QuarkEmu Lua Programming API

Introduction

QuarkEmu provide extra `emulator` library for interracting with LUA applications during execution. This library comes with new global variable set and new functions. Some of these functions have no other usage than be involved on TDD (Test Driven Development) approach to make this application. Some other (as chart or simulation) offer special extention to simulate sensors, controls and graph output results.

Global variables

Name	Description
`QEMU_SCREENCOPY_MAIN`	Specifies main window as source of screen copy
`QEMU_SCREENCOPY_TELEMETRY1`	Specifies Telemetry1 window as source of screen copy
`QEMU_SCREENCOPY_TELEMETRY2`	Specifies Telemetry2 window as source of screen copy
`QEMU_SCREENCOPY_CHART`	Specifies Chart window as source of screen copy
`QEMU_AXISTYPE_NOAXIS`	for createChart()
`QEMU_AXISTYPE_VALUE`	for createChart()
`QEMU_AXISTYPE_BARCATEGORY`	for createChart()
`QEMU_AXISTYPE_CATEGORY`	for createChart()
`QEMU_AXISTYPE_DATETIME`	for createChart()
`QEMU_AXISTYPE_LOGVALUE`	for createChart()
`QEMU_RC_TYPE_QUARKEMU`	Default RC Type when USB link not active
`QEMU_RC_TYPE_DC16`	DC 16 Radio control connected
`QEMU_RC_TYPE_DS16`	DS 16 Radio control connected
`QEMU_RC_TYPE_DC14`	DC 14 Radio control connected
`QEMU_RC_TYPE_DS14`	DS 14 Radio control connected
`QEMU_RC_TYPE_DC24`	DC 24 Radio control connected

`QEMU_RC_TYPE_DS24`	DS 24 Radio control connected
`QEMU_PLATFORM_MAC`	App is running on MacOS
`QEMU_PLATFORM_WIN`	App is running on Windows
`QEMU_PLATFORM_LIN`	App is running on Linux
`QEMU_ERROR_TYPE_NONE`	for showMessage()
`QEMU_ERROR_TYPE_INFO`	for showMessage()
`QEMU_ERROR_TYPE_WARN`	for showMessage()
`QEMU_ERROR_TYPE_ERR`	for showMessage()
`QEMU_ERROR_TYPE_CRIT`	for showMessage()

Emulator library

^{tdd} : This flag means that function is only implemented for Test Driven Devlopment usage (for making QuarkEmu) and may have equivalent function in regular Jeti libraries.

Name	Description
time ^{tdd}	Get current time
date ^{tdd}	Get local date since Jan 1st 2000
sleep	Freeze exec until approx. N sec
getFirmwareVersion	Get firmware value set in pref. dialogbox
setFirmwareVersion	Set firmware value
getLocale ^{tdd}	Alias to system.getLocale ()
getUserName ^{tdd}	Get user name value set in preferences dialogbox
getSerialCode ^{tdd}	Get serial code value set in preferences dialogbox
deleteFile	Removes the file specified by the filepath given
getControl	Get attributs of a given control
simulateInput	Affect a given numeric value to a given control
resetInputs	Set all inout control a reset value

createSwitchItem	Fastest way to create SwitchItem datatype
isFuzzyEqual tdd	Fuzzy numbers comparison
getPlatformType tdd	Get currently running platform type
showMessage	Popup a message then optionally raise error
resetSystemProperties	Reset system properties to default values
getInputDeviceType	Get current type of input dev connected to USB
changeInputDeviceType	Force input device to be a given type
copyScreen	Save given window to PNG image
compareScreen	Bitmap comparison of two PNG Image
setCheckMaxWindow tdd	Enable or disable max form/telemetry window check
resetWindowStack tdd	Clean all form or telemetry 1 & 2 windows
getBgColor	Get background color of current selected profil
getFgColor	Get foreground color of current selected profil
enterLoop tdd	Enter in Lua events loop
exitLoop tdd	Exit Lua events loop
simulatePressKey tdd	Send PressKey event to current event loop
createChart	Add new tab in chart dock window, create a new chart window
deleteChart	Delete chart window, remove corresponding tab panel
createSeries	Create a new series attached to a given chart
deleteSeries	
addPointChart	
clearChart	
setChartProperties	
getFormProperty	
sendCommand	
setFormEnableSelection	

setCallbackOnRegisterForm	
setCallbackOnLoopForm	
getSwitchItemAttr ^{tdd}	
resetCPUUsageCount ^{tdd}	

