

# NucleoSynth\_F446 Operating Manual

Status 21 November 2022

The NucleoSynth is a digital 10 voices polyphonic Audio Synthesizer built around the STM32 Nucleo-F446 evaluation board. The synthesizer is controlled completely digitally.

For sound generation 2 Oscillators, 3 Lfos, 11 independently operating ADSRs, a 300ms delay, 5 configurable digital filters are on board. Furthermore, 8 drums with digitally stored patterns are provided.



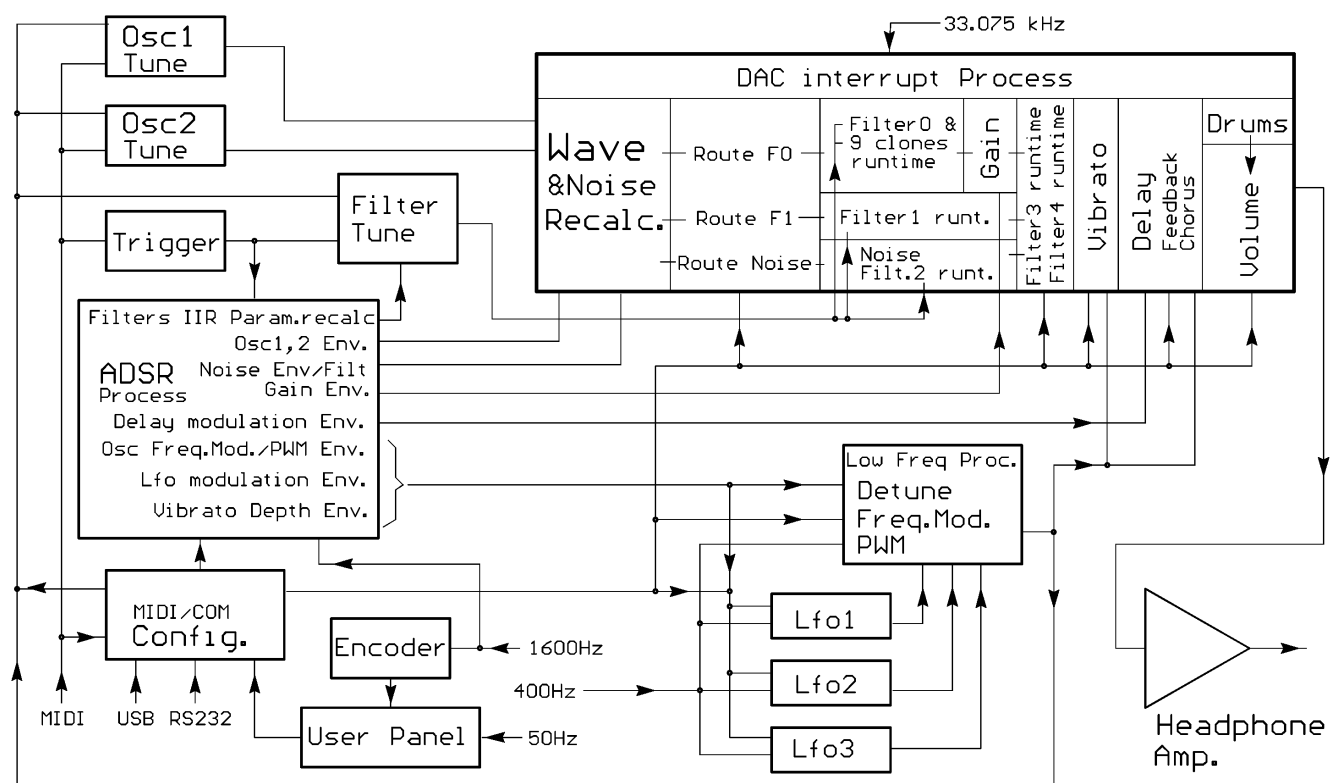
Sequencer and Arpeggiator are not implemented because it is impossible to implement a good tool in this restricted environment. As sequencer, I use my "Rhythmic" software <[www.midi-and-more.de/rhythmic.htm](http://www.midi-and-more.de/rhythmic.htm)> on a small Notebook.

