

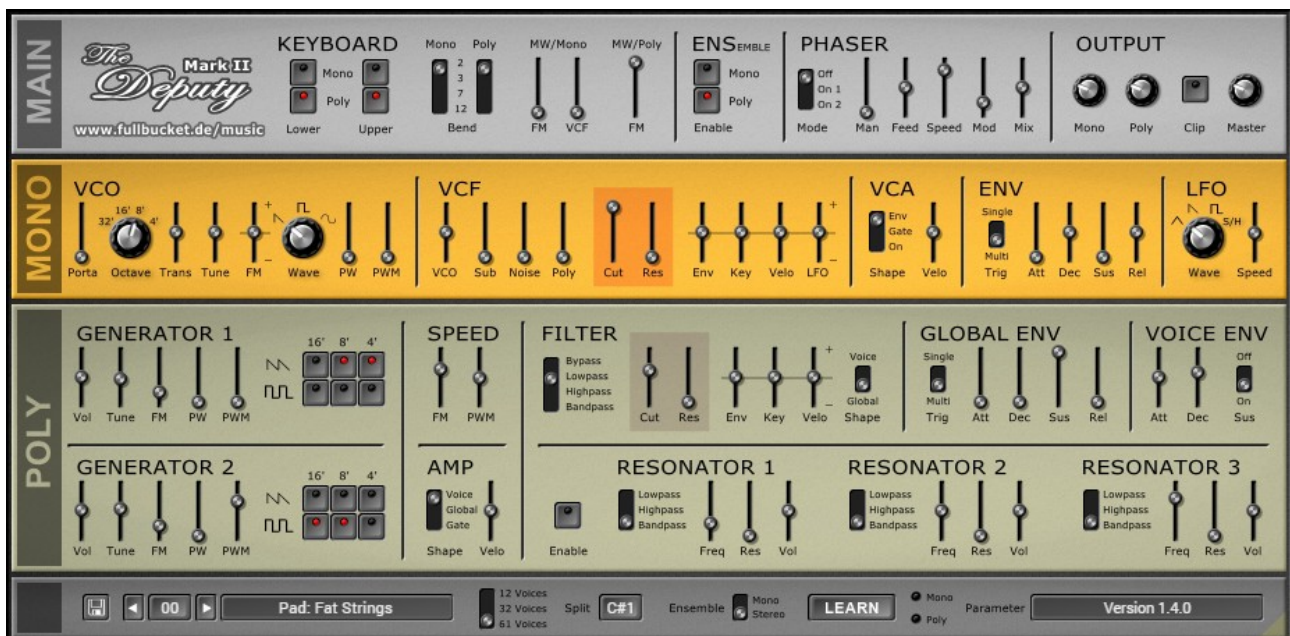
the Deputy

Mark II

Polyphonic Ensemble

Version 1.4

© 2012-2026 by Björn Arlt @ Full Bucket Music
<http://www.fullbucket.de/music>



VST is a trademark of Steinberg Media Technologies GmbH
Windows is a registered trademark of Microsoft Corporation
The Audio Units logo is a trademark of Apple Computer, Inc.
AAX is a trademarks of Avid Technology, Inc.

