µPlaits SE

Eurorack Module

TD–UPSE–B

Designed by Daniel & Michael Gilbert of *Tall Dog Electronics* in Western Massachusetts
tall-dog.com | **uplaits-se.com**

Based on the *Plaits* module designed by Émilie Gillet of *Mutable Instruments*
mutable-instruments.net

Partially manufactured by *Worthington Assembly* via *CircuitHub* in South Deerfield, MA
worthingtonassembly.com
circuithub.com

Partially manufactured by *PCBWay* in Hangzhou, China
pcbway.com

Assembled at *Rust Temple* in Easthampton, MA
rusttemple.today

Licensed under *Creative Commons Attribution-ShareAlike 4.0 International*
creativecommons.org/licenses/by-sa/4.0

Source materials are available on *GitHub*
github.com/loglow/uPlaits_SE

Contents

About

Plaits is a voltage-controlled sound source offering direct access to a large palette of easily tweakable raw sonic material generated by models spanning a wide range of synthesis techniques.

This redesigned *μPlaits SE* (Special Edition) is a new revision of *Plaits* that miniaturizes the module from 12HP to 8HP and redesigns its front panel layout.

μPlaits SE remains fully compatible with future upstream *Plaits* firmware upgrades and any other alternate firmwares that are compliant with the standard *Plaits* design.

Included Parts

μPlaits SE includes a standard 16-pin to 10-pin Eurorack power cable and two sets of mounting screws for racks/ enclosures with either M2.5 or M3 threads.

The **TD-UPSE-B/B** variant has a matte black front panel with gold markings made from FR-4 (glass-reinforced epoxy laminate) material. The included mounting screws are black.

The **TD-UPSE-B/S** variant has a satin silver front panel with black markings made from anodized aluminum material via the Metalphoto process. The included mounting screws are silver.

History

The original *Plaits*[1] was designed by Émilie Gillet of *Mutable Instruments*[2] and released under the free, open-source CC BY-SA 3.0[3] license.[4] *Plaits* is available as a 12HP Eurorack module.

Plaits is the spiritual successor of *Mutable Instruments'* best-selling voltage-controlled sound source, *Braids*.[5] Not just a mkII version: its hardware and software have been redesigned from scratch. Gone are the screen, menu system, hidden settings, and the long list of somewhat redundant synthesis models.

Thanks to additional CV inputs, and to the use of three timbre-shaping parameters per model, *Plaits* is straightforward to use, and much closer to the ideal of one synthesis technique = one model. What were fragmented islands of sound in *Braids* are now part of a continuum of sounds.

---

1 — **mutable-instruments.net/modules/plaits**
2 — **mutable-instruments.net/about**
3 — **creativecommons.org/licenses/by-sa/3.0**
4 — **github.com/pichenettes/eurorack**
5 — **mutable-instruments.net/modules/braids**

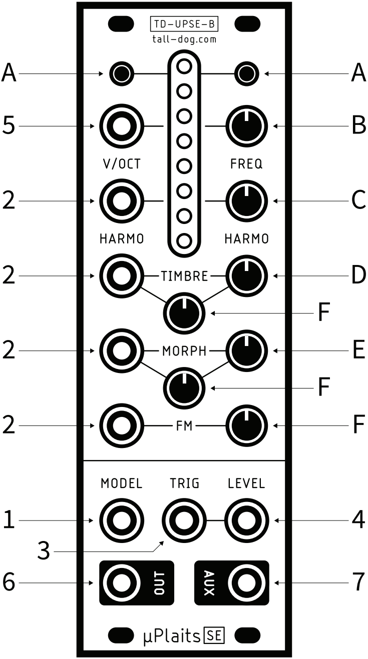
Warranty

This product is covered under warranty for one year following the date of purchase as indicated on the original sales receipt. This warranty covers any defect in the manufacturing of this product. This warranty does not cover any damage or malfunction caused by incorrect use—such as, but not limited to, power cables connected backwards, excessive voltage levels, exposure to extreme temperature or moisture levels, physical damage due to impact, or any aftermarket physical or electrical modifications or repairs.

This warranty covers either repair or replacement, at our sole discretion. This repair or replacement is subject to verification of the defect or malfunction and proof of purchase as confirmed by showing the model number on an original dated sales receipt. Shipping and handling fees are to be paid for by the customer.

