

ein entferntes vMix via Internet bedienen

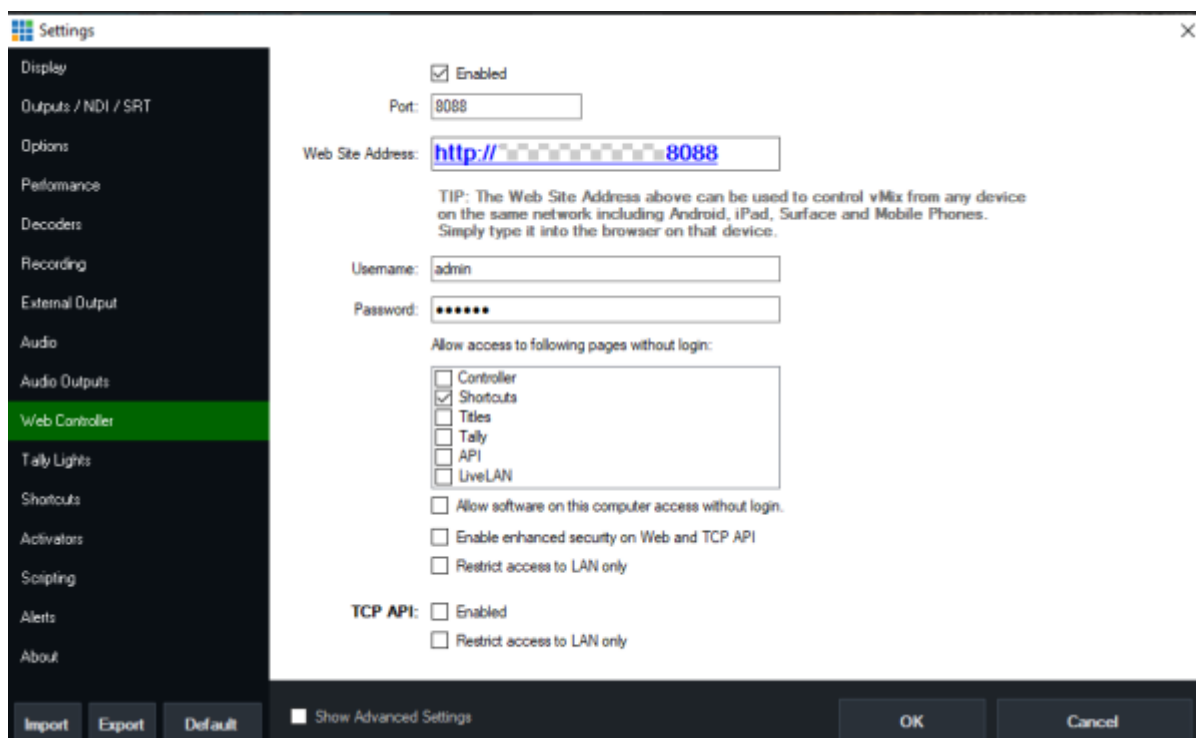
Web Controller

ACHTUNG: der TCP Port ist fix (8099) und falls via Internet geöffnet, nicht mit einem Passwort zu sichern. Also aufgepasst!

Es gibt verschiedene Wege vMix fernzusteuern. Die einfachste Variante ist via integriertes WEB Interface. Auf dieser Seite sehen wir einen Port: Port: Diesen Port müssen wir in unserem Internet Router forwarden. Zu Port Forwarding gibt es in diesem Wiki einen separaten Eintrag. [Port Forwarding](#)

Auch müssen die jeweiligen Checkboxen in vMix aktiviert werden, diese erlauben oder verhindern die externe Kommunikation mit vMix.

OHNE PORTFORWARDING FUNKTIONIERT DAS NICHT!



