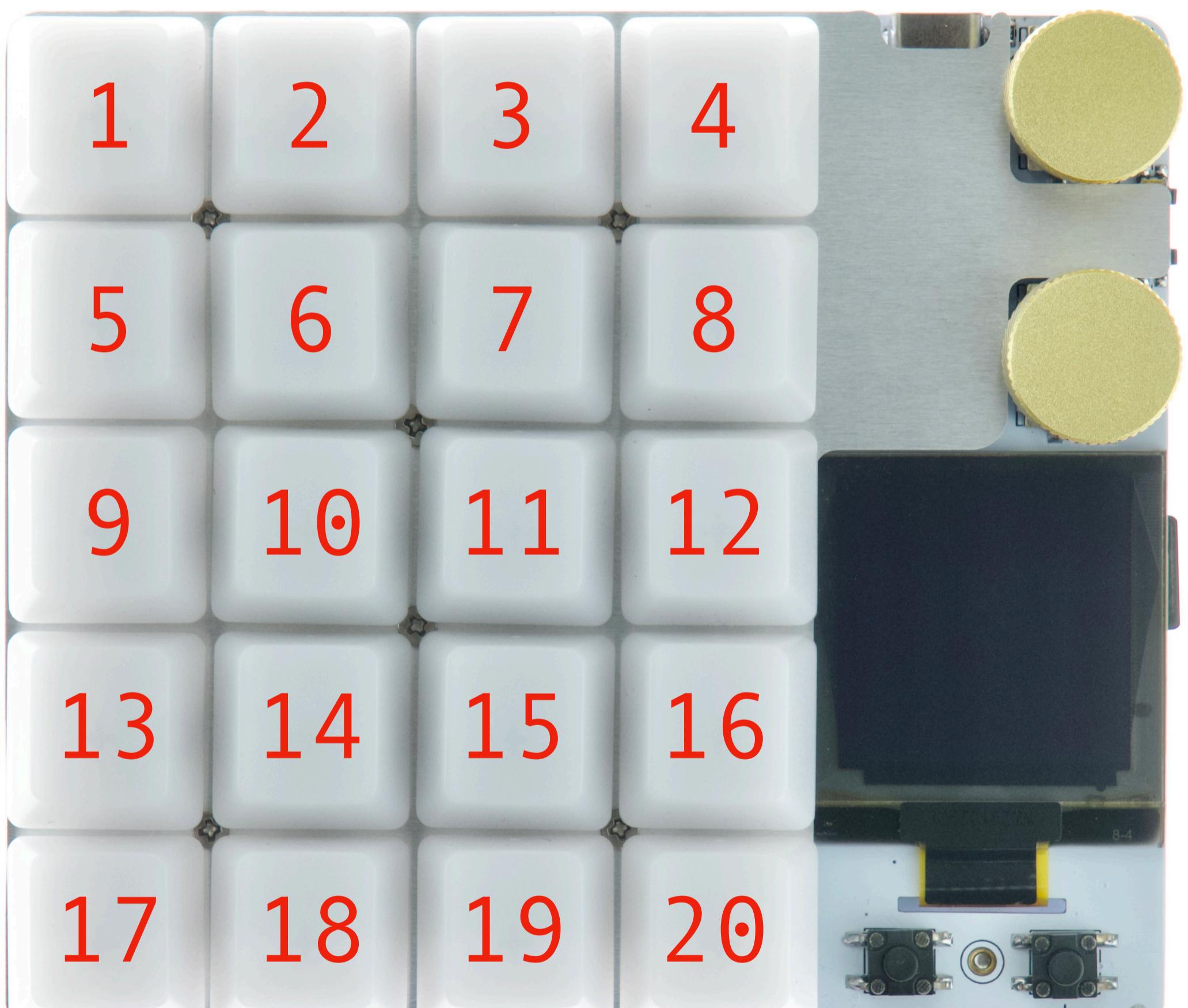
duckyPad Quick Ref 4/6

duckyPad Standard Library	
	Additional Helper Functions
	Add USE_STDLIB to include <a href="#">Click for details</a>
User Headers	
	Click “Edit Header” Button
	Write your own header
	Add USE_UH to include
Built-in Functions	
	Low-level Operations
	See <a href="#">Full Guide</a> for details
RANDCHR(value)	RANDINT(lower, upper)
PUTS(value)	HIDTX(addr)
Memory Access	
Read Signed	PEEK8(addr) PEEK16() PEEK32()
Read Unsigned	PEEKU8(addr) PEEKU16(addr)
Write	POKE8(addr, val) POKE16() POKE32()
For use in Scratch Memory Area <a href="#">Learn more about duckStack VM</a>	

Reserved Variables			
<code>_TIME_S</code>	RO	Elapsed time since power-on	
<code>_TIME_MS</code>			
<code>_BLOCKING_READKEY</code>			
<code>_READKEY</code>	RO	See “Reading Inputs”	
<code>_SW_BITFIELD</code>			
<code>_KBLED_BITFIELD</code>	RO	Keyboard LED Status	
<code>_IS_NUMLOCK_ON</code>			
<code>_IS_CAPSLOCK_ON</code>	RO	0: LED OFF 1: LED ON	
<code>_IS_SCROLLLOCK_ON</code>			
<code>_ALLOW_ABORT</code>	RW	0: Enable 1: Disable	
<code>_DONT_REPEAT</code>			
<code>_THIS_KEYID</code>	RO	See “Key ID”	
<code>_DP_MODEL</code>	RO	1: duckyPad, 2: duckyPad Pro	
<code>_KEYPRESS_COUNT</code>	RO		
<code>_RTC_IS_VALID</code>		Check this first before reading RTC	
<code>_RTC_YEAR</code>		4 Digits, e.g. 2025	
<code>_RTC_MONTH</code>		1 - 12	
<code>_RTC_DAY</code>		1 - 31	
<code>_RTC_HOUR</code>	RO	0 - 23	
<code>_RTC_MINUTE</code>		0 - 59	
<code>_RTC_SECOND</code>		0 - 60	
<code>_RTC_WDAY</code>		Day of Week, 0 = Sunday	
<code>_RTC_YDAY</code>		Day of Year, 0 - 365	
<code>_RTC_UTC_OFFSET</code>	RW	In Minutes	

# Key IDs: duckyPad Pro

- Each key on duckyPad has a **unique ID**
  - Used for **reading button status** and changing RGB colour



Rotary Encoder	Clockwise	Counter Clockwise	Press
Upper	21	22	23
Lower	24	25	26

Expansion Module	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8
Closest to duckyPad	37	38	39	40	41	42	43	44
2nd Closest	45	46	47	48	49	50	51	52

etc.

# Key IDs: duckyPad (2020)

duckyPad Quick Ref 6/6

