

PS-LCD软件速查表 V1.16

说明：
属性——控件的内部变量，在界面运行时，通过改变其值，达到更新控件外观的目的；
方法——控件的内部函数，在界面运行时，通过调用该函数，改变控件行为或状态；
事件——控件的内部敏感变量，当其值发生变化，界面可自动完成：1) 发送事件消息通知主控制器；2) 运行动作脚本；

► 表一： 控件操作

控件名称	属性 Property	方法 Method	事件 Event	事件消息例子 (PS-LCD发出，主控制器接收)	JS脚本例子 (在动作脚本中输入、或者由主控制器发出)	备注
标签 Label	text	clear()	text	E+xxx.text = '5678' 标签xxx的文本内容改为5678	xxx.text="test"; # 设定标签xxx的文本内容为test xxx.clear(); # 清空标签xxx的文本内容	
按钮 Button	text flat checked	setCheckable(bool) setChecked(bool) animateClick(int)	clicked checked	◆ 普通按钮： E+xxx.clicked=1 按钮xxx正被按下一次 ◆ 开关按钮： E+xxx.checked=1 按钮xxx处于按下状态 E+xxx.checked=0 按钮xxx处于抬起状态	xxx.text="按钮1"; # 设定按钮xxx的显示文字为"按钮1" xxx.setChecked(1); # 设置按钮xxx处于按下状态 xxx.animateClick(100); # 执行一次点击按钮xxx操作，100ms后按钮状态自动恢复 xxx.flat=1; # 设置按钮xxx为平坦风格	按钮可设置为普通按钮和开关按钮，由"可"选择"属性控制"
单行文本框 LineEdit	text, selectedText frame cursorPosition readOnly	clear(), selectAll(), cut(), copy(), paste()	text	E+xxx.text = '5678' 文本框xxx的文本内容改为5678 如果主控制器发送xxx.cursorPosition? 字符串，LCD将返回Q+xxx.cursorPosition=2表示当前光标位置在第二个字符	xxx.text = '1234'; # 设置文本框xxx的文本内容为1234 xxx.copy(); # 复制文本框xxx中选定的文本内容 xxx.paste(); # 粘贴复制的内容到文本框xxx xxx.frame=1; # 设置文本框有边框 xxx.readOnly=1; # 设置文本框为只读	cursorPosition为只读
滑动尺 Slider	value, maximum, minimum		value	E+xxx.value= 50 滑动块xxx的当前位置更新为50	xxx.value=20; # 设定滑动块xxx的当前位置为20 xxx.maximum=100; # 设定滑动块位置的最大值为100	
图片 Image	source scaledContents		source		xxx.source="1.gif"; 播放1.gif动画文件 xxx.source="test.jpg"; 显示test.jpg图片文件 xxx.scaledContents=1; 拉伸图片适合控件尺寸	支持的图片格式: bmp, jpg, png, tiff, gif等，图片必须位于picture目录下
进度条 Progress Bar	value, text, textVisible format	reset(), setRange()	value	E+xxx.value = 10; 进度条xxx位置更新为10	xxx.setRange(0, 100); # 设定进度条xxx的位置范围0-100 xxx.text = "123"; xxx.visible = 1; # 设定进度条xxx的文字显示为123并且可见; xxx.format = "%p" # 设定进度条xxx的文字显示百分比 xxx.format = "%v" # 设定进度条xxx的文字显示实际值	
复选框 Check Box	checked		checked	E+xxx.checked=0 单选框xxx选择被取消 E+xxx.checked=1 单选框xxx被选中	xxx.checked = 1; # 选择单选框xxx xxx.checked = 0; # 取消选择单选框xxx	
下拉菜单 Combo Box	currentText, currentIndex, editable		currentIndex	E+xxx.currentIndex = 1 下拉菜单xxx当前选中索引1	xxx.currentIndex=2; # 设定下拉菜单xxx当前索引值为2	
波形 Plot	value				xxx.value = '10, 20, 30'; 刷新三条曲线的Y坐标分别为10、20、30，X坐标自动加一	
仪表盘 Gauge	value				xxx.value=10; 仪表盘指针指到10%量程处	
刻度尺 Thermo	value				xxx.value=10; 刻度尺指针指到10刻度处	
段式数字 LCD Number	value				xxx.value=10; 显示数字10	