Table of Contents

| | |
|--|----|
| Introduction..... | 3 |
| Architecture..... | 4 |
| The <i>Poly</i> Section..... | 5 |
| Generator Banks..... | 5 |
| Filter..... | 5 |
| Envelopes..... | 6 |
| Amplifier..... | 6 |
| Resonators..... | 6 |
| The <i>Mono</i> Section..... | 7 |
| VCO..... | 7 |
| VCF..... | 7 |
| VCA..... | 7 |
| ENV and LFO..... | 7 |
| The <i>Main</i> Section..... | 8 |
| Keyboard..... | 8 |
| Ensemble..... | 8 |
| Phaser..... | 8 |
| Output..... | 8 |
| General Controls..... | 9 |
| MIDI Learn..... | 9 |
| Program and File Menu..... | 10 |
| Parameters..... | 11 |
| Main — Keyboard..... | 11 |
| Main — Ensemble..... | 11 |
| Main — Phaser..... | 11 |
| Main — Output..... | 12 |
| Mono — VCO..... | 12 |
| Mono — VCF..... | 12 |
| Mono — VCA..... | 13 |
| Mono — Env..... | 13 |
| Mono — LFO..... | 13 |
| Poly — Generator 1 & 2..... | 13 |
| Poly — Speed..... | 14 |
| Poly — Amp..... | 14 |
| Poly — Filter..... | 14 |
| Poly — Global Env..... | 14 |
| Poly — Voice Env..... | 15 |
| Poly — Resonators..... | 15 |
| General Controls..... | 15 |
| Hidden Controls (<i>Ensemble</i>)..... | 15 |
| Frequently Asked Questions..... | 16 |

Introduction

The *Deputy Mark II* is a software instrument plug-in for Microsoft Windows (CLAP/VST2/VST3/AAX) and Apple macOS (CLAP/VST2/VST3/AU/AAX) in the tradition of classic string machines and early “polyphonic” synthesizers, written in native C++ code. The main features are:

- Polyphonic string/synthesizer (*Poly*) section and monophonic synthesizer (*Mono*) section
- Built-in Ensemble and Phaser effects
- Flexible keyboard assignment for *Poly* and *Mono* section
- *Poly* section:
 - Up to 61 voices polyphony
 - Two band-limited frequency divider-driven oscillator banks
 - Sawtooth and PWM waveforms
 - 2-pole multi-mode filter and ADS envelope per voice
 - Global ADSR envelope for paraphonic modulation
 - Resonator bank (3 zero-delay feedback filters)
- *Mono* section:
 - Single VCO plus suboscillator and noise
 - 4-pole zero-delay feedback lowpass filter
 - ADSR envelope and LFO
 - Output of *Poly* section can be send to filter input
- All parameters can be controlled by MIDI controllers
- Plug-in supports Windows and macOS (32 bit and 64 bit)

The *Deputy* emulates an instrument that *might have been built* this way in the 1970s. Due to technical and economical reasons it would have been a compromise between which features a musician expected, what he was willing to pay for it, and what fitted in a single box. The Digital Revolution already lured around the corner and manifested itself in frequency dividers and sophisticated keyboard controllers, but *sound* was still shaped in the good old analog fashion (if you ignore some ingenious machines like the PPG *Wave Computer* for the moment). As a conclusion, the *Deputy* is (technically speaking, and as to nowadays standards) a fairly limited instrument. However, I strongly believe that technical limitation can be a source of great creativity. Prost!

The *Deputy* is based on the new **iPlug2** framework maintained by **Oli Larkin and the iPlug2 team**. Big thanks, guys!!! Without your work it would not have been possible to create a resizable *Deputy* user interface.

To resize the plug-in you just grab the yellow triangle at the bottom right of the window and drag it. You can save the current window size using the menu entry “Save Window Size” in the *Program and File Menu*.

If you have trouble with the standard version of the *Deputy*, please grab the (sound-wise identical) “N” version of the plug-in which is based on the original **iPlug** framework.

Architecture

The *Deputy* consists of three major blocks: the polyphonic *Poly* synthesizer section, the monophonic *Mono* synthesizer section, and the *Main* section hosting global effects, the mixer, and keyboard-related controls. Apart from that the *Deputy Mark II* has an additional section containing controls for program management and “tweak” parameters.

Each voice of the *Poly* section features two “generators” feeding a 2-pole multi-mode filter. The generators are driven by independent master oscillators and deliver sawtooth and/or pulse signals derived from frequency divider banks. Two simple LFOs are used for frequency and pulse width modulation. The amplitude and/or the filter cutoff can be controlled by an ADS envelope (per voice) or by an additional global ADSR envelope (for all voices). A resonator bank consisting of three 4-pole multi-mode filters is available to process the signal of the *Poly* section.

