

# 键鼠魔盒命令说明文档

所有命令**不分大小写**，参数如无特别说明，**分大小写**。

## 一、键盘命令

### 1、KeyPress 键盘按键

**命令格式：**KeyPress 按键, 次数

**参数 1：**按键名称或按键虚拟键码。按键名不分大小写。

支持的按键如下表：

序号	按键名	按键别名	备注
1	Enter		回车键
2	Esc		Esc 键
3	`		
4	BackSpace	Back Space	退格键
5	Tab		Tab 键
6	Space		空格
7	-		减号
8	=		等号
9	[		左中括号
10	]		又中括号
11	\		反斜杠
12	;		分号
13	'		单引号
14	,		逗号
15	.		点
16	/		斜杠
17	CapsLock	Caps Lock	大小写键
18	Print Screen		截屏键
19	ScrollLock	Scroll Lock/Scroll	滚屏锁定键
20	Pause	Break	暂停键
21	Insert		插入键

22	Home		Home 键
23	Pageup	Page Up	向上翻页键
24	Delete	Del	删除键
25	End		End 键
26	PageDown	Page Down	向下翻页键
27	Right		右箭头
28	Left		左箭头
29	Down		上箭头
30	Up		下箭头
31	NumLock	Num Lock	小键盘锁定键
32	小键盘/	Num/ Num /	小键盘/
33	小键盘*	Num*/Num *	小键盘*
34	小键盘-	Num-/Num -	小键盘-
35	小键盘+	Num+/Num +	小键盘+
36	小键盘 Enter	NumEnter/Num Enter	小键盘 Enter
37	小键盘 1	Num1/Num 1	小键盘 1
38	小键盘 2	Num2/Num 2	小键盘 2
39	小键盘 3	Num3/Num 3	小键盘 3
40	小键盘 4	Num4/Num 4	小键盘 4
41	小键盘 5	Num5/Num 5	小键盘 5
42	小键盘 6	Num6/Num 6	小键盘 6
43	小键盘 7	Num7/Num 7	小键盘 7
44	小键盘 8	Num8/Num 8	小键盘 8
45	小键盘 9	Num9/Num 9	小键盘 9
46	小键盘 0	Num0/Num 0	小键盘 0
47	小键盘.	Num./Num .	小键盘.
48	小键盘=	Num=/Num =	小键盘=
49	PowerOff	Power	电源键（一些多媒体键盘才有）
50	静音		静音
51	音量加	volume+	音量加
52	音量减	volume-	音量减
53	LCtrl		左 Ctrl 键
54	LShift		左 Shift 键
55	LAlt		左 Alt 键

56	LWin		左 Win 键
57	RCtrl		右 Ctrl 键
58	RShift		右 Shift 键
59	RAlt		右 Alt 键
60	RWin		右 Win 键
61	Ctrl		Ctrl 键
62	Shift		Shift 键
63	Alt		Alt 键
64	Win		Win 键
65	Menu		菜单键
66	0-9		大键盘数字键
67	A-Z		字母键
68	F1-F12		F1-F12

**参数 2:** 按键次数，如不写，则默认为 1 次。

### **注意事项:**

① 魔盒为完全模拟硬件键盘，如需输入大写，应先按 CapsLock 键再按键，或按住 Shift 再按键。如需输入感叹号的符号，也需按住 Shift 再按键。

② KeyPress 命令在键盘按下和弹起中间已经自动加入 50-110 毫秒的延时，以模拟真实按键操作，如无需延时，需自行使用 KeyDown 和 KeyUp 命令进行组合。

## **2、KeyDown 键盘按下**

**命令格式:** KeyDown 按键, 次数

**参数 1:** 同 KeyPress，按键名称或键码。

**参数 2:** 次数被忽略，可以不写，无论写多少都会只按下一次。

**注意事项：**魔盒分全速模式和低速模式，全速模式支持同时按下除 Ctrl、Alt、Shift、Win 外的 6 个按键，低速模式只支持 5 个按键。默认为全速模式，关于低速模式和全速模式转换，还在开发中。

### 3、KeyUp 键盘弹起

**命令格式：**KeyUp 按键, 次数

**参数 1：**同 KeyPress，按键名称或键码。

**参数 2：**次数被忽略，可以不写，无论写多少都会只弹起一次。

### 4、KeyAllup 全部按键弹起

**命令格式：**KeyAllup

### 5、SayString 输入字符串

**命令格式 1：**SayString "要输入的字符串"

**原理：**翻译为机内码按键，用 KeyDown 和 KeyUp 命令模拟进行输入。可以输入中文、英文、数字、符号、标点等。

使用该输入方法在 Unicode 编码的软件中输出是乱码。

**机内码输入方法：**模拟按住 Alt 然后用小键盘输入一串数字，然后放开 Alt，即可输入一个字。不受输入法影响。

如：Alt+49901=马

**命令格式 2:** SayString 变量名

作用: 将变量数字通过键盘输出。

## 6、SayStringUnicode 输入字符串

**命令格式:** SayStringUnicode "要输入的字符串"

**原理:** 翻译为机内码按键, 用 KeyDown 和 KeyUp 命令模拟进行输入。可以输入中文、英文、数字、符号、标点等。

与 SayString 不同的是采用 Unicode 编码的文字输入。

使用该输入方法在 GBK 编码的软件中输出是乱码。

**机内码输入方法:** 模拟按住 Alt 然后用小键盘输入一串数字, 然后放开 Alt, 即可输入一个字。不受输入法影响。

如: Alt+39532=马

## 7、PressRandom 随机按键

**命令格式:** PressRandom 延时最小, 延时最大, 按键 1, 按键 2... ..

参数: ①延时最小、延时最大: 按键按下后和弹起之间的间隔时间, 单位毫秒, 可以为 0, 最大 254 毫秒。

②按键 1、按键 2... ..: 在这些按键中随机按, 可以是按键名或者虚拟键码。

## 二、鼠标命令

### 1、LeftClick 左键单击

**命令格式:** LeftClick 次数

**参数 1:** 左键单击次数，如不写，则默认为 1 次。

**注意事项:** LeftClick 命令在左键按下和弹起中间已经自动加入 50-110 毫秒的延时，以模拟真实按键操作，如无需延时，需自行使用 LeftDown 和 LeftUp 命令进行组合。

### 2、LeftDoubleClick 左键双击

**命令格式:** LeftDoubleClick

无参数

**注意事项:** LeftDoubleClick 命令在左键按下和弹起中间已经自动加入 50-110 毫秒的延时，两次点击之间加入 60-120 毫秒的延时，以模拟真实按键操作，如无需延时，需自行使用 LeftDown 和 LeftUp 命令进行组合。

