

4. Somfy sunscreen

Somfy has a bridge that allows access access from Domoticz to the Somfy controlled sunscreens. That is a possible solution. But I had some trouble buying one, although availability now seems much better.

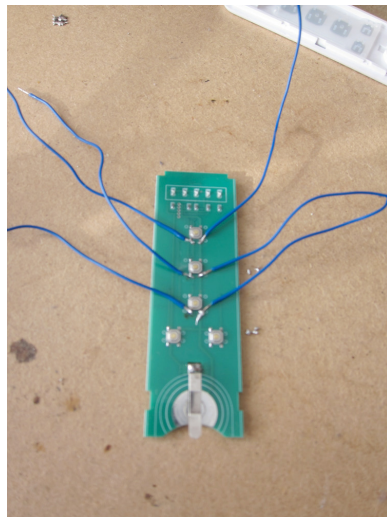
There is an alternative solution for those who are not afraid to use the soldering iron and do some programming.

4.1 Hardware

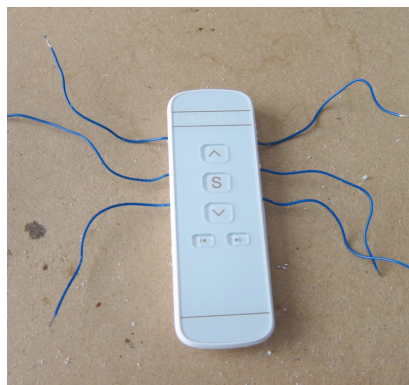
The idea is to control the buttons of a remote with relay. For this, a modified remote is needed and a USB relay card.

There are a number of relatively cheap Somfy remote clones available. I chose one from "<https://123afstandsbediening.nl/>"

I soldered wires on the contact points of the switches.



I made some holes in the sides of the remote to pull-out the wires.



A small blob of thermocol on the wires makes sure that the printed circuit board does not com under stress when handling the wires, I can now control the sunscreen up, down and stop by touching the appropriate wires.

The wires are connected to a USB relay card. I got a simple, 8 relay card from SOS solutions. I'll be using only 3 relay, but the price difference with the 4 relay card is negligible and it allows me to use the relay for other projects.



Connect the wires to the relays so that, when a relay is closed, a button is pushed, and when a relay opens, the button is released. I connected the down button to relay 1, stop to relay 2 and up to relay 3.

4.2 Software

4.2.1 *usbrelay*

First, the relay must be controllable from the Pi. `usbrelay` is an installable package, and if the relay card is in the right state, this is probably the easiest way.

However, if the card is in a wrong state, for example no name is given, then the latest version of `usbrelay` is required. That means compiling from source.

First, install the dependencies:

```
sudo apt install libhidapi-hidraw0 libhidapi-libusb0
sudo apt-get install python-dev cython libudev-dev libusb-1.0-0 libusb-dev libusb-1.0-0-dev libhidapi-dev libavahi-compat-libdnssd-dev
```

Then, get the latest version:

```
git clone https://github.com/darrylb123/usbrelay
```

And then compile:

```
cd usbrelay/
make
sudo make install
```

Because my card had no name attached to it, I had to give it its name.

```
usbrelay -d
```

gives information about the connected board, in my case:

```
libusbrelay: 1.2.1-4-gf858flad9f
usbrelay: 1.2.1-4-gf858flad9f
enumerate_relay_boards()Found 1 devices
Device Found
  type: 16c0 05df
  path: /dev/hidraw0
  serial_number:
Manufacturer: www.dcttech.com
  Product:      USBRelay8
  Release:     100
  Interface:   0
  Number of Relays = 8
  Module_type = 1
_1=0
_2=0
_3=0
_4=0
_5=0
_6=0
_7=0
_8=0
```

The serial number is missing, so that needs to be set. That can be done with

```
sudo ./usbrelay /dev/hidraw0=ZAA
```

or

```
sudo ./usbrelay /dev/hidraw0_0=ZAA
```

depending on what works for your card.

After that, `usbrelay -d` gives

```
libusbrelay: 1.2.1-4-gf858flad9f
usbrelay: 1.2.1-4-gf858flad9f
enumerate_relay_boards()Found 1 devices
Device Found
  type: 16c0 05df
  path: /dev/hidraw0
  serial_number: ZAA
Manufacturer: www.dcttech.com
  Product:      USBRelay8
  Release:     100
  Interface:   0
  Number of Relays = 8
  Module_type = 1
ZAA_1=0
ZAA_2=0
ZAA_3=0
ZAA_4=0
ZAA_5=0
ZAA_6=0
ZAA_7=0
ZAA_8=0
```

4.2.2 Domoticz

Thanks to a plugin, it is possible to launch a script with a slider for the sunscreens.

A Dummy-hardware is used to group the Somfy devices. I called it Somfy-virt. When the hardware is created, virtual sensors can be created. I created one with the type "schakelaar" (switch). It then appears under the tab "Schakelaars" and can be configured from there.



The On and Off actions refer to a script that is under the `domoticz/scripts` directory. The argument, here 2209, is the value of the Idx. That is required because the script needs to retrieve a value from domoticz.

4.2.3 The script

The script I use is as follows.

```
#!/bin/bash

index="$1"
DOMOTICZ='domoticz.home:8888'
VAR=/home/pi/domoticz/var
VALUE=$VAR/value.$index
tdown=20
now=$(date +%s)

sw() {
    rel=$1
    usbrelay ZAA_$rel=1
    sleep 0.1
    usbrelay ZAA_$rel=0
}

mkdir -p $VAR
if [ ! -f $VALUE ]; then
    echo "$now 0" > $VALUE
fi

read prevtime prevvalue < $VALUE

timeago=$((now-prevtime))

nieuwpct=$(curl -s "http://$DOMOTICZ/json.htm?type=devices&rid=${index}" | jq -r '.result[0].Level')

if [ $prevvalue -ge $nieuwpct ]; then
    direction=up
    diff=$((prevvalue-nieuwpct))
else
    direction=down
    diff=$((nieuwpct-prevvalue))
fi

if [ "$nieuwpct" -le 5 ]; then
    nieuwpct=0
elif [ "$nieuwpct" -ge 95 ]; then
    nieuwpct=100
fi

if [ "$nieuwpct" = 0 ]; then
    sw 3
    nieuwpct=0
elif [ "$nieuwpct" = 100 ]; then
    sw 1
    nieuwpct=100
else
    t=$((diff/4))
    if [ "$direction" = "up" ]; then
        sw 3
        sleep $t
        sw 2
    else
        sw 1
        sleep $t
        sw 2
    fi
fi
```

```
echo "$now $nieuwpct" > $VALUE  
logger "somfyrelais $* index=$index nieuwpct=$nieuwpct"
```

The argument of the script is the index of the device in the domoticz list. In that way, it is not necessary to search in the device list.

The function `sw` emulates a short push on a button.

The position of the slider is read using

```
nieuwpct=$(curl -s "http://$DOMOTICZ/json.htm?type=devices&rid=${index}" | jq -r '.result[0].Level')
```

Contents

4. Somfy sunscreen	1
4.1 Hardware	1
4.2 Software	2