duckyScript Cheatsheet: Key Combos

Latest Resources

- duckyPad Homepage
- duckyScript Full Doc
- Autoswitcher
- Syntax Highlighter
- Discord Chatroom

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**

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

duckyScript Commands

Typing

`STRING text`	Type text AS-IS
`STRINGLN text`	Type text AS-IS Press `ENTER` at end
`STRING_BLOCK` `END_STRING`	Type text block AS-IS No new lines
`STRINGLN_BLOCK` `END_STRINGLN`	Same as above Press `ENTER` after each line

Timing

`DELAY n`	Wait n Milliseconds
`DEFAULTDELAY n`	How long to wait between each non-letter keypresses
`DEFAULTCHARDELAY n`	How long to wait between typing each letter
`CHARJITTER n`	Additional random delay between 0 and n ms after typing each letter

Mouse

`LMOUSE`
`MMOUSE`
`RMOUSE`
`FMOUSE`
`BMOUSE`
	Left / Middle / Right
	Forward / Backward
	Mouse Button Click

`MOUSE_MOVE x y`	Move Mouse in Pixels x: + Right - Left y: + Up - Down
`MOUSE_SCROLL h v`	Scroll horizontal h lines vertical v lines

OLED

`OLED_CLEAR`	Clear Screen
`OLED_CURSOR x y`	Set Cursor `x y`: 0 to 127 (0,0) = Top Left
`OLED_PRINT text`	Print Text at Current Cursor
`OLED_CPRINT text`	Print Text **Center-Aligned**
`OLED_CIRCLE` `x y radius opts`	`x y`: Position `radius`: Pixels
`OLED_RECT` `x1 y1 x2 y2 opts`	`x1 y1`: Start `x2 y2`: End
`Opts` 0 = White & Outline 1 = White & Fill	2 = Black & Outline 3 = Black & Fill
`OLED_LINE` `x1 y1 x2 y2`	`x1 y1`: Start `x2 y2`: End
`OLED_UPDATE`	Commit Changes to Screen
`OLED_RESTORE`	Show Default Screen

RGB Backlight

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

Pressing Keys

Key Combos	See Page 1
`KEYDOWN key`	Hold key
`KEYUP key`	Release Key
`key` can be letter, number, or special key.	

Profile Switching

`PREV_PROFILE`	`NEXT_PROFILE`
`GOTO_PROFILE name`	Case sensitive
Also check out duckyPad Autoswitcher	

Miscellaneous

`DP_SLEEP`	RGB & Screen OFF Halts Execution
`HALT`	Halt Execution
`REPEAT n`	Repeat **line above** n times

Comments

`//`
`REM_BLOCK` `END_REM`

Random Letters

Type a random character
`RANDOM_LOWERCASE_LETTER`
`RANDOM_NUMBER`
`RANDOM_UPPERCASE_LETTER`
`RANDOM_SPECIAL`
`RANDOM_LETTER`
`RANDOM_CHAR`

duckyScript: Advanced Usage

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
+  Add
-  Subtract
*  Multiply
/  Divide
%  Modulus
** Exponent

== Equal
!= Not equal
>  Greater than
<  Less than
>= GTE
<= LTE

&& Logical AND
|| Logical OR
!  Logical NOT
&  Bitwise AND
|  Bitwise OR
^  Bitwise XOR
<< 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 Autoswitcher

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 Standard Library

Additional Helper Functions

Add `USE_STDLIB` to include

Click for details

User Headers

Click "`Edit Header`" Button

Write your own header

Add `USE_UH` to include

Built-in Functions

Low-level Operations

See Full Guide 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

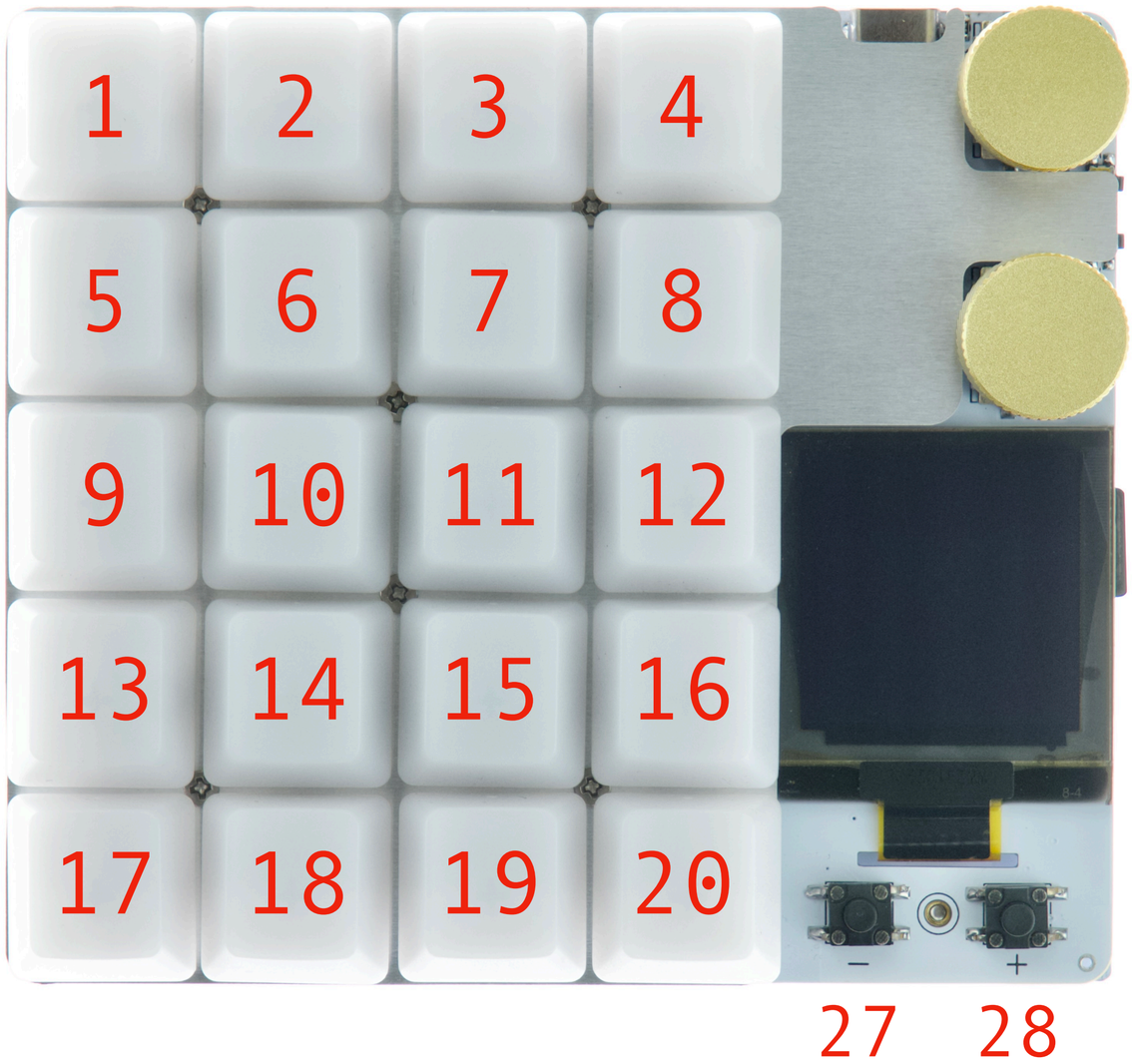
Learn more about duckStack VM

Reserved Variables

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