

Библиотека функций

Функции доступные из lua-кода

Log(string msg) - вывести сообщение в лог
msg - текст сообщения

PlaySound(string id) - воспроизвести аудиофайл
id - идентификатор аудиофайла. Список загружаемых файлов и их идентификаторы задаются через главный конфигурационный файл (см. «[Главный конфигурационный файл](#)»)

Set(string prop) - установить значение свойства.
prop - строка, вида <адрес>=<значение>, где <адрес> - адрес свойства, а <значение> - значение, которое должно быть присвоено свойству. Если *prop* имеет вид <имя_плагина>:<адрес>=<значение>, то <значение> присваивается соответствующему свойству плагина.

Set(string addr, string value) - установить значение свойства.
Эквивалентно вызову Set(addr..'='..'value)

string Get(string addr) - получить значение свойства
addr - адрес свойства. Может иметь вид <имя_плагина>:<адрес> для получения значения свойства плагина.

SetText(string addr, string textID) - установить значение свойства из строковых данных
addr - адрес свойства
textID - идентификатор строки из строковых данных

string GetText(string textID) - получить строку из строковых данных
textID - идентификатор строки из строковых данных

Include(string luaFileName) - выполнить lua-код из файла. Аналогична функции dofile(...), но с учетом путей
luaFileName - имя файла с кодом

Panel(string viewportName, string panelName) - присоединить панель к вьюпорту. Одну панель можно присоединить только к одному вьюпорту, один раз.
viewportName - имя вьюпорта или пустая строка, если панель надо отвязать от вьюпорта
panelName - имя панели

Text(string textResourceName) - загрузить строковые данные из файла
textResourceName - имя файла со строковыми данными

AddEventHandler(string event, string fname) - добавить ссылку на обработчик события
event - описание события в виде <адрес>[=<значение>] или <имя_плагина>:<адрес>[=<значение>]
fname - имя функции - обработчика события. Функция обработчик события должна иметь вид:

```
function <fname>(ev)
    ...
```

```
    return isContinue  
end
```

ev - таблица вида (описывает событие):

ev.Plugin

ev.Panel

ev.Element

ev.Entry

ev.Property

ev.Value

isContinue - если функция возвращает true, то виртуальная машина продолжит искать подходящие обработчики для этого события

PushEvent(string event) - поместить событие в очередь на обработку

event - событие (вида [<имя_плагина>:]<адрес>[=<значение>])

CheckMouse(string viewportName, number x, number y, bool isClick) - **НЕТ ДАННЫХ**

viewportName - **НЕТ ДАННЫХ**

x - **НЕТ ДАННЫХ**

y - **НЕТ ДАННЫХ**

isClick - **НЕТ ДАННЫХ**

sys.Exit() - закрыть приложение

sys.StartTimer(string id, integer time[, bool once = true]) - запустить таймер

id - имя таймера

time - засекаемое время, мс

once - true - если таймер должен сработать один раз, false - если должен работать циклично.

Если опустить этот параметр, то он по умолчанию считается равным true.

После истечения времени *time*, возникает событие вида: *timer.<id>.finished.=true*. Таймер не очень точный.

sys.StopTimer(string id) - остановить таймер

id - имя таймера

После остановки таймера этой функцией событие таймера не возникает.

sys.GetPanelList() - получить имена всех панелей

Возвращает массив строк с именами панелей

sys.GetElementList(string PanelID) - получить имена элементов в заданной панели

PanelID - имя панели, в которой ищем элементы.

Возвращает массив строк с именами элементов

sys.GetSubList(string PanelID, string ElementID) - Получить имена подэлементов в заданном элементе

PanelID - Имя панели

ElementID - Имя элемента

Возврат - Массив строк с именами подэлементов

sys.GetViewportList() - Получить имена всех вьюпортов

Возврат - Массив строк с именами вьюпортов

sys.GetViewportName(string PanelID) - Получить имена всех вьюпортов

PanelID - Имя панели

Возврат Имя вьюпорта или пустая строка, если такой панели нет или она не присоединена к вьюпорту

`sys.InitViewport(string ID, integer Left, integer Top, integer Width, integer Height, bool Windowed)` -

Добавить новый вьюпорт

ID - Имя вьюпорта

Left - Позиция вьюпорта на десктопе по горизонтали

Top - Позиция вьюпорта на десктопе по вертикали

Width - Ширина вьюпорта

Height - Высота вьюпорта

Windowed - Нужна-ли рамка

Возврат - true - если вьюпорт создан удачно

`sys.InitViewport(string ID, integer Width, integer Height, integer Hwnd)` - Добавить новый вьюпорт

ID - Имя вьюпорта

Width - Ширина вьюпорта

Height - Высота вьюпорта

Hwnd - Хэндл существующего окна

Возврат - true - если вьюпорт создан удачно

`sys.LoadLogic(string fileName)` - Перезапуск виртуальной машины. Все панели отвязываются от вьюпортов.

fileName - имя стартового lua скрипта

`sys.Exec(string id, string cmd)` - Выполнить внешнюю команду в отдельном потоке

id - id (имя) потока

cmd - команда

Возврат - true - если поток запустился успешно

После завершения работы возникает событие: `exec.<id>.finished.=<result_code>`

В текущей реализации одновременно можно запустить на выполнение только одну команду.