Please contact **support@tall-dog.com** for a return authorization prior to shipping anything to us. Please understand that we will not be able to service units under warranty that have been modified or previously repaired by the customer or a third party.

Front Panel



TD-UPSE-B
tall-dog.com
A
A
5
B
V/OCT
FREQ
2
C
HARMO
HARMO
2
TIMBRE
D
F
2
MORPH
E
F
2
FM
F
MODEL
TRIG
LEVEL
1
4
3
6
OUT
AUX
7
µPlaits SE

Controls

Note: The lettering and numbering scheme used in this manual has been kept consistent with the upsteam documentation for the original *Plaits* module.

[**A**] — **Model selection buttons and LEDs** displaying the active model. Each button cycles through a bank of 8 models. The second bank is focused on noisy and percussive sounds.

[**B**] — **Coarse frequency control**. By default, it covers a range of 8 octaves, but it can be narrowed down to 14 semitones (refer to **FREQ** knob range, page 10).

[**C**] [**D**] [**E**] — **Model-dependent tone controls**. Their actual function varies from model to model. In general, **TIMBRE** [**D**] sweeps the spectral content from dark/sparse to bright/dense, **MORPH** [**E**] explores lateral timbral variations and **HARMO** [**C**] (short for harmonics) controls the frequency spread or the balance between the various constituents of the tone.

[**F**] — **Attenuverters** for the **TIMBRE**, **FM** and **MORPH** CV inputs. When the corresponding CV input is left unpatched and the **TRIG** (trigger) input [**3**] is patched, the attenuverter adjusts the modulation amount from the internal decaying envelope generator. When unplugging a CV input, and if the trigger input is patched, remember to reset the attenuverter to 12 o’clock if you do not want the internal envelope to take over!

Inputs & Outputs

[**1**] — **Model selection CV input**. When this CV input is modulated, two LEDs are lit: the steadily lit LED indicates the current model, and the slowly blinking LED indicates the central value, which would be obtained with a CV of zero volts and which is still modifiable with the buttons [**A**].

Note that when the **TRIG** input [**3**] is patched, model changes occur only whenever a trigger is received.

[**2**] — **CV inputs** for the **TIMBRE**, **FM**, **MORPH** and **HARMO** parameters.

[**3**] — **Trigger input**. Serves four percussive purposes:

- Triggers the internal decaying envelope generator.
- Excites the physical and percussive models.
- Strikes the internal low-pass gate (unless the **LEVEL** CV input [**4**] is patched).
- Samples and holds the value of the **MODEL** CV input [**1**].

Inputs & Outputs *Cont'd*

[**4**] — **Level CV input**. Opens the internal low-pass gate, to simultaneously control the amplitude and brightness of the output signal. Also acts as an accent control when triggering the physical or percussive models.

[**5**] — **V/Oct CV input**. Controls the fundamental frequency of the sound, from -3 to +7 octaves relative to the root note set by the coarse **FREQ** knob [**B**].

[**6**] [**7**] — **Outputs**. The **AUX** output [**7**] carries a variant, sidekick, or by-product of the main signal produced on **OUT** [**6**].

Adjusting the internal LPG & envelope

Hold the **LEFT** button [**A**] and:

- Turn the **TIMBRE** knob [**D**] to adjust the response of the LPG (low pass gate) from VCFA to VCA.
- Turn the **MORPH** knob [**E**] to adjust the ringing time of the LPG and the decay time of the internal envelope.

The values of both settings are represented by four yellow LEDs.

Adjusting the FREQ knob range

Hold the **RIGHT** button [**A**] and turn the **HARMO** knob [**C**] to adjust the range of the **FREQ** knob. The first 8 settings correspond to **C0 ± 7** semitones, **C1 ± 7** semitones, and so on. The last setting, with all LEDs lit, corresponds to the **full 8-octave range** from C0 to C8.

After having turned the **HARMO**, **TIMBRE** or **MORPH** knob to adjust a setting, the position of the knob might no longer match the original value of the corresponding parameter. When this happens, the response curve of the knob is modified to account for this discrepancy, until the position of the knob and the parameter perfectly line up again!