### 3、LeftDown 左键按下

**命令格式:** LeftDown

无参数

### 4、LeftUp 左键弹起

**命令格式:** LeftUp

无参数

#### 5、RightClick 右键单击

**命令格式:** RightClick 次数

**参数 1:** 右键单击次数, 如不写, 则默认为 1 次。

**注意事项:** RightClick 命令在右键按下和弹起中间已经自动加入 50-110 毫秒的延时, 以模拟真实按键操作, 如无需延时, 需自行使用 RightDown 和 RightUp 命令进行组合。

#### 6、RightDown 右键按下

**命令格式:** RightDown

无参数

#### 7、RightUp 右键弹起

**命令格式:** RightUp

无参数

#### 8、MiddleClick 中键单击

**命令格式:** MiddleClick 次数

**参数 1:** 中键单击次数, 如不写, 则默认为 1 次。

**注意事项:** MiddleClick 命令在中键按下和弹起中间已经自动加入 50-110 毫秒的延时, 以模拟真实按键操作, 如无需延时,

需自行使用 MiddleDown 和 MiddleUp 命令进行组合。

#### 9、MiddleDown 中键按下

**命令格式:** MiddleDown

无参数

#### 10、MiddleUp 中键弹起

**命令格式:** MiddleUp

无参数

#### 11、MouseAllUp 弹起所有鼠标按键

**命令格式:** MouseAllUp

无参数

#### 12、MouseWheel、Wheel 鼠标滚轮

**命令格式:** MouseWheel 滚动量

**参数 1:** 滚动量在-128 到 127 之间，正数代表向前滚动，  
负数代表向后滚动

**注意事项:** 鼠标滚轮量不同系统滚动量不同，录制出来和  
实际使用时也不同，尽量少用。

#### 13、Move、MoveTo、MoveToEx 鼠标移动



**命令格式:** MoveTo x 坐标, y 坐标

**参数 1:** 鼠标移动目标点的 X 坐标, 可以是数字或者变量或者表达式

**参数 2:** 鼠标移动目标点的 Y 坐标, 可以是数字或者变量或者表达式

**示例:**

MoveTo 100, 100

MoveTo x, y

MoveTo x+10, y-10

**注意事项:** 因整数计算可能有误差, 有个别坐标点可能不能完全准确计算出来。

#### 14、MoveD 带轨迹的鼠标移动

**命令格式:** MoveD x 坐标, y 坐标, 抖动幅度, 移动速度

**参数 1:** 鼠标移动目标点的 X 坐标, 可以是数字或者变量或者表达式

**参数 2:** 鼠标移动目标点的 Y 坐标, 可以是数字或者变量或者表达式

**参数 3:** 鼠标位置抖动幅度, 只能是正整数, 建议不用太大。

**参数 4:** 鼠标两个移动动作的间隔, 单位毫秒, 可以省略。此参数为 0 或者省略, 则默认为 8 (标准鼠标速度)。间隔可以设置时间为 0-63 毫秒。

**命令原理：**魔盒根据抖动幅度，计算目标坐标

$X = x \text{ 坐标} + \text{随机数}(0 \text{ 到抖动幅度})$

$Y = y \text{ 坐标} + \text{随机数}(0 \text{ 到抖动幅度})$

然后生成贝塞尔曲线，有轨迹的移动到目标点。

**注意：**曲线的起始位置为前一个 MoveTo 或 MoveD 命令的终点坐标，非鼠标现在所在位置。

## 15、MoveR 鼠标相对移动

**命令格式：**MoveR dX, dY

**参数 1：**相对 X 坐标，可以是数字或者变量或者表达式

**参数 2：**相对 Y 坐标，可以是数字或者变量或者表达式

**注意事项：**

①相对坐标必须在-32768 至 32767 之间（50 版本增加为此范围，以前版本为-128 至 127，如需更好兼容性，可选择以前版本）

②如采用变量或表达式，大于 32767 和小于-32768 时将发生不可预料的情况。如超过范围，建议自行用循环分割多次进行移动。

## 16、SaveMousePos 保存鼠标当前位置

**命令格式：**SaveMousePos

无参数

**命令原理：**魔盒将前一个鼠标移动命令的目标点进行保存，等待 RestoreMousePos 调用。

**注意事项：**SaveMousePos 保存的命令移动的坐标点，用户操作其他鼠标移动坐标无法获取。

## 17、RestoreMousePos 恢复鼠标保存位置

**命令格式：**RestoreMousePos

无参数

**命令原理：**魔盒将 SaveMousePos 保存的目标点坐标提取出来，移动到这个点。

**注意事项：**RestoreMousePos 不会清除保存的点坐标。如果未用 SaveMousePos 保存坐标，会移动到未知的点上去。

## 18、GetCursorPos 鼠标位置保存到变量中

**命令格式：**GetCursorPos x,y

**参数 1：**保存 X 坐标的目标变量名

**参数 2：**保存 Y 坐标的目标变量名

**注意事项：**GetCursorPos 保存脚本中通过 MoveTo、MoveR 命令移动的坐标点，用户操作其他鼠标移动坐标无法获取。同时保存的是坐标缩放前的坐标值，如脚本参照分辨率和屏幕分辨率不同，将会显示不同值。

此命令对应脚本编辑器中的模拟获取鼠标位置方式。

## 19、GetCursorPosEx 鼠标真实位置保存到变量中

**命令格式：**GetCursorPosEx x,y

**参数 1：**保存 X 坐标的目标变量名

**参数 2：**保存 Y 坐标的目标变量名

**注意事项：**GetCursorPosEx 需与主机通信获取鼠标真实坐标，坐标会经脚本参照分辨率和屏幕分辨率进行同比例缩放。

此命令对应脚本编辑器中的真实获取鼠标位置方式。

## 三、其他命令

### 1、Delay 延时

**命令格式：**Delay 时长

**参数 1：**时长单位为毫秒，支持数字、变量或表达式。

**示例：**

Delay 100

Delay time

**注意事项：**时长最大为 2147483647 毫秒，约 24 天多。

### 2、DelayRandom 随机延时

**命令格式：**Delay 时长最小值，时长最大值

**参数 1：**时长单位为毫秒，支持数字、变量或表达式。

**参数 2:** 时长单位为毫秒，支持数字、变量或表达式。

**示例:**

DelayRandom 100, 200

DelayRandom a, b

DelayRandom 100, time

**注意事项:** 时长最小值应小于时长最大值。如时长为数字，最大支持 32767 毫秒，约 32 秒；如时长为变量最大为 2147483647 毫秒，约 24 天多。

### 3、Dim 定义变量

**命令格式:** Dim 变量 1, 变量 2... ..

**参数:** 变量名，多个变量名用英文逗号隔开

**注意事项:**

①所有变量必须先定义再使用。

②整个脚本支持的变量数为 80 个。

③变量名不能是纯数字。

④变量名不能含有运算符。

⑤变量名不能是关键字。关键字有： and、or、if、while、for、rem、goto、dim、next、sub、return、exit、call、break 等。

