

. , DEPO Sky M70.

- [fly-rotate-no-wm.sh](#)
- 

## fly-rotate-no-wm.sh

```
#!/bin/sh

name=
#nameWacom=

connected=$(xrandr -q --verbose | grep " connected primary")
output=$(echo "$connected" | head -n 1 | sed -e "s/\([a-zA-Z0-9]\+\) connected.*/\1/")

if [ -z "$output" ] ; then
    connected=$(xrandr -q --verbose | grep " connected")
    output=$(echo "$connected" | head -n 1 | sed -e "s/\([a-zA-Z0-9]\+\) connected.*/\1/")
    if [ -z "$output" ] ; then
        echo " "
        exit 1
    fi
fi

#Try to guess about current orientation
#Find the line in "xrandr -q --verbose" output that contains current screen orientation and "strip" out
current orientation.
rotation=$(echo "$connected" | egrep -o '\) (normal|left|inverted|right) \(' | egrep -o
'(normal|left|inverted|right)')
#cut the first connected
rotation=$(echo $rotation | cut -d' ' -f1)

#Get input devices to rotate too
#\t is default delimiter, to set other use -d option, i.e. -d' ' for space
#name=fly-wmfunc FLYWM_GET_INPUT_DEVICES 0x5 | grep -v Wacom | cut -f1 2> /dev/null | tr '\n' ';'
#nameWacom=fly-wmfunc FLYWM_GET_INPUT_DEVICES 0x5 | grep Wacom | cut -f1 2> /dev/null | tr '\n' ';'
#name=fly-wmfunc FLYWM_GET_INPUT_DEVICES 0x5 | cut -f1 2> /dev/null | tr '\n' ';'
name="4:11"
IFS=';'

#stop compton to restart
COMPTON_PID=`ps -C compton -o pid=`
[ ! -z $COMPTON_PID ] && kill $COMPTON_PID

if ! test -z "$1"; then

case "$1" in
    left)
        if [ "$rotation" = "right" ]; then
            xrandr --output $output --rotate normal
            if [ "$?" = 1 ]; then
                fly-dialog --caption " " --msgbox " "
                exit 1
            fi
            sleep 3
        fi
        ;;
    right)
        if [ "$rotation" = "left" ]; then
            xrandr --output $output --rotate normal
```

```

        if [ "$?" = 1 ]; then
            fly-dialog --caption " " --msgbox " "
            exit 1
        fi
        sleep 3
    fi
;;
esac

#Do required rotation
xrandr --output $output --rotate "$1"
if [ "$?" = 1 ]; then
    fly-dialog --caption " " --msgbox " "
    exit 1
fi

case "$1" in
    left)
        for name_i in ${name}; do
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axes Swap" 1
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 1 0
            xinput set-prop --type=int --format=8 "${name_i}" "Wacom Rotation" 2
        done
        ;;
    right)
        for name_i in ${name}; do
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axes Swap" 1
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 0 1
            xinput set-prop --type=int --format=8 "${name_i}" "Wacom Rotation" 1
        done
        ;;
    normal)
        for name_i in ${name}; do
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axes Swap" 0
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 0 0
            xinput set-prop --type=int --format=8 "${name_i}" "Wacom Rotation" 0
        done
        ;;
    inverted)
        for name_i in ${name}; do
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axes Swap" 0
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 1 1
            xinput set-prop --type=int --format=8 "${name_i}" "Wacom Rotation" 3
        done
        ;;
esac

else

rotateTo=
case "$rotation" in
    normal)
        rotateTo="left"
        ;;
    left)
        rotateTo="normal"
        ;;
esac

if ! test -z "$rotateTo"; then
    xrandr --output $output --rotate $rotateTo
    if [ "$?" = 1 ]; then
        fly-dialog --caption " " --msgbox " "
        exit 1
    fi
else
    fly-dialog --caption " " --msgbox " "
    exit 1
fi

dentry=
#if [ -f $HOME/.fly/toolbar/rotate.desktop ]; then

```

```

# dentry=$HOME/.fly/toolbar/rotate.desktop
#fi

# Using current screen orientation proceed to rotate screen and input tools.
case "$rotation" in
    normal)
        #rotate to the left
        if ! test -z "$dentry"; then
            sed -i 's/Icon[ \t]*[\=]*.*$/Icon=object-rotate-left/' "$dentry"
        fi
        for name_i in ${name}; do
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axes Swap" 1
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 1 0
            xinput set-prop --type=int --format=8 "${name_i}" "Wacom Rotation" 2
        done
        ;;
    left)
        #rotate to the normal
        if ! test -z "$dentry"; then
            cat $dentry | sed 's/Icon[ \t]*[\=]*.*$/Icon=object-rotate-right/' > /tmp/dentry.tmp
            mv -f /tmp/dentry.tmp $dentry
        fi
        for name_i in ${name}; do
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axes Swap" 0
            xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 0 0
            xinput set-prop --type=int --format=8 "${name_i}" "Wacom Rotation" 0
        done
        ;;
esac
fi

#alex: need delay?
#sleep 1
#TODO: do recalculation of icon positions instead of lineup
fly-wmfunc FLYWM_ORDER_ICON

[ ! -z $COMPTON_PID ] && compton &

```

:



, **xrandr fly-rotate\* /etc/X11/fly-dm/Xsetup.**

xinput, Virtual core pointer,

```

Virtual core pointer id=2 [master pointer (3)]
Virtual core XTEST pointer id=4 [slave pointer (2)]
Logitech USB Optical Mouse id=11 [slave pointer (2)]
...

```

4 11- . fly-rotate-no-wm.sh

**fly-rotate-no-wm.sh**

name="11;4"

```
id , ";" id ( 2)
```

```
, , :
```

**fly-rotate-no-wm.sh**

```
...
xinput set-prop --type=int --format=8 "${name_i}" "Evdev Axis Inversion" 1 0
...
```

```
1 0
0 1
0 0
1 1
```

```
, /usr/bin /etc/X11/fly-dm/Xsetup ( fly-rotate.sh)
```

**/etc/X11/fly-dm/Xsetup**

```
fly-rotate-no-wm.sh left
```

```
,
```

```
/usr/share/fly-wm/theme/*themerc* ( )
$HOME/.fly/theme/*themerc* ( )
:
```

```
;LockerDPMSInCallback="xinput disable %d"
;LockerDPMSOutCallback="xinput enable %d"
```

```
.. :
```

```
LockerDPMSInCallback=" "
LockerDPMSOutCallback=" "
```

```
:
```

```
LockerDPMSInCallback="xinput enable %d"
LockerDPMSOutCallback="xinput enable %d"
```