Kann der Port freigeschaltet werden, steht einer einfachen Bedienung via Internet nichts mehr im Weg. In unserem Fall ist ein Passwort gesetzt, ausser für die Shortcuts. Dies bedeutet, auf der Remote Seite kann nun jemand alle Shortcuts bedienen, aber keine anderen Remote Funktionen wie Controller API etc. Auf dieser Seite lassen sich noch weitere wesentliche Sicherheitsaspekte einschalten. z.B. Zugriff nur via lokales Netzwerk. Ist dieser Punkt aktiv, kann nicht mehr via Internet auf vMix zugegriffen werden. Auf meinem Ipad sieht der Controller folgendermassen aus:

HTML Controller



Dies ist ein einfaches Panel, welches in jedem Desktop Browser läuft. Kopiere den Code ins Notepad, ändere die IP, die Quellen und auf welchen Output es schalten soll. Dann speichere den Text als „irgendwas.html“ auf den Desktop und schon hast Du eine einfache angepasste Fernsteuerung für vMix. So ist es möglich, vierscheidene Controls zu bauen, welche dann nur die benötigten Quellen haben. Die einzige Limite ist leider, dass es nur 4 Outputs hat in vMix, welches aber selbst mit den heutigen Computern anscheinend nicht möglich ist zu erweitern.

```
<!DOCTYPE html>
<meta charset="UTF-8">
<html>
  <script>
    // be aware that with Mix=0 the output switches,
    // so it would be better to give the monitor control a separate mix
output :-) e.g. Mix =1

    // works also from forwarded public ip's -> TESTED
    // declare vMix IP , if your IP is a Public Address, the port 8088
must be open or forwarded.
    // settings in vMix must allow access from external,
    // -vMix > settings > web controller > allow software.../ API /
etc.

    // Mark the code, paste it into a empty notepad window and save it
as control.html
    // or anyothername.html. you can run it then in your browser.
    // the tiny little dot in the upper corner is a checkbox, which
allows to see the sendString

    window.IP = "192.168.10.102";

    // declare buttontext, one per button
    // if you want to take Program, add Program as Input and take the
corresponded Number
```

```
    window.btn1 = "K1";
    window.btn2 = "K2";
    window.btn3 = "K3";
    window.btn4 = "K4";
    window.btn5 = "K5";
    window.btn6 = "K6";
    window.btn7 = "K7";
    window.btn8 = "PGM";

    // declare vMix function, one per button, here Input 1-4 on MIX2
    // Mix=0 is PGM/MIX1/Output
    // Mix=1 is Mix2
    // Mix=2 is Mix3 etc.
    // You can also use a part of the buttons to switch another mix,
    e.g. just change Mix=1 to Mix=2

    window.str1 = "Function=ActiveInput&Input=1&Mix=0";
    window.str2 = "Function=ActiveInput&Input=2&Mix=0";
    window.str3 = "Function=ActiveInput&Input=3&Mix=0";
    window.str4 = "Function=ActiveInput&Input=4&Mix=0";
    window.str5 = "Function=ActiveInput&Input=5&Mix=0";
    window.str6 = "Function=ActiveInput&Input=6&Mix=0";
    window.str7 = "Function=ActiveInput&Input=7&Mix=0";
    window.str8 = "Function=ActiveInput&Input=8&Mix=0";

//-----no changes below here needed, only if you decide to add or delete
buttons -----
    //here the first button is selected at the start of the page
    (autostart)
    window.onload = function () {
        document.getElementById("btn1").click(); };

</script>

<style>
    button {
        background-color:lightgray;
        border-color:black;
        color:black;
        margin-top:30px;
        margin-left:20px;
        width: 200px;
        height: 100px ;
        font-size: 24px ;
        margin-right: 0px;
    }

    h1 {
        font-size: 2em;
        color: beige;
        font-family: Arial;
```

```
    }

    h2 {
        font-size: 5em;
        color: red;
        font-family: Arial;
    }
    p {
        font-size: 1em;
        color: lightgray;
        font-family: Arial;
    }

    input.smallCheckbox {
        width: 7px;
        height:7px;
    }

</style>

    <script type="text/javascript">
        //script for getting checkbox result, if clicked it shows the
vmix command onscreen
        function ShowHideDiv(chkvMixCMD) {
            var divvMixCMD = document.getElementById("divvMixCMD");
            divvMixCMD.style.display = chkvMixCMD.checked ? "block" : "none";
        }
    </script>

    <script>
        //this function sends a command to vMix and takes 3 variables, text,
vmix-Cmd and ID
        //text variable is for displaying which source is choosen
        //vMix variable is the vMix function
        //ID variable for button (1-x)
        //very simple send function, no error handling

        function send2vMix(text,vMix,id) {
            //writes the variable text into TAB
            document.title = text;
            //opens HTTPrequest and sends the vMix Function, defined above
as variables
            const request = new XMLHttpRequest();
            request.addEventListener("load", requestListener);
            request.open("GET", "http://" + IP + ":8088/api/?" + vMix);
            request.send();
            //here all colors are reset
            document.getElementById("btn1").style.backgroundColor = "beige";
            document.getElementById("btn2").style.backgroundColor = "beige";
            document.getElementById("btn3").style.backgroundColor = "beige";
            document.getElementById("btn4").style.backgroundColor = "beige";
```



```

str8,'8')">Input8</button>
  <p></p>

  <div id="dvvMixCMD" style="display: none">
    <p id="myText1" ></p>
  </div>
</body>
</html>