⑥变量值只支持整数，不支持浮点数、字符串。

⑦变量定以后都是全局变量，可以在脚本任意位置访问。

#### 4、数组

35 版新增对数组变量的支持。

**格式：**Dim 数组名 (数组长度-1)

如：Dim a(10)代表长度为 11 的数组。

**注意：**①括号必须用英文括号。

②只能使用固定长度的数组。

③每一个数组元素占用变量的数量，所以不宜定义过长。

④只能用于下面有列出的场合使用，其他地方使用会发生问题，而且编译器不会报错。

⑤访问变量时，如果序号超过规定的长度，可能可以访问到其他变量，或者造成硬件死机。编辑器对此无法检查，请规范使用。

**调用方式：**数组名(序号)

序号从 0 到定义时括号内的参数，可以是变量或者运算表达式，甚至是另外数组的元素。

**赋值：**数组名(序号)=数值

数值可以是数字或者变量。

如：a(0)=1    a(i)=b    a(i+2)=a(0)

**取值：**变量=数组名(序号)

其他可使用变量作为参数的命令也可以使用。如：

FindColor 等。

GetPixelColor 等函数数组元素不能用于。

变量作为序号的数组元素不能用于找色、获取鼠标位置、获取分辨率等命令的返回值参数。如确需动态保存在数组里，建议用临时变量保存，再赋值给数组元素。

不建议用于特别复杂的运算中。

## 5、变量运算

变量支持加、减、乘、除、求余等常规运算，分别对应运算符为+、-、\*、/、%。运算式支持括号。优先级：括号>乘除求余>加减。

单行表达式不能太复杂，最多支持 8 步运算。

示例：

```
Dim a, b, c
```

```
a=10
```

```
b=(a+5)*3+6
```

```
c=b*a
```

```
Msgbox c
```

## 6、Sub 子程序

子程序用于把一些重复的动作进行包装，调用一次就可进行一系列动作、运算。调用命令为 Call 子程序名 ()。

**格式:** Sub 子程序名 ()

语句 1

语句 2

语句 3

... ..

End Sub

### **注意事项:**

- ①子程序暂不支持参数。
- ②子程序内可以使用任意变量。
- ③退出子程序的命令为 return 或 Exit Sub。
- ④在子程序中使用 Goto 要注意不要 goto 到子程序外面，容易引起奇怪的问题。

## **7、Call 调用子程序**

**命令格式:** Call 子程序名 ()

**参数 1:** 子程序名，是否带括号都可以

在子程序中 Call 另一个子程序或这个子程序本身要**注意**:  
Call 不能超过 15 层，否则会出现奇怪的错误。脚本编辑器对此无法检查，请自行注意程序逻辑。

## **8、SetScreenScale 设置参照分辨率**

**命令格式:** SetScreenScale 屏幕宽度, 屏幕高度

**参数 1:** 屏幕宽度必须为正整数。



**参数 2:** 屏幕高度必须为正整数。

**命令原理:** 使用 SetScreenScale 设置参照分辨率，几乎等同于在界面的分辨率设置区设置屏幕分辨率。用处为按比例计算鼠标移动、找色等命令的坐标点。如 SetScreenScale 1024, 768，而运行目标设备的分辨率为 1920\*1080，那么，脚本里的 MoveTo 100, 100 就会移动到 (187, 140)，计算方法为

$X = \text{脚本坐标 } X * \text{设备分辨率宽度} / \text{参照分辨率宽度}$

$Y = \text{脚本坐标 } Y * \text{设备分辨率高度} / \text{参照分辨率高度}$

## 四、流程控制命令

### 1、Rem 标记

**命令格式:** Rem 标记名称

**参数 1:** 标记名称可以是任意名字。

**作用:** 用于 Goto 命令的目标位置。

### 2、Goto 跳转

**命令格式:** Goto 标记名称

**参数 1:** 标记可以先定义、也可以后定义。

**作用:** 跳转到标记所在的位置运行下一条指令。

### 3、For 次数循环

**命令格式：**For 次数

语句 1

语句 2

... ..

Next

**参数 1：**次数可以是数字、变量或者表达式

**作用：**循环执行 For 命令和 Next 之间的命令【次数】次。

**注意事项：**For 循环可以嵌套。即 For 循环里还能使用 For 循环。最多嵌套 4 层。

#### 4、For 变量循环

**命令格式：**For 变量=初始值 to 结束值 Step 步长

语句 1

语句 2

... ..

Next

**变量：**参与循环的变量名。

**初始值：**循环开始时变量赋值，支持数字、变量、表达式（表达式中不要含有空格）。

**结束值：**循环结束变量值，支持数字、变量、表达式（表达式中不要含有空格，最好不要使用在循环内部会改变的量或者随机函数）。

**步长：**变量结束一个循环后增加的值，步长可以省略，省略时必须把 Step 也省略了。步长可以是正整数或者负整数。

**原理：**此命令相当于(步长是正数时)：

变量=初始值

While 变量<=结束值

语句 1

语句 2

... ..

变量=变量+步长

Wend

## 5、if 判断

**命令格式：**if 比较表达式 1 then

语句 1

语句 2

... ..

ElseIf 比较表达式 2 then

语句 3

语句 4

... ..

Else

语句 5

语句 6

... ..

Endif

**作用：**判断比较表达式，如果为真，执行 if 和 Else 之间的所有命令；如果为假，执行 Else 和 Endif 之间的命令。

**注意事项：**

①Else 可以省略，那么如果为假时，就跳过这个 if，执行下面的语句。

②Else 可以是 Elseif，然后再接比较表达式，相当于一个新的 if。

③then 可以省略。

④Endif 可以写为 End If。Elseif 可以写为 Else If 或者 elif。

⑤比较表达式的语法参照本章第 6 节。

6、While 条件循环

**命令格式：**While 比较表达式

语句 1

语句 2

... ..