Library functions reference

This section describe all `Emulator` library functions embedded in QuarkEmu. ***Warning*** This library is not supported by Jeti Radio Control systems, and must not be used inside Lua code upload to them.

emulator.time () ^{tdd}

This function get current time by calling posix ::time(NULL) C function.

Parameters

- none

Return value

- (integer) - current time

Examples

```
-- Displays a blinking text
local function printForm()
    if(emulator.getTime() % 2 == 0) then
        lcd.drawText(10,30,"Blinking text",FONT_MAXI)
    end
end
```

emulator.date ([format [, time]]) ^{tdd}

Alias to LUA wellknown function *os.date ([format [, time]])*.

Parameters

- **inputFormat** (string) - see LUA os.date ([format [, time]]) documentation.

Return value

- (string or table) - Returns a string or a table containing date and time, formatted according to the given string format.

Examples

emulator.sleep (<delay>)

Freeze current execution.

Parameters

- **delay** (number) - time delay in second

Return value

- none

Examples

```
local function printForm()
    lcd.drawText(10,30,"text before",FONT_MAXI)
    emulator.sleep(5) -- wait 5 sec
    lcd.drawText(10,30,"text after",FONT_MAXI)
end
```

emulator.getFirmwareVersion ()

Get firmware value set in preferences dialogbox.

Parameters

- none

Return value

- (string) - version value

Examples

```
print(emulator.getFirmwareVersion())
-- print 4.22
```

emulator.setFirmwareVersion (<version>)

Set a new firmware value (same effect than direct setting in preferences dialogbox).

Parameters

- **version** (string) - version value to be set.

Return value

- none

Examples

```
emulator.setFirmwareVersion("4.20")
print(emulator.getFirmwareVersion())
-- print 4.20
```

emulator.getLocale () tdd

Alias to system.getLocale ()

Parameters

- none

Return value

- (string) – locale (cz, de, en, fr, it, pt...).

Examples

- see system.getLocale ()

emulator.getUserName () tdd

Get user name value set in preferences dialogbox.

Parameters

- none

Return value

- (string) – user name.

Examples

emulator.getSerialCode () tdd

Get serial code value set in preferences dialogbox.

Parameters

- none

Return value

- (string) – serial code.

Examples

emulator.deleteFile (<filepath>)

Removes the file specified by the filepath given.

Parameters

- **filepath** (string) - a specified filepath path located on the SD card. Only relative paths to root app directory are used. The path should be accessible.

Return value

- (boolean) – true if successful; otherwise returns false.

Examples

```lua
    emulator.deleteFile("Apps/foo.lua")
```

emulator.getControl (<controlNo>)

The getControl() function retreive attributs of a given control. Prior to calling this function the control must be successfully registered using system.registerControl().

Parameters

- **controlNo** (integer) - desired control identifier (from 1 to 10).

Return value

- (list) - **value** (number), **label** (string), **shortLabel** (string)
- nil – controlNo is not registered.

Examples

```
local ctrlIdx = system.registerControl(1, "LowPass Ctrl", "C01")

system.setControl(ctrlIdx, system.getInputs("P2"), 0, 0)

local value, label, shortLabel = emulator.getControl(1)
```

emulator.simulateInput (<input>, <value>)

Affect a given numeric value to a given control.

Parameters

- **input** (string) – specified input controls.