Synthesis Models

These 8 models are selected by the **LEFT** button [**A**] and are represented by a green LED indicator:

- **[L1]** **Pair of classic waveforms**
- **[L2]** **Waveshaping oscillator**
- **[L3]** **Two operator FM**
- **[L4]** **Granular formant oscillator**
- **[L5]** **Harmonic oscillator**
- **[L6]** **Wavetable oscillator**
- **[L7]** **Chords**
- **[L8]** **Vowel and speech synthesis**

These 8 models are selected by the **RIGHT** button [**A**] and are represented by a red LED indicator:

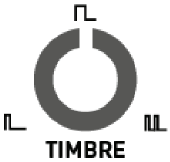
- **[R1]** **Granular cloud**
- **[R2]** **Filtered noise**
- **[R3]** **Particle noise**
- **[R4]** **Inharmonic string modeling**
- **[R5]** **Modal resonator**
- **[R6]** **Analog bass drum model**
- **[R7]** **Analog snare drum model**
- **[R8]** **Analog hi-hat model**

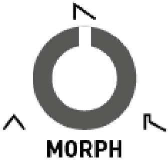
[**L1**] Pair of classic waveforms

Virtual-analog synthesis of classic waveforms.

- **HARMO** — detuning between the two waves.
- **TIMBRE** — variable square, from narrow pulse to full square to hardsync formants.
- **MORPH** — variable saw, from triangle to saw with an increasingly wide notch (*Braids'* CSAW).[1]
- **AUX** — sum of two hardsync'ed waveforms, the shape of which is controlled by **MORPH** and detuning by **HARMO**.

A **narrow pulse** or **wide notch** results in **silence**! Use this trick if you want to silence one of the two oscillators, to get a variable square or variable saw.





1 — **mutable-instruments.net/modules/braids**

[**L2**] Waveshaping oscillator

An asymmetric triangle processed by a waveshaper and a wavefolder. Sounds familiar? That's the same signal processing chain as in *Tides*, when it runs at audio rate![1]

- **HARMO** — waveshaper waveform.
- **TIMBRE** — wavefolder amount.
- **MORPH** — waveform asymmetry.
- **AUX** — variant employing another wavefolder curve, as available in *Warps*.[2]

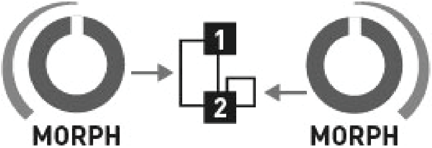
1 — **mutable-instruments.net/modules/tides**
2 — **mutable-instruments.net/modules/warps**

[**L3**] Two operator FM

Two sine-wave oscillators modulating each other's phase.

- **HARMO** — frequency ratio.
- **TIMBRE** — modulation index.
- **MORPH** — feedback, in the form of operator 2 modulating its own phase (past 12 o'clock, rough!) or operator 1's phase (before 12 o'clock, chaotic!).
- **AUX** — sub-oscillator.

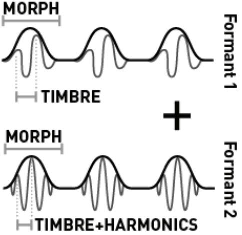
Note: turn **MORPH** fully CCW to get the same range of sounds as *Braids'* WTFM. Turn **MORPH** fully CW to recreate the same sounds as *Braids'* FBFM. A gentler palette equivalent to *Braids'* FM__ is found with **MORPH** at 12 o'clock.



[**L4**] Granular formant oscillator

Simulation of formants and filtered waveforms through the multiplication, addition and synchronization of segments of sine waves.

- **HARMO** — frequency ratio between formant 1 and 2.
- **TIMBRE** — formant frequency.
- **MORPH** — formant width and shape. This controls the shape of the window by which a sum of two synchronized sine oscillators is multiplied.
- **AUX** — simulation of filtered waveforms by windowed sine waves — a recreation of *Braids'* Z*** models. **HARMO** controls the filter type (peaking, LP, BP, HP), with smooth variation from one response to another.

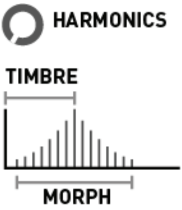


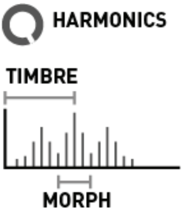
Synthesis Models *Cont'd*

[**L5**] Harmonic oscillator

An additive mixture of harmonically-related sine waves.