Wend

**作用：**判断比较表达式，如果为真，就运行 While 之间 Wend 包含的语句，直到表达式为假。

## 7、比较表达式

比较表达式是由比较符分隔开的两个整数或变量或表达式。支持的比较符如下表：

序号	含义	比较符
1	大于	>
2	大于等于	>=或=>或!<
3	小于	<
4	小于等于	<=或=<或!>
5	等于	=或==
6	不等于	!=或<>或><

比较表达式用在 if 中支持 and 或者 or 等逻辑判断运算进行合并判断。and 代表且（多个条件全都满足），or 代表或者（其中一个条件满足即可）。

例子：

```
if a>10
```

```
if a>10 and b<99 or b>200
```

While 1 （如果没有比较符，默认添加!=0，以符合 c 语言风格）

```
if a+5>15
```

```
While a>0
```

## 8、break 跳出循环

**命令格式:** break

无参数

**作用:** 跳出当前 While 或者 For 循环。

## 9、EndScript 脚本停止运行

**命令格式:** EndScript

无参数

**作用:** 停止运行脚本。下次按按钮会从头开始运行脚本。

## 10、RestartScript 重新开始脚本

**命令格式:** RestartScript

无参数

## 11、Pause 暂停脚本

**命令格式:** Pause

无参数

**作用:** 在当前位置暂停脚本，通过物理按键或者其他方式继续脚本时从下一行开始。

## 五、图像、颜色有关命令、函数

本章所有命令都必须开启脚本编辑器或者调用 D11 进行辅助才能正常运行！

**注意：**使用找色命令时，请确认关闭 windows 的缩放功能。Win7 在控制面板-》显示，Win10 在设置-》系统-》显示-》缩放与布局。要设为 100%，不然会导致找色的位置不准确。

### 1、GetPixelColor 获取指定点颜色

**命令格式：**结果变量=GetPixelColor(x,y)

**返回值：**结果变量必须是变量名。目标颜色值保存在变量中，为整数形式。可以通过“MsgBox #结果变量”命令以 16 进制显示到本地。格式为 BBGGRR，即前两位为蓝色，中间两位为绿色，后两位为红色。配套使用 GetR、GetG、GetB 函数可以获得颜色中红、绿、蓝颜色的分量。

**参数 1：**坐标 X 可以是数字、变量或表达式

**参数 2：**坐标 Y 可以是数字、变量或表达式

### 2、GetPixelColorHSV 获取指定点的 HSV 颜色

**命令格式：**结果变量=GetPixelColorHSV(x,y)

**返回值：**结果变量必须是变量名。目标颜色值保存在变量中，为整数形式。可以通过“MsgBox #结果变量”命令以 HSV 格式显示到本地。格式为 H.S.V。配套使用 GetH、GetS、GetV 函数可以获得颜色中色调(0-360)、饱和度(0-100)、亮度(0-100)

的分量。

**参数 1:** 坐标 X 可以是数字、变量或表达式

**参数 2:** 坐标 Y 可以是数字、变量或表达式

### 3、FindColor 查找颜色

**命令格式:** FindColor 左上角 X, 左上角 Y, 右下角 X, 右下角 Y, 颜色, 结果 X, 结果 Y

**参数:**

左上角 X: 可以是整数或变量或表达式

左上角 Y: 可以是整数或变量或表达式

右下角 X: 可以是整数或变量或表达式

右下角 Y: 可以是整数或变量或表达式

颜色: 参照第 6 节颜色格式和第 7 节偏色。

结果 X: 必须是变量名

结果 Y: 必须是变量名

**运行结果:** 命令运行后, 结果 X 和结果 Y 变量中保存找到指定颜色的坐标。如果未找到, 结果 X 和结果 Y 为负数。

### 4、FindCenterColor 从中间往外找色

除找色方式和命令名称, 其他和 FindColor 相同, 请参照上一节。



## 5、FindColorEx 模糊找色

**命令格式:** FindColorEx 左上角 X, 左上角 Y, 右下角 X, 右下角 Y, 颜色, 找色方式, 相似度, 结果 X, 结果 Y

### 参数:

左上角 X: 可以是整数或变量或表达式

左上角 Y: 可以是整数或变量或表达式

右下角 X: 可以是整数或变量或表达式

右下角 Y: 可以是整数或变量或表达式

颜色: 参照第 6 节颜色格式和第 7 节偏色。

找色方式: 只能是数字, 0 代表从左上往右下找色, 1 代表从中间往外找色, 2 代表从右下往左上找色。

相似度: 小于等于 1 大于等于 0.3 的两位小数。大于等于 0.99 时相当于完全相同。小于 0.99 时根据颜色差进行计算, 内部颜色差计算方法 (基本与按键精灵计算方法相同): (R 颜色差值的绝对值+G 颜色差值的绝对值+B 颜色差值的绝对值)/100。当颜色是 HSV 颜色时, 相似度=(H 颜色差值的绝对值+S 颜色差值的绝对值+V 颜色差值的绝对值)/100。如: 0.9 相似度代表三种颜色分量最多可以差 9。当颜色中包含偏色时, 忽略相似度。

结果 X: 必须是变量名

结果 Y: 必须是变量名

**运行结果:** 命令运行后, 结果 X 和结果 Y 变量中保存找到指定颜色的坐标。如果未找到, 结果 X 和结果 Y 为负数。

## 6、IfColor 如果指定点颜色

**命令格式：**IfColor 坐标 x, 坐标 y, 颜色, 比较方法 Then  
其他语句

....

EndIf

### 参数：

坐标 x: 可以是数字、变量或表达式

坐标 Y: 可以是数字、变量或表达式

颜色: 参照第 6 节颜色格式。

比较方法: 0 或 1 或 2。0 代表完全相等, 1 代表不相等, 2 代表颜色相似度大于等于 0.9。当比较方法大于 2 时, 看做是相似度, 如 10 相当于 0.1 的相似度。

## 6、颜色格式

颜色格式可以是变量 (由 GetPixelColor 产生的格式)、字符串格式 #RRGGBB 或 BBGGRR 或 H. S. V。

多个颜色字符串用 | 符号拼接而成。如 AABBC|C|#00EEFF|100.45.0。

**注意：**使用 H. S. V 颜色格式时, 因为转化计算误差, 一般用于偏色及模糊找色, 精确找色容易找不到颜色。

## 7、偏色

FindColor、FindCenterColor、FindColorEx、IfColor 支持偏色。当使用偏色时，FindColorEx 的模糊度被忽略。偏色功能需要固件版本 3 以上，编辑器版本 1.05 以上。

偏色格式为颜色-偏色。

如：7700AA-200020，代表 57008A 到 9700CA 颜色范围。

如：100.30.0-10.10.0，代表 90.20.0-110.40.0 颜色范围。

颜色字符串可以拼接多个颜色，带不带偏色都可以混合使用。

## 8、反色

偏色格式前面减号代表反色，指除偏色范围内的颜色，其他都满足条件。

如：-7700AA-200020

如：-100.30.0-10.10.0

## 9、FindMultiColor 多点找色

**命令格式：**FindMultiColor 左上角 X, 左上角 Y, 右下角 X, 右下角 Y, 查找颜色, "偏移颜色", 相似度, 找色方式, 结果 X, 结果 Y

**参数：**

左上角 X: 可以是整数或变量或表达式

左上角 Y: 可以是整数或变量或表达式

右下角 X: 可以是整数或变量或表达式

右下角 Y: 可以是整数或变量或表达式

查找颜色: 参照第 6 节颜色格式和第 7 节偏色。本命令中, 查找颜色不能是变量。

偏移颜色: 偏移颜色必须由英文引号包裹起来。格式为 "x1|y1|颜色, ... ..xn|yn|颜色"。偏移颜色不能过长, 最多约 21 个。xn、yn 代表找到查找颜色后与查找颜色的坐标偏差值。颜色可以是任意颜色格式、偏色、反色, 不能多个颜色拼接, 即不能 x1|y1|颜色 1|颜色 2 这种格式。