```

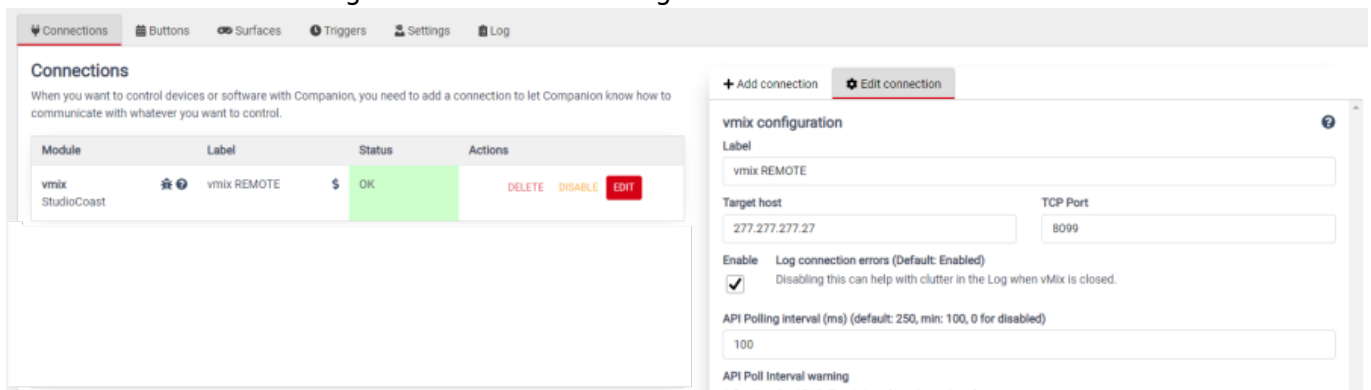
Bitfocus Companion

Eine bessere, aber auch aufwändigere Option ist die Fernsteuerung per Companion. Damit lässt sich ein Streamdeck/Companion/Set vorprogrammieren und zusammen mit dem RaspberryPI4 an die Remote Location senden. Raspi ins Interet bringen, Streamdeck daran anschliessen und los gehts, mit NUR den Funktionen, welche die Remotestelle auch darf. Wie z.B. das eigene Rücksignal steuern.