- **HARMO** — number of bumps in the spectrum. Starts with one big bump, and progressively adds ripples around it.
- **TIMBRE** — index of the most prominent harmonic. This control is somewhat similar to the cutoff frequency of a band-pass filter.
- **MORPH** — bump shape — from flat and wide to peaked and narrow. This control is somewhat similar to the resonance of a band-pass filter.
- **AUX** — variant including only the subset of harmonics present in the drawbars of a *Hammond* organ (frequency ratios of 1, 2, 3, 4, 6, 8, 10 and 12).



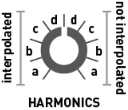


Synthesis Models *Cont'd*

[**L6**] Wavetable oscillator

Four banks of 8×8 waveforms, accessed by row and column, with or without interpolation.

- **HARMO** — bank selection. 4 interpolated banks followed by the same 4 banks, in reverse order, without interpolation.
    - **Bank A** — harmonically poor waveforms obtained by additive synthesis (sine harmonics, drawbar organ waveforms).
    - **Bank B** — harmonically rich waveforms obtained by formant synthesis or waveshaping.
    - **Bank C** — wavetables from the *Shruthi-1* / *Ambika*, sampled from classic wavetable or ROM playback synths.[1]
    - **Bank D** — a joyous semi-random permutation of waveforms from the other 3 banks.
- **TIMBRE** — row index. Within a row, the waves are sorted by spectral brightness (except for bank D which is a mess!).
- **MORPH** — column index.
- **AUX** — low-fi (5-bit) output.

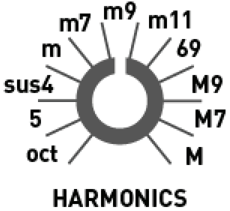


1 — **mutable-instruments.net/archive**

[**L7**] Chords

Four-note chords, played by virtual analogue or wavetable oscillators. The virtual analogue oscillators emulate the stack of harmonically-related square or sawtooth waveforms generated by vintage string & organ machines.

- **HARMO** — chord type.
- **TIMBRE** — chord inversion and transposition.
- **MORPH** — waveform. The first half of the knob goes through a selection of string-machine like raw waveforms (different combinations of the organ and string "drawbars"), the second half of the knob scans a small wavetable containing 16 waveforms.
- **AUX** — root note of the chord.



Synthesis Models *Cont'd*

[**L8**] Vowel and speech synthesis

A collection of speech synthesis algorithms.

- **HARMO** — crossfades between formant filtering, SAM (Software Automatic Mouth), and LPC (linear predictive coding) vowels, then goes through several banks of LPC words.
- **TIMBRE** — species selection, from Daleks to chipmunks. How does it work? This parameter either shifts the formants up or down independently of the pitch; or underclocks/overclocks the emulated LPC chip (with appropriate compensation to keep the pitch unchanged).
- **MORPH** — phoneme or word segment selection. When **HARMO** is past 11 o'clock, a list of words can be scanned through by turning the **MORPH** knob or by sending a CV to the corresponding input. One can also patch the **TRIG** input [**3**] to trigger the utterance of a word, use the **FM** attenuverter to control the intonation and the **MORPH** attenuverter to control speed.
- **AUX** — unfiltered vocal cords' signal.

Synthesis Models *Cont'd*

[**R1**] Granular cloud

A swarm of 8 enveloped sawtooth waves.

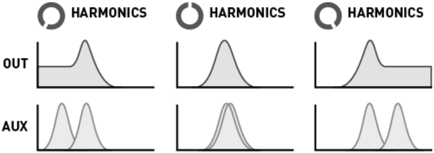
- **HARMO** — amount of pitch randomization.
- **TIMBRE** — grain density.
- **MORPH** — grain duration and overlap. When this setting is fully CW, the grains merge into each other: the result is a stack of eight randomly frequency-modulated waveforms.
- **AUX** — variant with sine wave oscillators.

To get a nice "supersaw" waveform, try a moderate amount of pitch randomization and grain density, with full grain overlap.

[**R2**] Filtered noise

Variable-clock white noise processed by a resonant filter. The cutoff frequency of the filter is controlled by the **FREQ** knob [**B**] and the **V/OCT** CV input [**5**]. This allows proper tracking!