Allowed controls	Code
Sticks/proportional controls	P1, P2, P3, P4, P5, P6, P7, P8
Switches	SA, SB, SC, SD, SE, SF, SG, SH, SI, SJ, SK, SL
PPM inputs	CH1, CH2, CH3, CH4, CH5, CH6, CH7, CH8
Servo outputs	O1, O2, ..., O24

- **value** (number) - desired value (from -1.0 to 1.0)

Return value

- (boolean) - true when **input** is valid, false otherwise

Examples

```lua
local ctrlIdx = system.registerControl(1, "LowPass Ctrl", "C01")
system.setControl(ctrlIdx, system.getInputs("P2"), 0, 0)
local value, label, shortLabel = emulator.getControl(1)

emulator.simulateInput("P2", -0.5)
```

emulator.resetInputs ()

Set P1 to P8 controls to 0.0, set SA to SL switches to -1.0, set trainer inputs T1 to T16 to 0, set PPM inputs CH1 to CH8 to 0, set servo outputs O1 to 24 to 0.0 .

Parameters

- none

Return value

- none

Examples

```lua
emulator.resetInputs()

-- check sticks/proportional controls
-- check default values
local p1 = system.getInputs("P1")

assert(emulator.isFuzzyEqual(p1,  0.0, 0.015))
```

emulator.createSwitchItem (<input>, <proportional>, <assigned>)

Fastest way to create SwitchItem datatype rather than use form.addInputbox().

Parameters

- input (string) - specified input controls.

Allowed controls	Code
Sticks/proportional controls	P1, P2, P3, P4, P5, P6, P7, P8
Switches	SA, SB, SC, SD, SE, SF, SG, SH, SI, SJ, SK, SL

- proportional (boolean) -
- assigned (boolean) -

Return value

- (SwitchItem) - new SwitchItem datatype

Examples

```
emulator.simulateInput("P1", 0.22)
local switch = emulator.createSwitchItem("P1", true, true)
local val = system.getInputsVal(switch)

assert(emulator.isFuzzyEqual(val, 0.22, 0.015))
```

emulator.isFuzzyEqual (<data1>, <data2>, <precision>) tdd

Fuzzy numbers comparison.

Parameters

- **data1** (number) - first data to compare
- **data2** (number) - second data to compare
- **precision** (number) - maximun unsigned spread allowed between data1 and data2

Return value

- (boolean) - **true** if comparison in lower than precision, otherwise **false**
- nil - when data1 or data2 are not number type

Examples

```
assert(emulator.isFuzzyEqual(0.2, 0.22, 0.015))
-- raise error as abs(0.2-0.22) > 0.015
```

emulator.getPlatformType (<bar>, <bar>) tdd

Get currently running platform type.

Parameters

- none

Return value

- (integer) - return `QEMU_PLATFORM_MAC` , `QEMU_PLATFORM_WIN` or `QEMU_PLATFORM_LIN`

Examples

emulator.showMessage (<message>, [<raiseErrorType>])

Popup a message then optionally raise error.

Parameters

- **message** (string) - Message text displayed on messageBox. HTML format allowed.
- **raiseErrorType** (integer) - must be one of following:

ErrorType	action
`QEMU_ERROR_TYPE_NONE`	no error raised
`QEMU_ERROR_TYPE_INFO`	information raised error reason is **message**
`QEMU_ERROR_TYPE_WARN`	warning raised error reason is **message**
`QEMU_ERROR_TYPE_ERR`	error raised error reason is **message**
`QEMU_ERROR_TYPE_CRIT`	critical raised error reason is **message**

Return value

- none

Examples

```
    emulator.showMessage("<html><font color=\"red\">Tests aborted
            </font>:<br> This test fails.</html>", QEMU_ERROR_TYPE_ERR)
```

emulator.resetSystemProperties ()

Reset system properties to following default values:

Property name	value
WirelessEnabled	0
Volume	8
VolumePlayback	50
VolumeBeep	50
Backlight	500
Color	color set in preferences dialogbox
BacklightMode	3
Model	Model name set in preferences dialogbox
ModelFile	Model filename set in preferences dialogbox

Parameters

- none

Return value

- none

Examples

emulator.getInputDeviceType (<deviceID>)

Get the current type of input device (RadioControl) connected (via USB link) to QuarkEmu.