Companion Port Forwarding

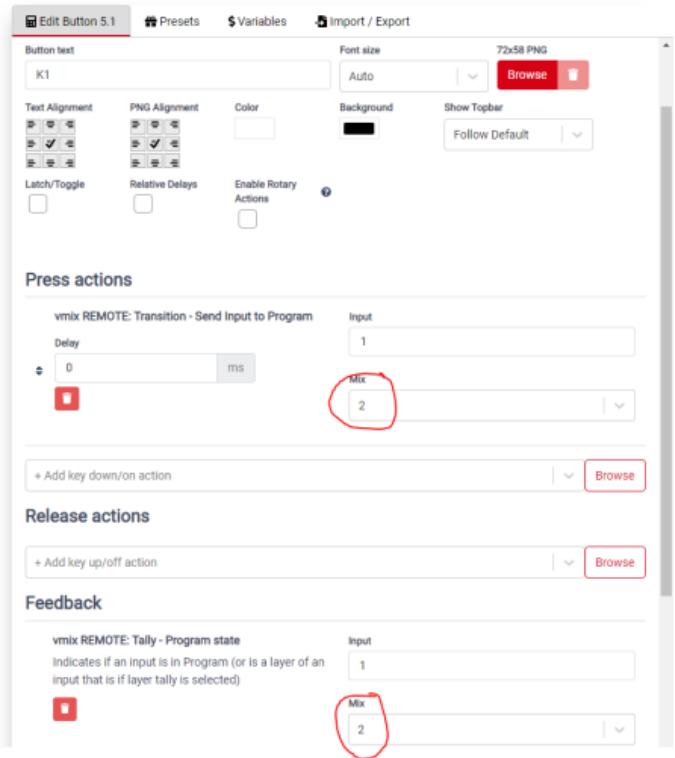
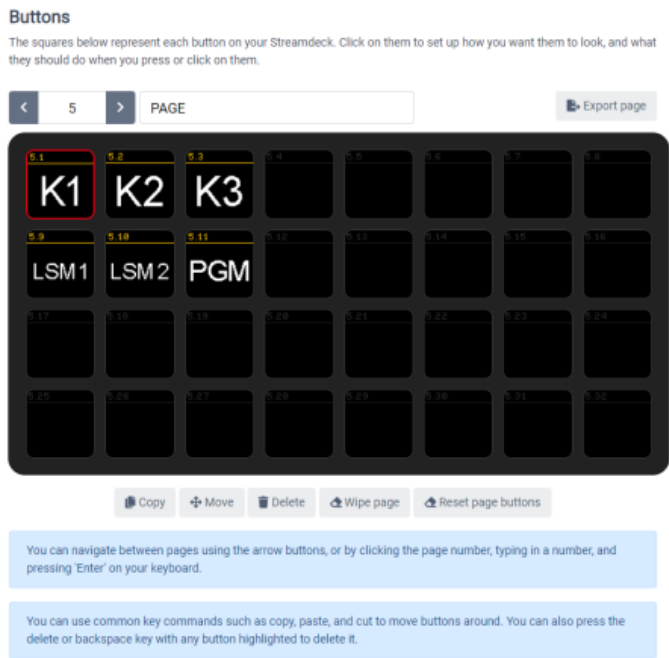
Um Companion zu benutzen muss PORT 8099 zum Zielrechner freigeschaltet werden (port forwarding)

Im Companion Browser legt man dann die Verbindung zur externen IP des Zielrechners an mit Port 8099. Wird die Verbindung GRÜN dann kanns losgehen.



Companion Browser

In diesem Beispiel belegen wir ein kleines 6-Button Streamdeck mit 6 Funktionen für MIX2. MIX2 senden wir als Rücksignal zum externen Kommentator. Dieser kann dann aus 6 Quellen wählen, welche er sehen will. Die Buttons werden auch signalisiert, der Kommentator sieht also, was er angewählt hat.

