

CtrlKeyWin

create GUI representation of an internal keyboard interface for CtrlBuss that shows the state of keys on the MatchBuss.new

keys from the keyboard are represented on the MatchBuss like a -> \k_a, s -> \k_s
some characters are encoded because they present challenges in the background

```
, \k_comma  
; \k_semicolon  
' \k_btick  
- \k_minus  
= \k_equal  
[ \k_bracketopen  
] \k_bracketclose  
\ \k_backslash  
' \k_tick  
, \k_comma  
. \k_period  
/ \k_slash  
^ \k_circumflex  
$ \k_dollar
```

CtrlKeyWin.new(func, buss)

func - a function

buss - a CtrlBuss

.keyTest(key) test for a match in this.ctrlbuss.match.buss

key - a symbol or array of symbols

.filter (key, direction, arr)

.update

.addKeyboard(num)

num - integer number of keyboard layers

.addLayout(layout, size)

.addLayer(layer)

.addDef(name, pattern)

add a label and pattern the gui. if name is indexed in MatchDef then it is the source for pattern
ignoring the extra argument

name - a symbol

pattern - a symbol or array of symbols

`.setLabel(layer, name, label)`

layer - integer number of keypad layer

name - symbol of the name to store the label under

label - string to display on key

`.at(key)`

retrieve a value from the local environment

`.put(key, value, ..., ..., keyN, valueN)`

put a value into the local environment

`CtrlKeyLabelGUI`

`CtrlKeyLabelDef`