- **HARMO** — filter response, from LP to BP to HP.
- **TIMBRE** — clock frequency.
- **MORPH** — filter resonance.
- **AUX** — variant employing two band-pass filters, with their separation controlled by **HARMO**.



[**R3**] Particle noise

Dust noise processed by networks of all-pass or band-pass filters.

- **HARMO** — amount of frequency randomization.
- **TIMBRE** — particle density.
- **MORPH** — filter type — reverberating all-pass network before 12 o'clock, then increasingly resonant band-pass filters.
- **AUX** — raw dust noise.

[**R4**] Inharmonic string modeling

[**R5**] Modal resonator

For your own pleasure, a mini-*Rings*![1] Please refer to the *Rings* manual for more information about modulated/ inharmonic string synthesis, and modal resonators.

When the **TRIG** input [**3**] is not patched, the string/ resonator is excited by dust (particle) noise. Otherwise, the string is excited by a short burst of filtered white noise, or by a low-pass filtered click.

- **HARMO** — amount of inharmonicity, or material selection.
- **TIMBRE** — excitation brightness and dust density.
- **MORPH** — decay time (energy absorption).
- **AUX** — raw exciter signal.

Note that *Plaits* uses a less powerful processor than *Rings*, and is thus limited to 3 voices of polyphony in inharmonic string modeling mode, and 1 voice of polyphony with 24 partials in modal resonator mode. *Plaits* does not allow you to control the position of the excitation, which is set to 25% of the length of the string/bar/tube.

---

1 — **mutable-instruments.net/modules/rings**

[**R6**] Analog bass drum model

No fancy acronyms or patented technology here, just behavioral simulation of circuits from classic drum machines! The drum machine emulated on **OUT** employs a bridged T-network excited by a nicely shaped pulse. As for the signal synthesized on **AUX**, it uses a frequency-modulated triangle VCO, turned into a sine with a pair of diodes, and shaped by a dirty VCA.

- **HARMO** — attack sharpness and amount of overdrive.
- **TIMBRE** — brightness.
- **MORPH** — decay time.

Without any signal patched to the **TRIG** input [**3**], a continuous tone is produced. Not particularly useful, but its amplitude can still be modulated by the **MORPH** knob and CV input!

[**R7**] Analog snare drum model

The drum machine emulated on **OUT** employs a bunch of bridged T-networks, one for each mode of the shell, excited by a nicely shaped pulse; plus some band-pass filtered noise. As for the signal synthesized on **AUX**, it is based on a pair of frequency-modulated sine VCOs, mixed with high-pass filtered noise.

- **HARMO** — balance of the harmonic and noisy components.
- **TIMBRE** — balance between the different modes of the drum.
- **MORPH** — decay time.

Synthesis Models *Cont’d*

[**R8**] Analog hi-hat model

The recipe is similar for both **OUT** and **AUX**: a bunch of square oscillators generate a harsh, metallic tone. The resulting signal is mixed with clocked noise, sent to a HPF, then to a VCA. While **OUT** uses 6 square oscillators and a dirty transistor VCA, **AUX** uses three pairs of square oscillators ring-modulating each other, and a clean, linear VCA.

- **HARMO** — balance of the metallic and filtered noise.
- **TIMBRE** — high-pass filter cutoff.
- **MORPH** — decay time.

Notes for [**R4**] [**R5**] [**R6**] [**R7**] [**R8**]

The physical and drum models employ their own decay envelope and filter. The internal LPG is disabled for them:

- The **TRIG** input [**3**] triggers the synthesis of the signal, but doesn’t strike the LPG.
- When the **TRIG** input is patched, the **LEVEL** input [**4**] works as an accent control.

Calibration

The module is factory-calibrated using precision voltage sources. Follow this procedure only if you want to compensate for inaccuracies in your CV sources, or if your module has lost its calibration settings following a fault or the installation of alternative firmware.

To calibrate the unit:

1. Disconnect all CV inputs.
2. Connect the note CV output of a well-calibrated keyboard interface or MIDI-CV converter to the **V/OCT** input jack [**5**]. Leave all the other CV inputs unpatched.
3. Press both buttons [**A**] at the same time. The first LED should slowly blink green.
4. Send a voltage of 1.000 V to the **V/OCT** input.
5. Press either button. The first LED should now blink orange.
6. Send a voltage of 3.000 V to the **V/OCT** input.
7. Press either button.