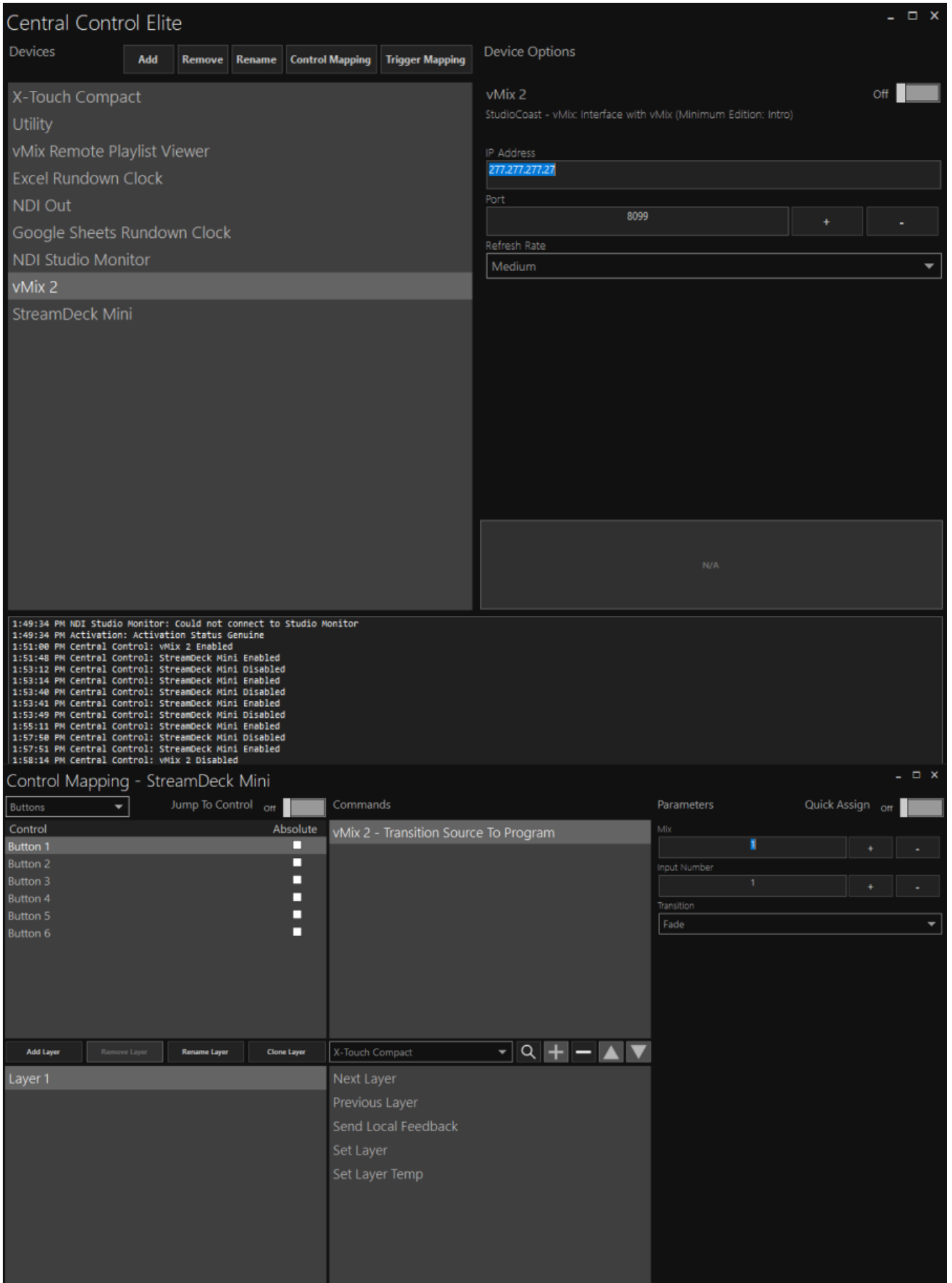


Central Control Intro

Eine weitere, zwar kostenpflichtige, Variante ist die Software [Central Control IO](#). Je nach Version lassen sich damit sehr komplexe Steuerungen realisieren wie:

- 1. 2 vMix parallel laufen lassen
- 2. jegliche Art von Midi Controller einbinden
- 3. Newtek Controller als vMix Bedienung
- 4. Behringer Faderbanken als vMix Audio Kontrolle etc.

Die Einbindung ist einfach und intuitiv, die Buttons lassen sich aus vMix anschreiben etc.



From:

<https://tvcrew.ch/vmix/> - **vMix Wiki Deutsch**

Permanent link:

https://tvcrew.ch/vmix/doku.php?id=remotely_control_vmix

Last update: **2023/10/20 20:27**