The *Mono* section features a single VCO plus suboscillator and white noise, a 4-pole lowpass VCF, an ADSR envelope, and a LFO offering four waveforms. The output of the *Poly* section can be sent to the VCF input as well.

Finally, in the *Main* section the *Poly* and *Mono* parts can be assigned to one or both of the two keyboard ranges while pitch and modulation wheels can be routed to several modulation targets. Furthermore, an Ensemble and a Phaser effect are available as well as an overall output mixer.

The *Poly* Section

The *Poly* section is the heart of the *Deputy Mark II*: A fully polyphonic synthesizer (up to 61 voices) plus a three-stage resonator bank. Its architecture resembles that of many well-known instruments like the *Polymoog* or the Korg *PS* series which also used electronic organ and classic string machine technology.

Generator Banks

The *Poly* synthesizer section follows a common, yet (at that time) highly discussed pattern of 1970s polyphonic keyboard designs: The tone signals are not generated directly by individual oscillators but by using overall “Master” oscillators driving a cascade of frequency dividers. While this design is rather cost-effective — in this case only one Master oscillator and a “Top-Octave Synthesizer” followed by octave dividers are needed to create a *fully polyphonic* generator bank — it also means that the waveform signals of different octaves of the same *key* (i.e. *C* to *B*) are totally correlated. Purists never stopped to point out that such an arrangement sounds “cold” and “unnatural”, many even simply rejected it because this technology was used in (and originally developed for) electronic organs. Some 40 years later purists will assure you that this is *absolutely necessary* to reproduce the “warm” tone of, for example, famous string machines like the Eminent *Solina* etc.

A nice feature of such generator banks is that one gets the upper and lower octaves of a played key (almost) “for free” — think of choirs in (again) electronic organs. The *Poly* section provides two different waveform outputs (sawtooth and pulse) and three such octave signals (16', 8', and 4') which are individually selectable per waveform. To mitigate the static character of the generated signal it is possible to modulate the width of the pulse waveform by a dedicated LFO. However, the “real synthesizer feeling” is introduced by a second, identical generator bank which can be (de)tuned with respect to the first bank; this at least produces the famous “beating” when signals undergo a periodical phase cancellation.

Finally, frequency modulation of the generator banks by a separate LFO (vibrato) is available as well.

Filter

In contrary to other early polyphonic synthesizers (e.g. the *Polymoog* but also the later Korg *Poly 800*), the *Poly* section offers a multi-mode filter (lowpass, highpass, bandpass) *per voice*. Thus, each voice (in fact each of the 61 keys!) has its own filter in place — an enormous amount of additional hardware! To reduce the costs, the filter layout follows a simple 2-pole design which is *not* capable of self-oscillation (today an absolute *No-Go*). The filters' cutoff frequency can be modulated by key tracking, velocity (an uncommon feature in the 1970s), and the *Voice* or *Global* envelopes (see below). Here, it should be mentioned that both positive and negative modulation is possible.

Envelopes

There is a dedicated envelope per voice (the *Voice* envelope) which can be used to modulate the amplifier and/or the filter. To save costs, this envelope has a reduced control set, namely *Attack* and *Decay* plus a switchable *Sustain* level (*On* or *Off*) where *Decay* also controls the release time.

An additional *Global* envelope featuring the full ADSR control set is available as well. However, there is only one such envelope for the whole *Poly* section resulting in a *paraphonic* playing mode. Thus, the trigger mode can be changed from *Single* (a new note only triggers the envelope when all previous notes have been released) to *Multiple* (the envelope is triggered for each new note).

Amplifier

The amplifier section offers modulation by velocity (!) and a selectable envelope (*Voice*, *Global*, and *Gate*, a simple rectangular-shaped envelope). Again, there is one amplifier per voice, a luxury and only possible due to progress in large scale circuit integration.