相似度: 小于等于 1 大于等于 0.1 的两位小数。等于 1 时代表完全相同。此处相似度算法为按键精灵的算法。当颜色中包含偏色时, 忽略相似度。

找色方式: 只能是数字, 0 代表从从左到右, 从上到下找色, 1 代表从从左到右, 从下到上找色, 2 代表从从右到左, 从上到下找色, 3 代表从右到左, 从下到上找色, 4 代表从中间往外找色。

结果 X: 必须是变量名

结果 Y: 必须是变量名

**运行结果:** 命令运行后, 结果 X 和结果 Y 变量中保存找到指定颜色的坐标。如果未找到, 结果 X 和结果 Y 为负数。

找到: 找到查找颜色所在的坐标, 并满足所有偏移颜色。

未找到: 找不到查找颜色所在的坐标或不是所有偏移颜色

的要求都满足。

#### 10、CaptureMode 设置截图模式

**命令格式：**CaptureMode 0 或者 1

**参数：**0 代表自动截图模式，默认就是 0。1 代表手动截图模式。

**自动截图模式：**编辑器或者其他配合软件启动后默认是自动截图模式。此时每个图色命令时都会自动截取适当大小的屏幕区域进行取色或者找色。

**手动截图模式：**设置为手动截图模式后，用 CaptureScreen 命令进行一次手动截图，截图区域为全屏幕（约耗时 20 毫秒）。截图后到下一次截图期间，所有图色命令都在这张图上进行操作。

**如何选择：**如果同一时间需要多个找色或者取色时，使用手动截图只截一次图可以大大提高效率。但如果只有一个找色，且范围不大，使用手动截图模式反而速度更慢。

前面运行的脚本将编辑器或者 dll 调用设置为手动截图模式后，程序不退出的话将一直是手动模式。

#### 49.4 版新增

CaptureMode 9: 将上一次手动截图或者找图命令自动截图的缓存，保存为 bmp 文件。

## 11、CaptureScreen 手动截图

**命令格式：** CaptureScreen

**无参数**

参照 CaptureMode 命令的说明。

## 12、FindPic 找图

**命令格式：** FindPic 左上角 X, 左上角 Y, 右下角 X, 右下角 Y, "图片路径", 偏色, 相似度, 找色方式, 结果 X, 结果 Y

**参数：**

左上角 X: 可以是整数或变量或表达式

左上角 Y: 可以是整数或变量或表达式

右下角 X: 可以是整数或变量或表达式

右下角 Y: 可以是整数或变量或表达式

图片路径: 图片的文件名, 可以包含全路径。用|符号隔开多个图片。如: a. bmp|b. bmp|C: \图片\c. bmp。

偏色: 偏移颜色格式是 BBGRR。代表找色过程中每个像素点颜色差值在偏色范围内就算符合要求。如图片中一个点颜色是 304050, 而屏幕上点的颜色是 313853, 那么如果偏色是 010203, 则这个点满足要求。

相似度: 小于等于 1 大于等于 0.1 的两位小数。等于 1 时代表完全相同。该命令中相似度分别计算 R、G、B 三种颜色分

量。相似度计算方法位： $(0.99 - \text{相似度}) * 100 = \text{每种颜色分量能偏移的值}$ 。在这个命令中，相似度和偏色可以叠加，如 0.95 的相似度，偏色是 030405，那么实际计算中，偏色相当于 070809。

找色方式：只能是数字，0 代表从从左到右，从上到下找色，1 代表从从左到右，从下到上找色，2 代表从从右到左，从上到下找色，3 代表从右到左，从下到上找色，4 代表从中间往外找色。

结果 X：必须是变量名

结果 Y：必须是变量名

**运行结果：**命令运行后，结果 X 和结果 Y 变量中保存找到指定颜色的坐标。如果未找到，结果 X 和结果 Y 为负数。

### **注意事项：**

1、图片必须是 bmp 格式 24 位位图，建议使用配套的抓点抓色工具生成。

2、当图片路径使用相对路径时，如未打开脚本时，字典应该放在配合找色软件的运行目录，即调用找图功能的脚本编辑器或者 d11 或者自编软件所在的目录。当打开脚本时，需要放在与脚本相同的目录。

3、找图时，软件会判断图片四个角的颜色是否是同样的颜色，如果是同色，则该颜色视为透明色，图片中其余位置出现这个颜色这个位置就判断为匹配。可以用画图或者抓色工具的切图功能将需要处理为透明的像素涂成同一种颜色。

4、为提升速度，配合找图时所有找过的图片都会缓存在内

存中。如果图片有修改，或者更换脚本，建议关闭配合找图的软件再次打开或者脚本初始化时使用 FreePic 命令。

### 13、FindPic2 找图命令 2

语法与 FindPic 完全相同。

**不同之处：**1、采用的匹配算法不同，FindPic 必须每个点满足相似度和偏色条件才能找得到，FindPic2 满足相似度和偏色条件的数量达到图片像素点数的相似度比例就算找到。

2、因上述算法，计算量会比 FindPic 大好多倍，因此速度会慢很多。

### 14、FreePic 清除找图缓存

**命令格式：**FreePic

**参数无**

**作用：**请求主机辅助程序（编辑器或者 dll）清空找图的缓存。

### 15、FindShape 找形状

**命令格式：**FindShape 左上角 X, 左上角 Y, 右下角 X, 右下角 Y, "坐标偏移描述", 相似度, 找色方式, 结果 X, 结果 Y

**参数：**

左上角 X: 可以是整数或变量或表达式



左上角 Y: 可以是整数或变量或表达式

右下角 X: 可以是整数或变量或表达式

右下角 Y: 可以是整数或变量或表达式

坐标偏移描述: 坐标偏移描述必须由英文引号包裹起来。  
格式为 "x1|y1|匹配 1 或不匹配 0, ... ..xn|yn|1 或 0"。坐标偏移描述最多 255 个。xn、yn 代表偏移点与基准点坐标偏差值。匹配的意思是偏移点与基准点颜色在相似度范围内, 不匹配意思是偏移点与基准点颜色不在相似度范围内。

相似度: 小于等于 1 大于等于 0.1 的两位小数。等于 1 时代表完全相同。此处相似度算法为按键精灵的算法。

找色方式: 只能是数字, 0 代表从从左到右, 从上到下找色, 1 代表从从左到右, 从下到上找色, 2 代表从从右到左, 从上到下找色, 3 代表从右到左, 从下到上找色, 4 代表从中间往外找色。

