The Griffin PowerMate as flexible DAW and Midi Controller

by Johannes Hoffmann

I have neither the need nor the space (nor the money) for one of the big control surfaces to control my DAW (Logic Pro to be precise). But I always wanted a knob in addition to mouse and keyboard. Ideally this device should be flexible enough to control various things:

1) scrolling/start/stop
2) change plug-in parameters
3) control channel strip volume/pan/send

And I wanted to switch between all of them.

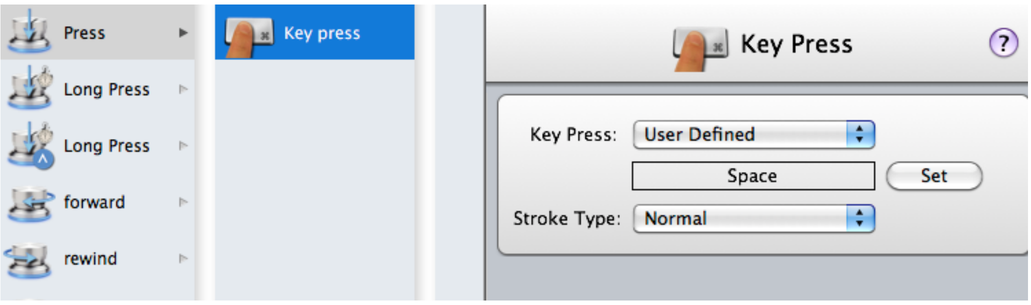
The Griffin PowerMate filled that bill partly for years, but it was more or less limited to one or two tasks and controlling the channel strip of the selected track was cumbersome. But this changed now with the new PowerMate 3 software (free dowload at griffintechnology.com).

Here is how I turned my PowerMate into a universal one knob Logic control (that should work similar for other DAW like Cubase or ProTools). This tutorial shows how to setup each of the three tasks above. If you want to use all three of them you should read down to the switching part bevor starting the setup (otherwise you or the PowerMate or Logic might mix it up).

Download a zip with all scripts.

1. Scrolling the timeline, start and stop.

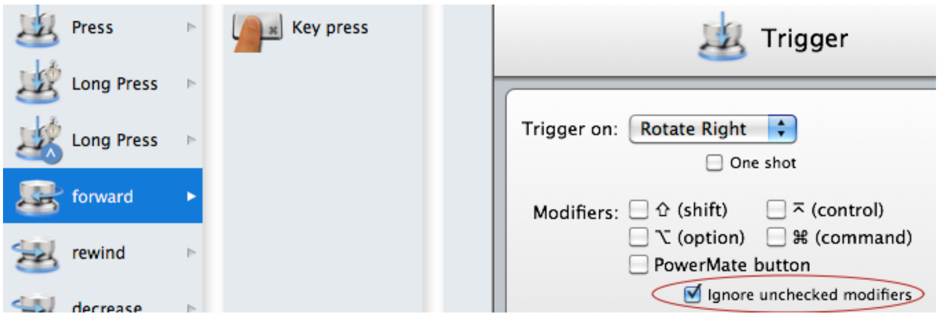
This is the easy task that always worked well. Logic has plenty of key-commands for scrolling, scrubbing, jumping, start and stop. And the PowerMate has been based on keystrokes since day one.



Key Press: User Defined
Space
Stroke Type: Normal

So I defined three triggers in PowerMate 3 (the keystrokes depend on the settings in Logic):

Press: key press »space« (=start/stop)
Rotate right: key press »F7« (=Forward on my setup)
Rotate left: key press »F7« (=Rewind on my setup)



Trigger on: Rotate Right

☐ One shot

Modifiers: ☐ ⇧ (shift) ☐ ⌃ (control)
☐ ⌥ (option) ☐ ⌘ (command)
☐ PowerMate button
☑ Ignore unchecked modifiers

The PowerMate can be configured to pass any modifier to Logic so pressing cmd while turning the know will trigger a fast forward/rewind (as cmd F7/F9 are the keystrokes for this on my system)

2. Plug-in parameter control

This is easy too: Most plug-ins allow parameter change by hoovering with the mouse above a control and using the scroll wheel. So for this task you simply make two triggers in PowerMate 3:

Rotate left: scroll down

Rotate right: scroll up

PowerMate lets you set the sensitivity for each trigger so you can customize the necessary movement to your desire.

3. Controlling the selected channel strip

This was the hard one. Unfortunately Logic ties the scroll wheel to window scrolling. So hoovering above the fader and using the scroll wheel will not change the fader of the Mixer. The only option to access a fader in the Mixer via PowerMate is to use the »increase/decrease last clicked parameter by 1/10« key command in Logic and define key-press-triggers in PowerMate. This works somehow, but I never could get the parameters adjusted in a natural feeling sensitivity. Either I had to scroll endless to get the fader from bottom to top, or (using the »increase by 10« command in Logic) the fader movement was not fine enough.