Resonators

The *Deputy* is equipped with three independent 4-pole multi-mode filters (lowpass, highpass, bandpass) working in parallel mode on the overall output of the *Poly* section. Although it is not possible to modulate their cutoff frequency, these *Resonators* become very handy when it comes to create fixed formants.

The *Mono* Section

Many keyboards of the 70s were in fact “assemblies” of multiple instruments, e.g. String, Brass, Synthesizer, and Organ sections combined in one box, sharing a common keyboard controller; examples are the ARP *Quadra*, the Moog *Opus*, but also the Siel *Trilogy*, to name a few. Along with the *Poly* section the *Deputy* “just” features an additional monophonic synthesizer, the *Mono* section. Following a very basic VCO/VCF/VCA design, there is nothing spectacular about this little buddy except that the output of the *Poly* section can be routed through the *Mono*'s VCF as well.

VCO

The VCO offers three waveforms (sawtooth, pulse, sine) in four octaves (32', 16, 8', 4'). The pulse width can be modulated by the LFO, and it is also possible to tune the VCO in a range of ± 7 semitones.

VCF

This 4-pole lowpass filter is capable of self-oscillation and takes the outputs of the VCO, the rectangular suboscillator (one octave below the VCO frequency), the white noise generator, and the *Poly* section as input signals. Cutoff frequency will be positively or negatively modulated by velocity, keyboard tracking, LFO, and the ADSR envelope.

VCA

To modulate the amplitude of the *Mono* synthesizer, one can select the ADSR or a simple gate envelope. Furthermore, it is possible to set the VCA into a permanent “On” mode; this becomes handy when the *Mono* section is configured to process only the output of the *Poly* section. Last but not least, the VCA can be modulated by velocity.

ENV and LFO

The ADSR envelope of the *Mono* section is identical to the *Global* envelope of the *Poly* section and is used as a modulation source for the VCF and the VCA.

The LFO features four waveforms (triangle, sawtooth, rectangle, sample/hold) which can modulate the VCO frequency and pulse width and/or the VCF cutoff.

The *Main* Section

This section provides common functionality for both the polyphonic and the monophonic synthesizer sections: keyboard controls, effects, and mixer.

Keyboard

The keyboard of the *Deputy* has 61 keys (C-1 to C-6), and each key makes up one individual voice (one can decrease the number of total voices in the plug-in version to reduce computational load). Thus, only MIDI keys C-1 to C-6 generate a sound when played.

The keyboard is divided into two *zones*, the *Lower* and the *Upper* zone — the split point between both regions was originally fixed to C-3 but is variable in the plug-in version). The *Poly* and the *Mono* sections can be assigned to one or both of these zones independently, allowing for split, layered, or combinations of both modes.

The amount of pitch wheel modulation (2, 3, 7 or 12 semitones) is selectable for both sections as well; this also holds for the modulation wheel which controls the intensity of vibrato and LFO respectively. Furthermore, the modulation wheel can be assigned to manipulate the VCF cutoff frequency.

Ensemble

Many classic string machines like the Eminent *Solina* are famous (or infamous) for their modulation effects (often called *Chorus*, *Ensemble*, or simply *Modulation*). While these effects are very important for the typical sound characteristics, the original rationale for integrating them into such an instrument was to beef up the rather static output of the single generator bank. In contrary, the *Polymoog* (like the *Deputy*) features two individual generator banks with a detuning option resulting in a “fat” sound. However (and unlike the *Polymoog*), the *Deputy* does have a built-in *Ensemble* effect that can be individually activated for both the *Poly* and the *Mono* section.

Phaser

The *Phaser* effect modulates the sum of both section outputs. Implemented by a classic 4-pole allpass filter the *Phaser* comes with two different modes and features feedback control plus a separate LFO for modulation.

Output

Last but not least, the *Output* section sums up the individual *Mono* and *Poly* output signals. There is also a very simple signal limiter (called *Clip*) which in the signal path lies before the master volume control. However, for “loud” signals this circuit works more like an overdrive effect.