Calibration is complete.

Firmware Warnings

Before starting the audio firmware update procedure, please double-check the following:

- Make sure that no additional sound (such as email notification sounds, background music, etc.) from your computer will be played during the procedure.
- Make sure that your speakers/monitors are muted or not connected to your audio interface—the noises emitted during the procedure are aggressive and can harm your hearing.
- On non-studio audio equipment (for example the line output from a desktop computer), you might have to turn up the volume to the maximum.

Firmware Update

Unplug all inputs/outputs from the module. Connect the output of your audio interface or sound card to the **MODEL** input jack [**1**]. Set the **FREQ** knob [**B**] to 12 o'clock. Power on your modular system with the **LEFT** model selection button [**A**] pressed.

When you are all set, play the firmware update file into the module. While the module receives data, the first group of 4 LEDs will act as a VU meter (2 or 3 LEDs are lit when the signal level is optimal), while the remaining 4 LEDs represent which proportion of the current packet has been received. You can use the **FREQ** knob to boost or reduce the gain. When the end of the audio file is reached, the module will automatically restart. If it doesn't restart, please retry the procedure.

If the signal level is too weak, the LEDs will blink red. Press the **LEFT** button [**A**] and retry with a higher gain. If this doesn't help, try the procedure from a different computer or audio interface, and make sure that no other equipment (such as an equalizer or FX processor) is inserted in the signal chain.

SWD & JTAG Programming

The module can also be programmed using an SWD or JTAG programmer connected to the black shrouded header labeled **SWD/JTAG** (H7) on the module's PCB. Power must be supplied separately via a Eurorack power cable connected to the **POWER** (H8) header.

An example of this kind of programmer is the *Olimex ARM-USB-OCD*. Their *ARM-JTAG-20-10* adapter is also necessary in order to accommodate the module's miniature 0.05″ pitch 10-pin header.

No additional steps are necessary to prepare the module for programming via this method.

If you are interested in SWD or JTAG programming, or in further firmware tweaking or development, please explore these additional online resources:

- *Vagrant environment for Mutable Instruments modules hacking*
  github.com/pichenettes/mutable-dev-environment
- *How to compile and upload the firmware of MIs eurorack modules*
  forum.mutable-instruments.net/t/4336
- *How to get started writing your own firmware for Mutable Instruments Clouds*
  medium.com/music-thing-modular-notes/a08173cec317

Documentation

Much of the information in this manual was created by Émilie Gillet of *Mutable Instruments* and released under the **CC BY-SA 3.0** license. This content includes both written text and several images.

Modifications and additions to this material were created by *Tall Dog Electronics* and are released under the compatible **CC BY-SA 4.0** license.

For an overview of the implications and terms of these licenses, please read the following page.

CC BY-SA 4.0

This is a human-readable summary of (and not a substitute for) the license, which can be found at **creativecommons.org/licenses/by-sa/4.0/legalcode**

You are free to:

- **Share** — copy and redistribute the material in any medium or format.
- **Adapt** — remix, transform, and build upon the material for any purpose, even commercially.

Under the following terms:

- **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- **No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Support

For all support inquiries, please send an email to **support@tall-dog.com**

Who We Are

Tall Dog Electronics is located in the Pioneer Valley region of Western Massachusetts. *Tall Dog* has primarily focused on producing a variety of breakout boards for the *Teensy* microprocessor development platform and conducts the majority of its business via the *Tindie* marketplace. *Tall Dog* released their first Eurorack module, *μBraids SE*, in late 2017.

Daniel Gilbert is a designer and engineer with a background in film, photography, and animation. After graduating from *Hampshire College* he spent the next several years working in the Los Angeles film industry. He now resides in Easthampton, Massachusetts, where he designs, builds, and distributes *Tall Dog* products.

Michael Gilbert is a composer, recording artist, and teacher of electronic music, for over 40 years. His music is a creative mix of electronic, jazz, world, and contemporary classical idioms, and is available on 9 albums of original work as *Michael William Gilbert*. The music has featured Adam Holzman, Mark Walker, Peter Kaukonen, David Moss, and Tony Vacca. He has also designed and built electronic music equipment, using it in his own studio and making it available to other musicians.

µPlaits SE User Manual
Revised 2019-05-30
Documentation CC BY-SA 4.0