定时器 Timer	interval start			E+xxx.timeout=1 定时器xxx到时	xxx.interval=500; 设定定时器xxx的时间间隔为500ms xxx.start=1; 启动定时器xxx	最小interval 需大于等于 500ms
通用属性 适合所有控件	enabled visible x y width height verbose	setFocus()			xxx.visible=0 隐藏控件xxx xxx.visible=1 显示控件xxx xxx.enabled=1 使能控件xxx xxx.enabled=0 禁止控件xxx xxx.verbose=1 设置控件xxx的事件通知有效 xxx.setFocus(); 让控件xxx获得焦点	x y width height属性 为只读，写 无效
风格属性 适合所有控件	styleSheet			styleSheet(样式表)的格式定义如下： "属性名称1: 值; 属性名称2: 值;" 样式表 无事件消息 ，可用的属性名称和值的定义如下：		各属性设置 以分号隔开 ，属性个数 不限
				属性名称	可用值	
				color (字体颜色)	red(红色) green(绿色) rgb(10, 20, 255) 等等	
				background-color (背景色)	同上	
				selection-color (选中部分前景色)	同上	
				selection-background-color (选中部分背景色)	同上	
				border-color (边框颜色)	同上	
				border-style (边框风格)	dashed dot-dash dot-dot-dash dotted double groove inset outset ridge solid none	
				border-width (边框宽度)	1px (一个像素) 2px (两个像素) 等等	
				border-radius: (边框弧度)	同上	
				margin (外边框)	同上	
				padding (内边框)	同上	
				icon-size (图标大小)	同上	
				font-size (字体大小)	同上	
				font-weight (字体粗细)	normal (正常) bold (加粗) 100 - 900	
				font-style (字体风格)	normal (正常) italic (斜体)	
				text-transform (文本转换)	none uppercase (大写) lowercase (小写)	
				text-align (文本对齐)	top bottom left right center	
				text-decoration (文本装饰)	overline underline line-through blink	
				qproperty-geometry (控件几何属性)	rect(x y w h) (坐标x, y, 宽w, 高h)	
				xxx.styleSheet = "color: green; background-color: black"; # 设置控件xxx的字体颜色为绿，背景色为黑 xxx.styleSheet = "color: rgb(100, 200, 100)"; # 设置控件xxx的字体颜色为 RGB的组合色，红色100，绿色200，蓝色100（每种颜色的最大值为255） xxx.styleSheet = "icon-size: 16px; font-size: 20px; margin: 1px 2px 3px 4px"; # 设置控件xxx的图标尺寸为16x16像素，字体大小为20像素，上、右、下、左空白区分别占1、2、3、4个像素 xxx.styleSheet="border-style: inset; border-width: 5px ; border-color: red;"; # 设置控件xxx的风格为inset，外框宽5个像素，外框颜色为红 xxx.styleSheet="*:indicator { width: 13px; height: 130px}"; # 设置复选框xxx的前框宽度为13像素，高度为130像素 xxx.styleSheet="qproperty-geometry: rect(10 20 30 40);"; # 设置控件的位置坐标为x=10, y=20, 宽30像素，高40像素		

注: xxx为控件的ID, 由用户在designer设计界面时指定

► 表二: 通用函数 (适用于任意界面, 任意控件)

函数名称	用法举例	备注
show()	show("yyy"); 切换到界面yyy	
set()	set("brightness", 10); 设置背光亮度值为10 set("shutdown"); 关闭cooky模块 set("reboot"); 重启cooky模块	
get()	var value = get("brightness"); # 初始化变量value为当前背光亮度值	背光值范围: 0-10

注：其他JS脚本相关函数，请参考JS脚本基础教程

►表三： 通讯函数 （适用于任意界面，建议在全局脚本或者界面动作脚本中调用）

协议模式	函数名称	用法举例	备注
CTP协议 Cooky Talking Protocol	binWrite()	binWrite(100, 0x55, 'A'); 向主控制器发送二进制数： 0x64 0x55 0x41	参数个数最大25
	asciiWrite()	asciiWrite("xxx.value=", 100); 向主控制器发送ASCII码： xxx.value=100	
自定义协议 UserDefine	usrTx()	usrTx(100, 0x55, 'A'); 向主控制器发送二进制数： 0x64 0x55 0x41	参数个数最大25
	usrRx()	usrRx(); 读取接收数据缓冲区数据，返回值为对应大小的数据字节数组	只能在接收回调函数中使用
	usrRxSetHnd()	usrRxSetHnd("xxx", 10); 设置接收回调函数名称为xxx， 调用的触发条件：接收缓冲区数据超过10字节； 回调函数xxx由用户自定义，声明应为 function xxx(size) , 其中size为调用该函数时，可读取的数据字节数；	
	usrRxClr()	usrRxClr(); 清空接收缓冲区数据	

注： 1） PS-LCD运行时，只能选择其中一种协议模式：协议模式在designer中“界面配置”菜单中设定。

2）如果工作在CTP协议模式下，当系统启动完毕后，PS-LCD将发出如下CTP消息**E+sys.state=ready**（后有回车字符作为结束符）