General Controls

Some general controls live in a section at the very bottom of the *Deputy's* user interface. Besides the program selector there are also some “tweak” parameters: The number of *Poly* voices (12, 32, 61) to reduce computational load, the split point between the keyboard zones, and the *Ensemble* mode (mono or stereo). Furthermore, the MIDI Learn switch (see section *MIDI Learn*) is followed by two LEDs that report the status of the *Mono* and *Poly* sections (on or off).

A quite handy tool is the parameter display: At start-up it displays the software version while as soon as one hovers over a parameter the respective value is shown.

MIDI Learn

The easiest way to assign MIDI controllers to *Deputy* parameters is to use the MIDI Learn function. To activate *MIDI Learn*, click on the respective button and wiggle both the MIDI controller and the *Deputy's* parameter that you want to link. If you want to unlearn the assignment, right-click the *MIDI Learn* button (the label now reads “UNLEARN”) and activate it. Now wiggle the MIDI controller or the parameter that you want to unlearn.

Make sure to save the *Deputy's* configuration to persist your MIDI controller assignments (see section *Program and File Menu*).

Program and File Menu

When clicking on the *File* button (disk symbol on bottom left of the screen), a context menu opens with the following options:

| | |
|---------------------------------------|--|
| Copy Program | Copy current program to internal clipboard |
| Paste Program | Paste internal clipboard to current program |
| Load Program | Load a program file containing a patch to the <i>Deputy's</i> current program |
| Save Program | Save the <i>Deputy's</i> current program to a program file |
| Load Bank | Load a bank file containing 64 patches into the <i>Deputy</i> |
| Save Bank | Save the <i>Deputy's</i> 64 patches to a bank file |
| Init Program | Initialize the current program |
| Reload Configuration | Reload the <i>Deputy's</i> configuration (see section <i>MIDI Learn</i>) |
| Save Configuration | Save the <i>Deputy's</i> configuration (see section <i>MIDI Learn</i>) |
| Select Startup Bank | Select the bank file that should always be loaded when the <i>Deputy</i> is started |
| Load Startup Bank | Load the Startup bank file; can also be used to check what the current Startup bank is |
| Unselect Startup Bank | Unselect the current Startup bank |
| Default Path for Program Files | Sets the default path for program and bank files |
| MIDI Thru | Set globally if MIDI data sent to <i>Deputy</i> should be sent through to its MIDI output (stored in configuration file) |
| Ignore Program Change | Set globally if MIDI Program Change data sent to <i>Deputy</i> should be ignored (stored in configuration file) |
| Reload Configuration | Reload <i>Deputy's</i> configuration file |
| Save Configuration | Save <i>Deputy's</i> configuration file |
| Window Size... | Change the window size of <i>Deputy</i> |
| Save Window Size | Stores the current window size to the configuration file so that it will be restored on the next loading of <i>Deputy</i> |
| Switch View | Switch between alternative built-in skins |
| Check Online for Update | When connected to the Internet, this function will check if a newer version of the <i>Deputy</i> is available at fullbucket.de |
| Visit fullbucket.de | Open fullbucket.de in your standard browser |

Parameters

Main — Keyboard

| parameter | ID | description |
|--------------------|----|--|
| <i>Lower/Mono</i> | 4 | activates <i>Mono</i> section for <i>Lower</i> keyboard zone |
| <i>Lower/Poly</i> | 3 | activates <i>Poly</i> section for <i>Lower</i> keyboard zone |
| <i>Upper/Mono</i> | 6 | activates <i>Mono</i> section for <i>Upper</i> keyboard zone |
| <i>Upper/Poly</i> | 5 | activates <i>Mono</i> section for <i>Upper</i> keyboard zone |
| <i>Bend/Mono</i> | 8 | pitch bend amount ($\pm 2, 3, 7, 12$ semitones) for <i>Mono</i> section |
| <i>Bend/Poly</i> | 7 | pitch bend amount ($\pm 2, 3, 7, 12$ semitones) for <i>Poly</i> section |
| <i>MW/Mono FM</i> | 10 | mod wheel amount for frequency modulation of <i>Mono</i> section |
| <i>MW/Mono VCF</i> | 11 | mod wheel amount for cutoff frequency of VCF (<i>Mono</i> section) |
| <i>MW/Poly FM</i> | 9 | mod wheel amount for frequency modulation of <i>Poly</i> section |