Unfortunately PowerMate 3 is not able to send Midi. I suggested this feature to the Griffin people during the beta testing but it seems I was not convincing enough ;-) I hope Midi will be implemented in a future version (and I think it would help, if everybody interested in this would drop a note to Griffin).

Embrace MidiPipe

But there is a solution right now: combine PowerMate, AppleScript and MidiPipe. MidiPipe is a powerful Midi rooting and manipulation tool and might scare people who are not familiar with Midi programming stuff. But don’t worry about MidiPipe, you’ll master some AppleScript in a minute too.

How it works

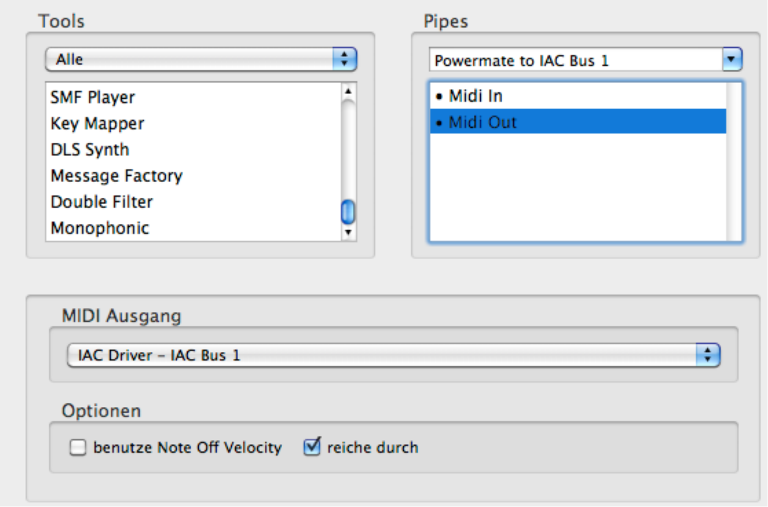
It basically boils down to this:
When turned the PowerMate will send an AppleScript that tells MidiPipe to send Midi commands to the Core Midi’s IAC bus. Logic listens to the IAC bus and you can use its Controller Assignment to map it to almost anything you want.

Configure MidiPipe

As we only need MidiPipe to pipe the data from PowerMate to the Midi bus the setup is quite simple. The chain has only too items:

Midi in: MidiPipe AppleAcript input

Midi out: IAC Driver – IAC Bus 1



Powermate to IAC Bus 1

MIDI Ausgang: IAC Driver - IAC Bus 1

☐ benutze Note Off Velocity ☑ reiche durch

Downlad the ready made pipe together withe the AppleScript examples. Simply start the pipe PowerMate2Midi.mipi when you want to use Midi from the PowerMate.

Configure PowerMate

In PowerMate create trigger pairs (Rotate Left/Rotate Right) each with an AppleScript as action. The script is straight forward:

```
tell application "MidiPipe"
  MIDISend toPort "MidiPipe AppleScript Input" withData {176, 80, 1}
end tell
```

The magic lies to the three figures in brackets:

> 176 means »Control Change Message Channel 1« – don’t change this unless you know what to do (see midi.org for details).
>
> 80 is the number of the controller. Any number from 1 to 127 is possible, many of them are defined by standard (1 for example would be the modulation wheel). I choose some of the »general purpose« controllers to stay away from interfering with common controllers.
>
> the last figure is the value to the send (again 1-127 would be possible)

For an endless controller like the PowerMate a range of values is not recommended. So I defined separate controller for left and right turn (one »+« and one »-« Button like on many keyboards). The AppleScript for both triggers is the same except for the controller number.

> Rotate left: 80 (like above)
>
> Rotate right: 81

I defined three of this trigger pairs in PowerMate 3 one without modifier to control the Volume-Fader, one with shift (using Midi controller 82/83) for panning and one with ctrl (using 84/85) to control the first effect send. (It is also possible to configure one of these to rotating the PowerMate while pressing it, but I find it hard to press and turn, so I used the modifier keys instead)

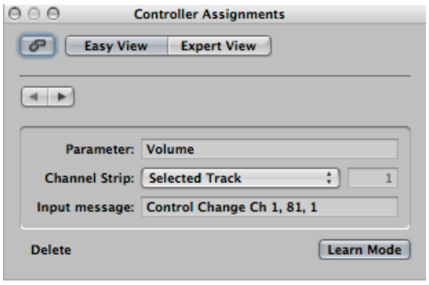
Controller Assignment in Logic

If everything is set correctly Logic should get some Midi input when turning the PowerMate. Now we only have to tell Logic which of the incoming Midi controls what.