结果 X: 必须是变量名

结果 Y: 必须是变量名

**运行结果:** 命令运行后, 结果 X 和结果 Y 变量中保存找到指定颜色的坐标。如果未找到, 结果 X 和结果 Y 为负数。

找到: 找到基准点所在的坐标, 并满足所有坐标偏移的条件。

未找到: 不是所有坐标偏移的条件都满足。

**建议:** 找形状适合整体基本相同、显示颜色会变、形状不变情况下的找字、找物体的场合。如果颜色固定, 使用多点找

色更合适一些。

**性能：**找形状命令因为满足条件的情况会比较多，运算量大大提高，消耗时间约是同等量多点找色的 2 倍。

#### 16、GetAveColor 获取区域平均颜色 BGR

**命令格式：**结果变量=GetAveColor (x1, y1, x2, y2)

**参数：**x1, y1, x2, y2 为区域左上角坐标和右下角坐标。

**原理：**通过求和区域内所有像素点的 R、G、B 分量，然后取平均值。

#### 17、GetAveHSV 获取区域平均颜色 HSV

**命令格式：**结果变量=GetAveHSV (x1, y1, x2, y2)

**参数：**x1, y1, x2, y2 为区域左上角坐标和右下角坐标。

**原理：**通过求和区域内所有像素点的 H、S、V 分量，然后取平均值。

#### 18、SetDict 设置找字字典

**命令格式：**SetDict 字典路径

**说明：**字典格式同大漠，可使用大漠综合工具生成。

#### 19、FindStrFast 找字

**命令格式：**FindStrFast 左上角 X, 左上角 Y, 右下角 X, 右下

角 Y, "字符串或者数据", 偏色, 相似度, 结果 X, 结果 Y

**参数:**

左上角 X: 可以是整数或变量或表达式

左上角 Y: 可以是整数或变量或表达式

右下角 X: 可以是整数或变量或表达式

右下角 Y: 可以是整数或变量或表达式

字符串或者数据: ①当需要查找文字时, 需要在该命令使用前用 SetDict 命令设置字典。需要查找多个文字, 用|可以隔开多个文字。目前不支持查找一串文字。

②当直接数据作为参数时, 不需要设置字典。数据格式与字典中相同, 取最前面部分, 如 FF911222444888\$E\$0.0.21\$9, 填写 FF911222444888 即可。

偏色: 偏移颜色格式是 BBGGRR-bbggrr 或者 H. S. V-h. s. v。偏色用于处理原始图片, 符合偏色的部分, 视作有颜色, 否则视作无颜色。匹配时, 使用数据包含的黑白图像进行是否相同的运算匹配。

相似度: 小于等于 1 大于等于 0.1 的两位小数。等于 1 时代表完全相同。相似度计算方法为: (有颜色的像素匹配数\*4+无颜色的像素匹配数)/完美匹配的匹配数。

结果 X: 必须是变量名

结果 Y: 必须是变量名

**运行结果:** 命令运行后, 结果 X 和结果 Y 变量中保存找到

指定颜色的坐标。如果未找到，结果 X 和结果 Y 为负数。

### 注意事项：

1、当字典路径使用相对路径时，如未打开脚本时，字典应该放在配合找色软件的运行目录，即调用找图功能的脚本编辑器或者 d11 或者自编软件所在的目录。当打开脚本时，需要放在与脚本相同的目录。

2、字典格式同大漠，可使用大漠综合工具生成。具体使用方法参考：

<https://www.bilibili.com/video/BV1Cb4y1x7dd>

[http://www.360doc.com/content/17/1009/11/48203422\\_693438240.shtml](http://www.360doc.com/content/17/1009/11/48203422_693438240.shtml)

3、此命令不仅可以用于找字，也可以用于找别的图形，参考：<https://zhuanlan.zhihu.com/p/442280288>

## 六、需程序辅助命令

本章所有命令都必须开启脚本编辑器作为辅助工具才能正常运行！

### 1、MsgBox 弹窗

命令格式：MsgBox 弹窗内容

### 注意事项：

①弹窗内容可以是单独字符串或变量。

②如果弹窗内容是变量，将以整数形式显示，如果需要显

示为十六进制，在变量前加#号。

③多个弹窗内容可以通过&符号连接，可以将变量和字符串相连。连接后长度可以突破 16 字节。

④弹窗以后脚本暂停运行，直到用户点击确定。

⑤当使用脚本编辑器时，弹窗内容最后以\$\$\$结尾，则弹窗后脚本不暂停。

## 2、Input 用户输入

**命令格式：**Input 保存变量,提示文字

**作用：**弹出输入框，等待用户输入

**参数：**

保存变量：输入内容保存的变量

提示文字：弹出输入框的标题文字（最多 15 字节，7 个半中文字）

**注意事项：**

①输入的内容只能是数字

②弹出输入框后，脚本暂停运行，直到用户输入内容或关闭输入框。输入框关闭，结果默认为 0。

## 3、Curl 获取网页内容

**命令格式：**Curl 结果变量,网址

**作用：**获取网页的内容，转化为整数，保存在结果变量中。

**参数 1:** 结果变量必须为变量名

**参数 2:** 网址可以由变量和字符串拼接而成，拼接符号为|

**示例:** Curl <http://127.0.0.1:8080/> | id | / | password

**注意事项:**

①仅支持 HTTP 协议，不支持 https、ftp 等其他协议。

②正常网页转换为整数都为 0，因为常规 html 第一个字符为<。

③此功能专门开发用于脚本作者做网络验证，示例代码可以在群共享找到。

#### 4、GetCursorShape 获取鼠标形状

**命令格式:** 结果变量=GetCursorShape()

**注意事项:**

①获取到的结果为一个整数。

②获取到的鼠标形状和按键精灵不同，可能原理不同。

#### 5、GetScreenResolution 获取主机分辨率

**命令格式:** GetScreenResolution x,y

**参数:** 保存结果的两个变量

**作用:** 获取主机实际的分辨率，用于脚本内计算。

#### 6、GetWindowPos 获取窗口位置

**命令格式:** GetWindowPos 结果变量 x, 结果变量 y, 窗口标题, 窗口类名

**作用:** 获取窗口标题和窗口类名指定顶级窗口的坐标。

**参数 1:** 窗口位置的 x 坐标。窗口不存在时返回-1。

**参数 2:** 窗口位置的 y 坐标。窗口不存在时返回-1。

**参数 3:** 窗口标题, 可通过辅助程序“鼠标位置句柄”获取。

47.2 版开始, 窗口标题可以使用\*和?作为通配符。

**参数 4:** 窗口类名, 可通过辅助程序“鼠标位置句柄”获取。

47.2 版开始, 窗口类名可以使用\*和?作为通配符。

**注意事项:**

①如果有多个符合的窗口, 返回最靠前的那个窗口。

②窗口标题和类名不需含双引号。

③获取的为窗口的客户区位置, 不含窗口标题栏和菜单栏。

④本命令获取的坐标不根据参照坐标进行缩放。

## 7、GetWindowSize 获取窗口大小

**命令格式:** GetWindowPos 结果变量宽度, 结果变量高度, 窗口标题, 窗口类名

**作用:** 获取窗口标题和窗口类名指定顶级窗口的客户区大小。

**参数 1:** 窗口客户区的宽度。窗口不存在时返回-1。

**参数 2:** 窗口客户区的高度。窗口不存在时返回-1。

**参数 3:** 窗口标题, 可通过辅助程序“鼠标位置句柄”获取。

**参数 4:** 窗口类名, 可通过辅助程序“鼠标位置句柄”获取。

**注意事项:**

- ①如果有多个符合的窗口, 返回最靠前的那个窗口。
- ②窗口标题和类名不需含双引号。
- ③获取的为窗口的客户区大小, 不含窗口标题栏和菜单栏。
- ④本命令获取的坐标不根据参照坐标进行缩放。