Main — Ensemble

| parameter | ID | description |
|--------------------|----|---|
| <i>Enable/Mono</i> | 97 | activates <i>Ensemble</i> effect on <i>Mono</i> section |
| <i>Enable/Poly</i> | 96 | activates <i>Ensemble</i> effect on <i>Poly</i> section |

Main — Phaser

| parameter | ID | description |
|--------------|-----|--|
| <i>Mode</i> | 98 | <i>Phaser</i> effect mode (<i>Off</i> , <i>On – Mode 1</i> , <i>On – Mode 2</i>) |
| <i>Man</i> | 99 | manual setting (i.e. center frequency) |
| <i>Feed</i> | 100 | feedback |
| <i>Speed</i> | 101 | modulation speed |
| <i>Mod</i> | 102 | modulation amount |
| <i>Mix</i> | 103 | dry/wet balance |

Main — Output

| parameter | ID | description |
|---------------|----|---|
| <i>Mono</i> | 86 | volume of the <i>Mono</i> section |
| <i>Poly</i> | 46 | volume of the <i>Poly</i> section |
| <i>Clip</i> | 1 | controls if the overall output is unlimited or clipped to unity |
| <i>Master</i> | 0 | overall volume |

Mono — VCO

| parameter | ID | description |
|---------------|----|---|
| <i>Porta</i> | 68 | portamento time (0 to 5 seconds) |
| <i>Octave</i> | 69 | octave (32', 16', 8', 4') |
| <i>Trans</i> | 70 | transpose (0 to ± 7 semitones) |
| <i>Tune</i> | 71 | fine tuning (± 1 semitone) |
| <i>FM</i> | 75 | amount of frequency modulation by LFO |
| <i>Wave</i> | 72 | waveform (sawtooth, pulse, sine) |
| <i>PW</i> | 73 | pulse width |
| <i>PWM</i> | 74 | amount of pulse width modulation by LFO |

Mono — VCF

| parameter | ID | description |
|--------------|----|--|
| <i>VCO</i> | 76 | volume of VCO signal |
| <i>Sub</i> | 77 | volume of suboscillator signal (rectangle, -1 octave) |
| <i>Noise</i> | 78 | volume of white noise signal |
| <i>Poly</i> | 79 | volume of <i>Poly</i> section output signal |
| <i>Cut</i> | 80 | cutoff frequency |
| <i>Res</i> | 81 | resonance (up to self-oscillation) |
| <i>Env</i> | 82 | amount of cutoff frequency modulation by envelope |
| <i>Key</i> | 83 | amount of cutoff frequency modulation by keyboard tracking |
| <i>Velo</i> | 84 | amount of cutoff frequency modulation by velocity |
| <i>LFO</i> | 85 | amount of cutoff frequency modulation by LFO |

Mono — VCA

| parameter | ID | description |
|--------------|----|---|
| <i>Shape</i> | 88 | envelope selection (<i>Env, Gate, On</i>) |
| <i>Velo</i> | 87 | amount of amplitude modulation by velocity |

Mono — Env

| parameter | ID | description |
|----------------|----|--|
| <i>Trig</i> | 89 | envelope trigger selection (<i>Single, Multiple</i>) |
| <i>Attack</i> | 90 | attack time |
| <i>Decay</i> | 91 | decay time |
| <i>Sustain</i> | 92 | sustain level |
| <i>Release</i> | 93 | release time |

