



# BATTERY LIMITER

On your "SO SMART" testing window you can enter the receiving end for BL send message. This should be your service or script to communicate with smart plugs.

## Local setup:

Let's say you have a local based script on set on your network, i.e. 192.168.100.100 on port 2000

So, your ENDPOINT would be: `http://192.168.100.100:2000`

BL will send to this address with variable **status** and values 0 or 1 respectively. Testing feature will send every five seconds, interchanging values between 0 and 1, and if your script is running properly it should turn ON/OFF your smart plug depending on values received.

**What should your script do?** Your script should receive posted values automatically, and respond by calling the smart plug interface/service/web application/TCP/UDP (or any other remote control you could find) and send the correct ON or OFF command to the smart plug device.

I suggest you use IFTTT service to access your smart device the easiest way if it is supported in one of the applets. Go and setup your free account for this service <https://ifttt.com/>

After you setup your account you can access all of the available applets or create yours on the way. <https://ifttt.com/applets>

For example, let's say you have WeMo smart plug device. There is an IFTTT applet that let's you toggle the device: <https://ifttt.com/applets/TQFeYtca-toggle-wemo-smart-plug-on-off>

Or even ones to turn ON: <https://ifttt.com/applets/ZKSAmxeD-turn-on-wemo-smart-plug>

Or turn OFF: <https://ifttt.com/applets/K2nG97Qg-turn-off-wemo-smart-plug>

Now, connect your device to these applets, and than you will have access to this applet and control of your device.

After you have managed to setup the correct applet for your device, your script should only decide when to call the applet, i.e. once your script receives status 0 (turn off), call the turn OFF applet.

**Result should be this, your device will be plugged all the time, turning ON/OFF (charging/discharging) will be left to BL and will be done automatically.**