`lic.SaveLicence(string key)` Сохранить лицензионный ключ в файле ключа (указан в `configs.xml`)

key - текст ключа

`lic.GetLicence(string serial, string salt)` Сгенерировать ключ из серийного номера и соли

serial - серийный номер устройства

salt - соль

Возврат - лицензионный ключ

`lic.GetSerial()` - Получить серийный номер устройства

Возврат - Серийный номер

В текущей реализации работает только на RPi.

`sys.SavePanelPNG(string PanelName, string FileName, integer Width, integer Height)`

`sys.SaveElementPNG(string PanelName, string ElementName, string FileName, integer Width, integer Height[, number Scale])`

`sys.SaveXML()`

`sys.SaveLib()`

`sys.SetXMLAttr(string Prop)`

`sys.SetLibAttr(string Prop)`

`sys.GetXMLAttr(string Prop)`

`sys.GetLibAttr(string Prop)`

`sys.ShowViewport(string ViewportID)`

ViewportID - Имя вьюпорта

`sys.HideViewport(string ViewportID)`

ViewportID - Имя вьюпорта

`sys.MoveViewport(string ViewportID, number x, number y)`

ViewportID - Имя вьюпорта

`sys.ScaleViewport(string ViewportID, number x, number y, number scale)`

ViewportID - Имя вьюпорта

sys.ResetViewport(string ViewportID)

ViewportID - Имя вьюпорта

sys.GetMousePos(string ViewportID) - Получить координаты курсора мыши относительно вьюпорта

ViewportID - Имя вьюпорта

Возврат - number x, number y - координаты курсора или nil

sys.SetMousePos(string ViewportID, integer x, integer y) - Установить курсор мыши относительно вьюпорта

ViewportID - Имя вьюпорта

x и y - Координаты

sys.ShowCursor(string ViewportID) - Показать курсор мыши

ViewportID - Имя вьюпорта

sys.HideCursor(string ViewportID) - Спрятать курсор мыши

ViewportID - Имя вьюпорта

sys.GrabCursor(string ViewportID) - Запретить курсору мыши выходить за пределы вьюпорта. Этот режим работает только если окно вьюпорта имеет фокус.

ViewportID - Имя вьюпорта

sys.UngrabCursor(string ViewportID) - Выключить режим запрета выхода курсора за пределы вьюпорта.

ViewportID - Имя вьюпорта

sys.RequestFocus(string ViewportID) - Запросить получение фокуса для окна вьюпорта.

Фактическое получение фокуса зависит от ОС.

ViewportID - Имя вьюпорта

sys.HasFocus(string ViewportID) - Запросить состояние фокуса у окна вьюпорта

ViewportID - Имя вьюпорта

Возврат - true - если окно имеет фокус.

sys.SetVPPos(string ViewportID, number x, number y)

sys.SetVPScale(string ViewportID, number ScaleX, number ScaleY, number Scale)

sys.SetVPZoomScale(string ViewportID, number ZoomScaleX, number ZoomScaleY, number ZoomScale)

Функции, доступные из Lua кода в пропертях элементов

Log(string msg) - Вывести сообщение в лог файл

msg - текст сообщения

PlaySound(string name) - Воспроизвести аудиофайл

id - идентификатор аудиофайла. Список загружаемых файлов и их идентификаторы задаются через главный конфигурационный файл.

Set(string prop) - Установить значение своего свойства.

prop - строка, вида <имя_подэлемента>.<имя_свойства>=<значение> или <имя_свойства>.<значение>.

Set(string addr, string value) - Установить значение своего свойства.

Эквивалентно вызову Set(addr..'='..value)

string Get(string addr) - Получить значение своего свойства

addr - адрес свойства. Должен иметь вид <имя_подэлемента>.<имя_свойства> или <имя_свойства>..

SetText(string addr, string textID) - Установить значение своего свойства из строковых данных

addr - адрес свойства

textID - идентификатор строки из строковых данных

string GetText(string textID) - Получить строку из строковых данных

textID - идентификатор строки из строковых данных

PushEvent(string event) - поместить событие в очередь на обработку

event - событие (вида [<имя_плагина>:]<адрес>[=<значение>])

From:

<https://wiki.lambda-mu.com/> - **Lambda-Mu Wiki**

Permanent link:

https://wiki.lambda-mu.com/lm2/ce/s_func?rev=1596531933

Last update: **2020/11/30 14:12**