Mono — LFO

| parameter | ID | description |
|--------------|----|--|
| <i>Wave</i> | 94 | LFO waveform (<i>triangle, sawtooth, rectangle, Sample & Hold</i>) |
| <i>Speed</i> | 95 | LFO speed (0 to 30 Hz) |

Poly — Generator 1 & 2

| parameter | ID | description |
|------------------|--------|--|
| <i>Vol</i> | 15; 26 | volume of the generator bank |
| <i>Tune</i> | 16; 27 | fine tuning (± 1 semitone) |
| <i>FM</i> | 19; 30 | amount of frequency modulation |
| <i>PW</i> | 17; 28 | pulse width |
| <i>PWM</i> | 18; 29 | amount of pulse width modulation |
| <i>Saw 16'</i> | 20; 31 | activates the 16' <i>Sawtooth</i> register |
| <i>Saw 8'</i> | 21; 32 | activates the 8' <i>Sawtooth</i> register |
| <i>Saw 4'</i> | 22; 33 | activates the 4' <i>Sawtooth</i> register |
| <i>Pulse 16'</i> | 23; 34 | activates the 16' <i>Pulse</i> register |
| <i>Pulse 8'</i> | 24; 35 | activates the 8' <i>Pulse</i> register |
| <i>Pulse 4'</i> | 25; 36 | activates the 4' <i>Pulse</i> register |

Poly — Speed

| parameter | ID | description |
|------------|----|--|
| <i>FM</i> | 13 | speed of frequency modulation (0 to 30 Hz) |
| <i>PWM</i> | 14 | speed of pulse width modulation (0 to 30 Hz) |

Poly — Amp

| parameter | ID | description |
|--------------|----|---|
| <i>Shape</i> | 44 | envelope selection (<i>Voice, Global, Gate</i>) |
| <i>Velo</i> | 45 | amount of amplitude modulation by velocity |

Poly — Filter

| parameter | ID | description |
|--------------|----|--|
| <i>Mode</i> | 37 | filter mode (bypass, lowpass, highpass, bandpass) |
| <i>Cut</i> | 38 | cutoff frequency |
| <i>Res</i> | 39 | resonance |
| <i>Env</i> | 41 | amount of cutoff frequency modulation by envelope |
| <i>Key</i> | 41 | amount of cutoff frequency modulation by keyboard tracking |
| <i>Velo</i> | 43 | amount of cutoff frequency modulation by velocity |
| <i>Shape</i> | 40 | envelope selection (<i>Voice, Global</i>) |

Poly — Global Env

| parameter | ID | description |
|----------------|----|--|
| <i>Trig</i> | 63 | envelope trigger selection (<i>Single, Multiple</i>) |
| <i>Attack</i> | 64 | attack time |
| <i>Decay</i> | 65 | decay time |
| <i>Sustain</i> | 66 | sustain level |
| <i>Release</i> | 67 | release time |

Poly — Voice Env

| parameter | ID | description |
|----------------|----|---------------------------|
| <i>Attack</i> | 60 | attack time |
| <i>Decay</i> | 61 | decay/release time |
| <i>Sustain</i> | 62 | sustain level (on or off) |

Poly — Resonators

| parameter | ID | description |
|-----------------|----------|---|
| <i>Enable</i> | 47 | activates the resonators |
| <i>Mode 1-3</i> | 48;52;56 | filter mode (lowpass, highpass, bandpass) |
| <i>Cut 1-3</i> | 50;54;58 | cutoff frequency |
| <i>Res 1-3</i> | 51;55;59 | resonance |
| <i>Vol 1-3</i> | 49;53;57 | volume |

General Controls

| parameter | ID | description |
|-----------------|-----|--|
| <i>Program</i> | — | program selector (programs 1 – 64) |
| <i>Voices</i> | 12 | number of <i>Poly</i> voices (12, 32, 61) |
| <i>Split</i> | 2 | split point between <i>Lower</i> and <i>Upper</i> zone |
| <i>Ensemble</i> | 104 | <i>Ensemble</i> mode (mono or stereo) |
| <i>Knob</i> | — | knob mode (linear or circular) |