Parameters

- **deviceID** (integer) - for future use, must always set to 1

Return value

- (integer) - code returned could be:

return code
`QEMU_RC_TYPE_QUARKEMU`
`QEMU_RC_TYPE_DC16`
`QEMU_RC_TYPE_DS16`
`QEMU_RC_TYPE_DC14`
`QEMU_RC_TYPE_DS14`
`QEMU_RC_TYPE_DC24`
`QEMU_RC_TYPE_DS24`

Examples

emulator.changeInputDeviceType (<deviceID>, <deviceType>)

Force input device to be a given type, even if Radio Control is connected to USB link.

Parameters

- **deviceID** (integer) - for future use, must always set to 1
- **deviceType** (integer) - code as defined in emulator.getInputDeviceType

Return value

- (boolean) - true when succeed, false otherwise

Examples

```
emulator.changeInputDeviceType(1, 0x02A6) -- change RC to DC 24
```

emulator.copyScreen (<windowType>, <formNo>, <filepath>)

Save given window to PNG image.

Parameters

- **windowType** (integer) - use following values:

windowType
`QEMU_SCREENCOPY_MAIN`
`QEMU_SCREENCOPY_TELEMETRY1`
`QEMU_SCREENCOPY_TELEMETRY2`
`QEMU_SCREENCOPY_CHART`

- **formNo** (integer) - form identifier
- **filepath** (string) - a specified path located on the SD card. Relative or absolute paths can be used. The path should be accessible.

Return value

- (boolean) - true when succeed, false otherwise

Examples

```
    emulator.copyScreen(QEMU_SCREENCOPY_MAIN, 1, "Apps/img/form1.png")
```

emulator.compareScreen (<windowType1>, <formNo1>, <filepath2>)

Bitmap comparison between image1 and image2.

Parameters

- **windowType** (integer) - Image1, use following values:

windowType
`QEMU_SCREENCOPY_MAIN`
`QEMU_SCREENCOPY_TELEMETRY1`
`QEMU_SCREENCOPY_TELEMETRY2`
`QEMU_SCREENCOPY_CHART`

- **formNo** (integer) - Image1 form identifier
- **filepath** (string) - Image2 path located on the SD card. Relative or absolute paths can be used. The path should be accessible.

Return value

- none

Examples

```
emulator.copyScreen(QEMU_SCREENCOPY_MAIN, 1, "Apps/img/form1.png")
assert(emulator.compareScreen(QEMU_SCREENCOPY_MAIN, 2, "Apps/img/form1.png")
-- compare formNo 1 vs formNo 2
```

emulator.setCheckMaxWindow (<value>) tdd

Enable or disable max form or telemetry window check during registering. Use to bypass 2 max windows allowed.

Parameters

- (boolean) - true for enabling check, false for disabling.

Return value

- none

Examples

emulator.resetWindowStack () tdd

Clean all form or telemetry 1 & 2 windows.

Parameters

- none

Return value

- none

Examples

emulator.getBgColor ()

Get background color of current selected profil in preferences dialogbox.

Parameters

- none

Return value

- (list) - red (integer), green (integer), blue (integer)

Examples

```
    local r,g,b = emulator.getBgColor()
```

emulator.getFgColor ()

Get foreground color of current selected profil in preferences dialogbox.

Parameters

- none

Return value

- (list) - red (integer), green (integer), blue (integer)

Examples

```lua
    local r,g,b = emulator.getFgColor()
```

emulator.enterLoop ([loopCallback]) tdd

Enter in Lua events loop. This loop processed all init(), print() functions attached to forms and windows telemetry.

warning: This function cannot be nested.

Parameters

- **loopCallback** (function) - optional function that is called each time a new loop is started.

Return value

- (boolean) - true if no errors occurs during loop, false otherwise.

Examples

```lua
local function initForm()
end

local function printForm()
   emulator.exitLoop()
end

system.registerForm(1, MENU_MAIN, "foobar", initForm, nil, printForm)

assert(emulator.enterLoop() == true)

system.unregisterForm(1)
```

emulator.exitLoop () tdd

Exit current Lua events loop.

Parameters

- none

Return value

- none

Examples

emulator.simulatePressKey (<keyCode>)

Send PressKey event to current event loop.