## 8、GetTimeStamp 获取时间戳

**命令格式:** 结果变量=GetTimeStamp()

**作用:** 获取当前时间戳(从北京时间 1970 年 01 月 01 日 08 时 00 分 00 秒到现在的秒数)。

**示例:** 使用时间戳计算现在是几点几分几秒

```
Dim sec, hour, minite
sec=(GetTimeStamp()+28800)%86400
hour=sec/3600
minite=sec%3600/60
sec=sec%60
MsgBox 现在是 & hour & : & minite & : & sec
```

## 9、TracePrint 调试输出

**命令格式:** TracePrint 弹窗内容



**作用：**使用方法同 MsgBox，但将内容输出到文本文件 log.txt 中。

## 10、WaitKey 获取键盘按键

**命令格式 1：**结果变量=WaitKey()

**命令格式 2：**结果变量=WaitKey(虚拟键码, 超时)

**格式 1 作用：**脚本暂停等待键盘按键，当键盘任意键按下时，返回刚才按下的键的虚拟键码。

**格式 2 作用：**脚本暂停等待指定按键（由虚拟键码指定），当该键按下时，赋值结果变量等于 1。当超时时，复制结果变量为 0。

**参数 1：**虚拟键码，为数字，当虚拟键码为 0 时，等同于格式 1，其他值参考下表

虚拟键码	对应按键名
1	鼠标左键
2	鼠标右键
3	Cancel
4	鼠标中键
5	鼠标后退键
6	鼠标前进键
8	Backspace
9	Tab
12	Clear
13	Enter
16	Shift
17	Ctrl
18	Alt
19	Pause
20	Caps Lock
27	Esc
32	Space

33	Page Up
34	Page Down
35	End
36	Home
37	Left Arrow
38	Up Arrow
39	Right Arrow
40	Down Arrow
41	Select
42	Print
43	Execute
44	Snapshot
45	Insert
46	Delete
47	Help
48	0
49	1
50	2
51	3
52	4
53	5
54	6
55	7
56	8
57	9
65	A
66	B
67	C
68	D
69	E
70	F
71	G
72	H
73	I
74	J
75	K
76	L
77	M
78	N
79	O
80	P
81	Q
82	R

83	S
84	T
85	U
86	V
87	W
88	X
89	Y
90	Z
91	左 WIN 键
92	右 WIN 键
93	应用程序键
95	睡眠键
96	小键盘 0
97	小键盘 1
98	小键盘 2
99	小键盘 3
100	小键盘 4
101	小键盘 5
102	小键盘 6
103	小键盘 7
104	小键盘 8
105	小键盘 9
106	小键盘 *
107	小键盘 +
108	小键盘 Enter
109	小键盘 -
110	小键盘 .
111	小键盘 /
112	F1
113	F2
114	F3
115	F4
116	F5
117	F6
118	F7
119	F8
120	F9
121	F10
122	F11
123	F12
144	Num Lock
145	Scroll

160	左 shift
161	右 shift
173	VolumeMute
174	VolumeDown
175	VolumeUp
186	;:
187	= +
188	,
189	- _
190	.
191	/ ?
192	` ~
219	[ {
220	\
221	] }
222	' "