Hidden Controls (*Ensemble*)

| parameter | ID | description |
|----------------------|-----|--|
| <i>LFO1 Speed</i> | 105 | Speed of <i>Ensemble</i> LFO 1 |
| <i>LFO2 Speed</i> | 106 | Speed of <i>Ensemble</i> LFO 2 |
| <i>Mod. Amount 1</i> | 107 | Modulation amount <i>Ensemble</i> for delay line 1 |
| <i>Mod. Amount 2</i> | 108 | Modulation amount <i>Ensemble</i> for delay line 2 |
| <i>Mod. Amount 3</i> | 109 | Modulation amount <i>Ensemble</i> for delay line 3 |

Frequently Asked Questions

How do I install Deputy (Windows VST2 32 bit version)?

Just copy the files `deputy.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST2 plug-in folder. Your DAW should automatically register the *Deputy* VST2 plug-in the next time you start it.

How do I install Deputy (Windows VST2 64 bit version)?

Just copy the file `deputy64.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST2 plug-in folder. Your DAW should automatically register the *Deputy* VST2 plug-in the next time you start it.

Note: You may have to remove any existing (32 bit) `deputy.dll` from your VST2 plug-in folder or else your DAW may screw the versions up...

How do I install Deputy (Windows VST3 64 bit version)?

Just copy the files `deputy.vst3` from the ZIP archive you have downloaded to your system's or favorite DAW's VST3 plug-in folder. Your DAW should automatically register the *Deputy* VST3 plug-in the next time you start it.

How do I install the Deputy (Windows AAX 64 bit version)?

Copy the file `deputy_AAX_installer.exe` from the ZIP archive you have downloaded to any of your system's folder and run it. Your AAX-enabled DAW (Pro Tools etc.) should automatically register the *Deputy* AAX plug-in the next time you start it.

How do I install Deputy (Mac)?

Locate the downloaded PKG package file in Finder (!) and do a right- or control-click on it. In the context menu, click on "Open". You will be asked if you really want to install the package because it comes from an "unidentified developer" (me ☺). Click "OK" and follow the installation instructions.

What is the plug-in ID of the Deputy?

The ID is **T D M 2** .

How can I decrease the Deputy's CPU load?

Whenever it does not degrade the *sound* you need, try this:

- Reduce polyphony, i.e. the number of voices.
- Deactivate a section, an effect, or the resonators.
- If you don't need a waveform/octave to be generated by one or both generator bank, switch off the respective buttons.

How can I prevent output clipping?

Decrease the sections' *Volume* levels. Alternatively, switch on the *Clip* parameter in the *Main* section but that might lead to distortion effects. Hey, maybe that's exactly what you need? ;-)

Can I tweak the Ensemble effect parameters?

Yes, but not from the *Deputy*'s front panel. Like on the old *Solina* thy shall seek and thy will be given.

How do I know if a new version of the Deputy is available?

When connected to the Internet, open the *Program and File Menu* by clicking the disk icon and select the entry "Check Online for Updates". If a new version of the *Deputy Mark II* is available on fullbucket.de the respective information will be shown in a message box.

Seriously, the Deputy has never existed as a real "hardware" instrument...!?

I never said it would! The *Deputy Mark II* is an emulation of a *fictive* polyphonic synthesizer from the 1970s of an alternative history. But the point is that an instrument like this *could have been built* just this way — see the appendix.

So why did you create the Deputy?

Because I love those weird machines with their strange concepts and technical designs. They tell us where the sophisticated instruments of today originally come from blablabla. And because it's fun!

What does "Prost!" mean?

You do not know this famous German word? It means something like "To your health!". Typically, German words tend to be indefinitely longer than their English counterparts but in this case the Germans came to the point.