Parameters

- **keyCode** (integer) - Use to simulate a button event

Code
KEY_1
KEY_2
KEY_3
KEY_4
KEY_5
KEY_MENU
KEY_ESC
KEY_ENTER
KEY_CW
KEY_ACW
KEY_POWER
KEY_RELEASED

- **keyCode** (string) - Use to simulate a key pressed to selectBox, intBox, ...

Return value

- none

Examples

emulator.createChart (<title>, <tabName>, <tx>, <ty>)

Add a new tab in chart dock window and create a new chart window.

Parameters

- **title** (string) - title centered in the top of chart window.
- **tabName** (string) - tab's name
- **tx** (integer) - X Axis type
- **ty** (integer) - Y Axis type

Axis type
`QEMU_AXISTYPE_NOAXIS`
`QEMU_AXISTYPE_VALUE`
`QEMU_AXISTYPE_BARCATEGORY`
`QEMU_AXISTYPE_CATEGORY`
`QEMU_AXISTYPE_DATETIME`
`QEMU_AXISTYPE_LOGVALUE`

Return value

- (integer) - chartID used by ***deleteChart()***, ***createSeries()***, ***deleteSeries()***, ***addPointChart()***, ***clearChart()***, ***setChartProperties()***

Examples

```
    local chartID1 = emulator.createChart("Chart1", "Chart1", QEMU_AXISTYPE_VALU
```

emulator.deleteChart (<chartID>)

Delete chart window and remove corresponding tab panel.

Parameters

- **chartID** (integer) - ID of chart to delete

Return value

- (boolean) - true id chartID is found and chart deleted, false otherwise

Examples

```
local chartID1 = emulator.createChart("Chart1", "Chart1", QEMU_AXISTYPE_VALU
emulator.deleteChart(chartID1)
```

emulator.createSeries (<chartID>)

Create a new series attached to a given chart.

Parameters

- **chartID** (integer) - ID of chart to add a series into

Return value

- (integer) - serieID

Examples

```
local chartID1 = emulator.createChart("Chart1", "Chart1", QEMU_AXISTYPE_VALU
local seriesID1 = emulator.createSeries(chartID1)
...
emulator.deleteChart(chartID1)
```

emulator.deleteSeries (<charID>, <serieID>)

Remove a given chart serie, then delete the serie.

Parameters

- **chartID** (integer) - ID of chart to delete
- **serieID** (integer) - ID of serie to delete

Return value

- (boolean) - true if succeed

Examples

```
local chartID1 = emulator.createChart("Chart1", "Chart1", QEMU_AXISTYPE_VALU
local seriesID1 = emulator.createSeries(chartID1)
...
emulator.deleteSeries(chartID1, seriesID1)
```

emulator.addPointChart (<chartID>, <serieID>, <x>, <y>)

Add a new point (x,y) to the given serie

Parameters

- **chartID** (interger) - ID of chart
- **serieID** integer) - ID of serie
- **x** (number) - X value of point
- **y** (number) - Y value of point

Return value

- (boolean) - true if succeed

Examples

```
local chartID1 = emulator.createChart("Chart1", "Chart1", QEMU_AXISTYPE_VALU
local seriesID1 = emulator.createSeries(chartID1)
...
emulator.addPointChart(chartID1, seriesID1, 10, 20) -- add point (10, 20)
```

emulator.clearChart (<chartID>, <serieID>)

Remove all points of a given serie, without deleting it.

Parameters

- **chartID** (integer) - ID of chart
- **serieID** (integer) - ID of serie

Return value

- (boolean) - true if succeed

Examples

```
local chartID1 = emulator.createChart("Chart1", "Chart1", QEMU_AXISTYPE_VALU
local seriesID1 = emulator.createSeries(chartID1)
...
emulator.clearChart(chartID1, seriesID1)
```

emulator.setChartProperties (<chartID>, <params>)

Parameters

- **chartID** (integer) - ID of chart
- **params** (table) -

Field name	type	Value
`SamplingRate`	number	from 0 (no sampling) to 5000 ms
`RefreshRate`	number	from 50 to 5000
`ScrollMode`	boolean	***true*** or ***false***
`Title`	string	Text written near axis
`Enable`	boolean	***true*** or ***false***
`Cx`	number	multiplier coef applied to X axis (default 1.0)
`Cy`	number	multiplier coef applied to Y axis (default 1.0)
`AxisX`	table	axis params, see **Axis table format**
`AxisY`	table	axis params, see **Axis table format**