**参数 2:** 超时，单位毫秒，精度为 10 毫秒，最大为 655360 毫秒，即 655.35 秒，代表多少时间内等待按键，当超时小于等于 0 时，代表无限等待。

## 11、GetKeyState 获取按键状态

**命令格式:** 结果变量=GetKeyState(虚拟键码)

**作用:** 获取指定按键当前状态，当按键正按着的时候返回 1，否则返回 0。

**参数:** 虚拟键码，为 1-255 的数字，键码值参考 WaitKey 命令后面的表格。

## 12、Beep 蜂鸣器

**命令格式:** Beep 频率, 时长

频率和时长均需为正整数。省略的话默认是 500。

**作用：**控制主机蜂鸣器发出声音。因操作系统和主板不同，在不同情况下可能使用音响或者主板蜂鸣器发声。

发声时脚本处于暂停状态。

### 13、RunApp 运行程序

**命令格式：**RunApp 运行目标路径

**作用：**启动程序或者打开一个文件

①目标可以是程序、也可以是设置了打开方式的各类文件。

②path 里的程序不需要完整路径。

③目标路径长度不能超过 60 个字符。

### 14、FindFile 文件查找

**命令格式：**FindFile 结果变量, 文件路径, 查找文本

**参数 1：**保存找到文本行号的结果变量，找不到或者文件打不开为-1

**参数 2：**文件路径，如 D:\a.txt，无需引号

**参数 3：**要查找的文本，可以使用\*和?通配符，无需引号

因传输协议限制，文件路径和要查找的文本长度加起来不能超过 58 个字节。

**作用：**打开文本文件，从上往下匹配要查找的文本，匹配到返回文本所在的行号。

比如存在一个文件 a.txt，内容为

123

Aaa

abc

命令 FindFile line,a.txt,abc 运行后，line 变量为 3

## 15、GetFileLine 读取文件

**命令格式：**GetFileLine 结果变量,文件路径,行号变量

**参数 1：**读取到的文本保存的变量（只能读取数字）

**参数 2：**文件路径，如 D:\a.txt，无需引号

**参数 3：**读取的行号，只能使用变量，不能使用数字

**作用：**读取文本文件指定行号的文字，转化为整数保存在变量里。可与上面的 FindFile 配合。

## 七、其他函数

### 1、abs 绝对值

**命令格式：**结果变量=abs(值)

**作用：**计算值的绝对值，即正数还是正数，负数计算为相反的正数。

**参数：**值可以是数字、变量或表达式

## 2、random 随机数

**命令格式：**结果变量=random(最小值,最大值)

**作用：**取最小值到最大值之间的随机整数，不含最大值。

**参数：**最小值和最大值都可以是数字、变量或表达式

## 3、GetR 取红色值

**命令格式：**结果变量=GetR(颜色值)

**作用：**取颜色值中红色分量的值，结果为 0-255 之间。

**参数：**颜色值可以是变量或表达式

**注意：**颜色需要是 GetPixelColor 获取的 RGB 颜色。

## 4、GetG 取绿色值

**命令格式：**结果变量=GetG(颜色值)

**作用：**取颜色值中绿色分量的值，结果为 0-255 之间。

**参数：**颜色值可以是变量或表达式

**注意：**颜色需要是 GetPixelColor 获取的 RGB 颜色。

## 5、GetB 取蓝色值

**命令格式：**结果变量=GetB(颜色值)

**作用：**取颜色值中蓝色分量的值，结果为 0-255 之间。

**参数：**颜色值可以是变量或表达式

**注意：**颜色需要是 GetPixelColor 获取的 RGB 颜色。

## 6、GetH 取色调值

**命令格式：**结果变量=GetH(颜色值)

**作用：**取颜色值中色调分量的值，结果为 0-360 之间。

**参数：**颜色值可以是变量或表达式

**注意：**颜色需要是 GetPixelColorHSV 获取的 HSV 颜色。

## 7、GetS 取饱和度值

**命令格式：**结果变量=GetS(颜色值)

**作用：**取颜色值中饱和度分量的值，结果为 0-100 之间。

**参数：**颜色值可以是变量或表达式

**注意：**颜色需要是 GetPixelColorHSV 获取的 HSV 颜色。

## 8、GetV 取亮度值

**命令格式：**结果变量=GetH(颜色值)

**作用：**取颜色值中亮度分量的值，结果为 0-100 之间。

**参数：**颜色值可以是变量或表达式

**注意：**颜色需要是 GetPixelColorHSV 获取的 HSV 颜色。

## 9、Similarity 颜色相似度

**命令格式：**结果变量=Similarity(颜色值 1, 颜色值 2)

**作用：**计算两个颜色的相似度，返回值为 0-99 之间的整数。



计算方式：99-(R 颜色差值的绝对值+G 颜色差值的绝对值+B 颜色差值的绝对值)

**参数：**颜色值可以是颜色字符串、变量或表达式。颜色字符串格式为 RRGGBB 或#BBGGRR。

## 八、设备相关命令、函数

### 1、GetDeviceTime 获取设备启动时间

**命令格式：**结果变量=GetDeviceTime()

**作用：**将当前设备启动以来的时间保持到变量中，单位为毫秒，最大约 24 天，超过上限以后会变为 0。精度为 10 毫秒。

### 2、GetDeviceID 获取设备序列号

**命令格式：**结果变量=GetDeviceID()

**作用：**将设备序列号保持到变量中，可能是负数。

### 3、定时器

**命令格式：**Timer 定时器编号,启动或停止,间隔时间

**参数 1：**定时器编号为 1-4，共支持 4 个定时器。

**参数 2：**1 代表启动，0 代表停止。

**参数 3：**定时器触发间隔，单位为毫秒，最小需要 10 毫秒。最大支持 655360 毫秒，即 10 分钟 55 秒。

**定时器子函数：**设定定时器后，还需要定时器子函数。定

时器子函数名字格式必须为 on\_timerN N 为定时器编号，不区分大小写。如 Sub on\_timer1()。

**注意：**①因盒子内部计时器精度设置问题，定时器精度为 10 毫秒，即时间间隔设置 101-109 和设置 100 的效果是一样的。

②定时器不会打断一个命令，如 Delay，会等 delay 完成后再运行定时器子函数。

③定时器子函数内部耗时建议尽量短，防止定时器内部运行中，定时器又再次触发，那么就会不停运行这个定时器。

④同一时刻只有一个定时器子函数会运行，只有前面子函数运行完毕，才会运行其他定时器。优先度是 1-4，如 1 和 2 号定时器都触发时，优先运行 1 号。

⑤调用 Timer 命令时如果定时器已经在运行，就会重置定时器计时。

#### 4、GetConfig 获取配置区数据

**命令格式：**结果变量=GetConfig(数据序号)

**参数：**数据序号为 0-31 的数字或者变量，如果发生错误，结果变量会等于 0。

**作用：**配置区为固件 21 版新增功能，长度为 64 个字节，即 32 个-32768 至 32767 的 16 位整数。配置区需由 COM 组件或者 D11 调用 api 进行写入。配置区可写入次数为 1 万次，请勿太频繁写入。

## 5、On\_Pause 子程序

**作用：**当一个子程序名词为 On\_Pause 时（不区分大小写），脚本运行次数完成，或者硬件按钮、热键、D11、命令等各种途径控制硬件暂停脚本时，会自动进行调用，调用完毕后脚本再停止。

### **注意事项：**

- ①On\_Pause 中不要再调用其他 sub，会导致无法全部运行。
- ②On\_Pause 子程序只会在运行的脚本在子程序定义后面时才会生效，故建议在脚本一开始就定义。

## 6、ResumeScript 恢复脚本运行

**作用：**在 On\_Pause 子程序中使用，可以使子程序结束后从暂停的位置继续往下运行，而不是暂停脚本。

**注意事项：**只在 On\_Pause 子程序中起作用。

## 7、Randomize 重置随机数种子

**作用：**重置随机数种子

**参数：**参数可以是数字或者变量，也可以不带。

**注意事项：**1. 当参数是数字或者变量时，以确定的值为种子。如果不带参数，以硬件启动后的毫秒数为种子。

2. 因为硬件启动后毫秒数精度为 10，如果 10 毫秒内 2 次以上调用重置命令，将会重置为相同的数字，导致后续生成的随机数序列一样。

**示例**（以主机的时间戳作为随机数种子）：

```
Dim time  
time=GetTimeStamp()  
Randomize time
```

## 8、ClearStack 清空调用堆栈

**作用：**清空调用堆栈。即子程序调用的记录。

**参数：**无参数

**解释：**因为子程序调用的堆栈最大为 15 层，如果碰到无限嵌套的情况或者子程序内 Goto 到子程序外，会导致堆栈溢出，产生未知的错误。使用该命令后堆栈初始化。那么 Goto 到子程序外或者无限嵌套调用也不会出错。

**注意事项：**①清空堆栈后如碰到 return 命令会无视，继续往下运行，如碰到子程序末尾会继续往子程序下方运行。

②只适合 Goto 到子程序外或者确定不再返回的子程序中使用，其他情况不建议使用。

## 9、GetLED 获取键盘灯状态

**命令格式：**结果变量=GetLED(键盘灯序号)

**作用：** 获取键盘灯的状态并保存到变量，1 为亮，0 为灭。

**参数：** 键盘灯序号：0 为小键盘灯（NumLock）、1 为大写灯（CapsLock）、2 为滚动灯（ScrollLock）