> Rotate PowerMate => Controller 80/81 => Fader
>
> Shift + rotate PowerMate => Controller 82/83 => Pan
>
> Ctrl + rotate PowerMate => Controller 84/85 => Send 1

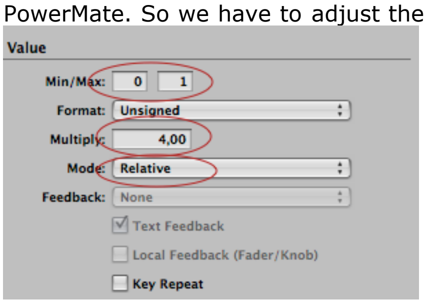
Let's set it up in Logic:

> Click on the Volume Fader and hit cmd-L (I hope this is the standard key command) or go to Preferences>Control Surfaces>Learn Assignment for »Volume«.
>
> Turn the PowerMate in one (!) direction. The midi data should be entered by Logic. And the »Channel Strip« Popup should read »Selected Track«:



Parameter: Volume
Channel Strip: Selected Track 1
Input message: Control Change Ch 1, 81, 1

Go to the Expert View. Most of the data in the right column should be fine (change the name and label if you want). Only the Value section at the bottom needs attention as Logic expects the Fader to be scaled from 1-127. This is fine for a real fader with a defined start and end but does not fit to the endless nature of the PowerMate. So we have to adjust the data a bit:



Min/Max: 0 1
Format: Unsigned
Multiply: 4,00
Mode: Relative
Feedback: None
☑ Text Feedback
☐ Local Feedback (Fader/Knob)
☐ Key Repeat

Min/Max: 0/1 (we never send 0 though)

Mode: Relative (we have no absolute values only increase/decrease)

Multiply: This needs a - (minus) if you want to decrease. You also can adjust the sensitivity of the movement here. I found 4 (or-4) pleasing.

Repeat the same procedure for the other direction (and don't forget the minus for the left turn).

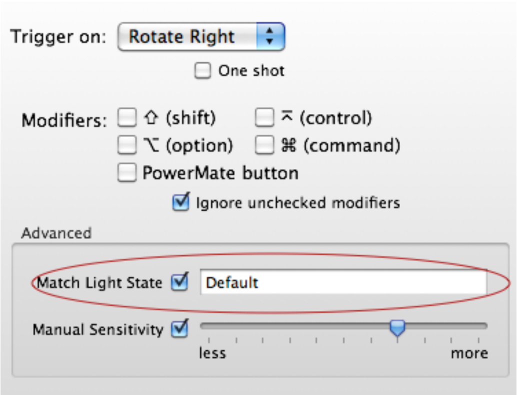
Repeat the steps for pan and effect send (and don't forget to press the modifier key defined in PowerMate)

Finished: I have the three most important controls of the selected track on a single knob. The PowerMate is sitting left of my keyboard. So my thumb can reach shift or ctrl while I turn the knob.

Enlightenment or getting it all together using »Light States«

Going through the setup steps of the three task you will have noticed that they can not be used at the same time. So we need to tell PowerMate what task is needed and find a way for easy switching. The key concept here is the new feature »Light State« introduced with PowerMate 3. You can tell each trigger to work only with a particular Light State. I use three Light States matching the three tasks:

1) »Default« (scrolling/start/stop)
2) »Scroll« (scroll wheel plugin parameter control)
3) »Fader« (control channel strip)



Trigger on: Rotate Right
☐ One shot
Modifiers: ☐ ⇧ (shift) ☐ ⌃ (control)
☐ ⌥ (option) ☐ ⌘ (command)
☐ PowerMate button
☑ Ignore unchecked modifiers
Match Light State ☑ Default
Manual Sensitivity ☑

So for each trigger I check the checkbox »Match Light State« and enter the appropriate name.

How to set and switch Light States

This is done via AppleScript. It looks like this:

```
tell application "PowerMate"
  set aDevice to device named "PowerMate"
    tell aDevice
      make light state with properties {state type:steady, brightness:0.05, pulse rate:0.1, name:"Default"}
    end tell
end tell
```

The important part is the name of the Light State. But I use the blue light to hint what state the PowerMate is in letting it shine or pulse differently.

To switch Light States I simply run the appropriate script. For fast switching I found two solutions:

> 1) Switching with the PowerMate using the long Press trigger and an AppleScript to toggle between two Light States. (The Toogle Script is in the download as well
>
> 2) Using FastScripts a nice Application that lets you assign shortcuts to Applescripts (the light version is free and allows up to 10 shortcuts).
> I trigger the three light state scripts with F13/F14/F15.

Download a zip with all AppleScript examples and the pipe for Midipipe.

That's it. I hope this tutorial was helpfull. Unleash your PowerMate!