Axis table format

Field name	type	Description
`Title`	string	axis title
`LinePen`	integer	defines the pen styles, see **Axis LinePen style**
`Format`	string	The format string supports the following conversion specifiers, length modifiers, and flags provided by printf() in the standard C++ library: d, i, o, x, X, f, F, e, E, g, G, c

Axis LinePen style

Value	Description
0	No line at all
1	A plain line
2	Dashes separated by a few pixels
3	Dots separated by a few pixels
4	Alternate dots and dashes
5	One dash, two dots, one dash, two dots

Return value

- (boolean) - true of succeed

Examples

```

    chartID1 = emulator.createChart("Radio Control axis", "Axis", QEMU_AXISTYPE_

     local prop =
    {
       ScrollMode = true,
       AxisX = {Title = "Samples",Format = "%d", LinePen = 2},
       AxisY = {Title = "Value", Format = "%.2f"}
    }

    emulator.setChartProperties(chartID1, prop)
```

emulator.getFormProperty (<componentIndex>)

Parameters

- **componentIndex** (integer) - an index of the component that was returned by calling the appropriate form.addXXX() function of the current form.

Return value

- (table) -
    - if componentIndex >= 0

      a table with mixture of supported parameters:

        - visible (boolean) – specifies the form item visibility.
        - enabled (boolean) – specifies that the form item can be focused.
        - label (string) – specifies a new text label (supported only by label, link and intbox).
        - alignRight (boolean) – true for right-aligned text (supported only by label and link).
    - if componentIndex < 0

      a table getting current form IDs

        - ID (integer) - current form ID
        - subID (integer) - current form subID

Examples

emulator.sendCommand (<cmd>)

Send a command to internal debugger or input device layer. All these commands are also available by direct typying them in console.

Parameters

- **cmd** (string) -

Commands	Description
`c`	Continue
`f`	Finish/stop
`s`	Step in function
`n`	Step over function
`p` var	Print on console variable var
`sdl`	Toggle command enable/disable SDL trace

Return value

- none

Examples

emulator.setFormEnableSelection (<mode>)

Enable/disable selection of the current form.

Parameters

- **mode** (boolean) - ***true*** or ***false***

Return value

Examples

emulator.setCallbackOnRegisterForm (<registerCallback>)

Set a Lua callback/hook that has been called after form is registered, with form ID as single parameter.

Parameters

- **registerCallback** (lua function) - callback function

Return value

- (boolean) - true if succeed

Examples

```lua
local function onRegisterCallback(formID)
   if(formID == -3) then  -- -3 is formID of inner input form choice
      ...
   end
end

emulator.setCallbackOnRegisterForm(onRegisterCallback)

system.registerForm(1, MENU_MAIN, "checkFormInputbox", initForm, nil, printF

```

emulator.setCallbackOnLoopForm (<formID>, <loopCallback>)

Set a Lua callback/hook that has been called after form is registered in looping phase, with form ID as single parameter.

Parameters

- **formID** (Lua function) - ID of the form
- **loopCallback** (Lua function) - callback function

Return value

- (boolean) - true if succeed

Examples

```lua
local function onLoopCallback()
     emulator.simulateInput("P1", 0.85)
end

emulator.setCallbackOnLoopForm(formID, onLoopCallback)
```

emulator.getSwitchItemAttr ^{tdd} (<SwitchItem>)

Return attributes of a SwitchItem.

Parameters

- **SwitchItem** (SwitchItem) - The given SwitchItem

Return value

- (table) -

Fields	Type
ID	string
isProportional	boolean
isAssigned	boolean
value	number

Examples

emulator.resetCPUUsageCount ***tdd*** *()*

Reset the counter involved in CPU usage calculation.

Parameters

- none

Return value

- none

Examples

Version history

Version	Description
2017.09.25	First release