To get a most compact synthesizer, the ST-LINK part of the Nucleo board must be cut off and an 8MHz crystal clock oscillator must be added on the main part as described in the Nucleo manual. Details see DIY manual.

An alternative firmware for an unmodified Nucleo board is available.

A simplified "Minimal Hardware" version for an unmodified Nucleo board without operator panel is described at the end of the manual.

## Simplified Block Diagram



Digital sound data are transformed to an analog output signal using the internal DAC of the STM32F446 processor and finally amplified to headphone impedance with an audio amplifier IC. This has some advantages concerning the hardware complexity, but disadvantages too. Due to the low operation voltage and 12 bit resolution of the DAC, the "headroom" of the sound signal is quite restricted, especially when played heavily polyphonic and filters have high resonance. There are many hooks for volume setting and the user/instrument player is demanded to find her/his optimum balance between loudness/noise margin and possible distortion. With default volume setting, an S/N ratio of ca.50 dB is achieved at the analog output. Up to 10 dB better if not connected with USB data (powered by internal accumulator).

The **program flow** has to pass two critical bottle necks: one is the 33.075 kHz interrupt routine of the DAC, where up to 24 waves have to be read and their pointers updated simultaneously, noise is recalculated, the actual signal filtering of up to 12 IIR filters has to be recalculated in realtime and the digital delay must be handled. The other one is the 1.6 kHz main level routine for ADSR processing.

As long as all low level tasks fit into the DAC interrupt time slot, the system is stable. If the remaining CPU time is too short for handling all high level tasks (essentially ADSRs and recalculation of Lfo driven features), the high level processing gets increasingly slowed down up to 2x, but it does not cause a crash or reset by watchdog. This has been tested carefully, but unexpected situations may exist.

Without use of an oscilloscope, distorted sound levels are recognized by a harsh sound. Extreme modulation of filters with high resonance setting (e.g. by high Keytrack or Square Wave Lfo) may cause audible sound clicks, which is a result of filter behaviour as such. Zero settings of Attack, Decay and Release or heavy Chorus may cause audible sound clicks, too, which is a physically necessary result of a sudden "knick" in the output volume. Real musical instruments cannot play such harsh level changes (strings e.g) or are designed for it (percussion e.g). Please take this into account and adjust filters or ADSR times in case.

## Data Communication

**A Local Operator Panel is provided** (except "Minimal Hardware"). This is the most intended mode of operation and completes the NucleoSynth as a real user playable instrument. But for tests or "embedded" solutions driven by other software it is not absolutely necessary.

**Playing the instrument and configuration is possible via USB and USART.**

**Modified Nucleo:** Communication modes are selected by a DIP switch or at operator panel

--- left switch OFF: USB works as USB CDC device (→virtual COM port)

--- left switch ON: USB works as MIDI interface on the PC

--- right switch OFF: USART works as conventional MIDI IN (fixed 31250 Baud).

--- right switch ON: USART works as RS232 with 115.2 kBaud. Features like USB COM port.

Firmware for **unmodified Nucleo:** the DIP switch is not supported. USART1(5p DIN) is permanently MIDI IN, USART2 is always virtual bidirectional COM via ST-LINK, 115.2kBaud. STM32F446 USB works as described for modified Nucleo. Mode is selected at operator panel

## Local Operation with Keys, Encoder and LCD Display

The **top(UP) and bottom(DOWN) pushbuttons(keys)** are used to select a "theme" (group of similar features): General, Oscillator, ADSR, Delay, Filter, Routing, System Configuration

The **right(FWD) and left(BACK) pushbuttons(keys)** are used to select items of the selected theme in cyclic order. The actual selection is marked with the cursor.

The **rotary encoder** is used to adjust parameter values of the selected item.

**When the encoder knob is pushed down**, a switch is closed, which changes levels of operation.

Some themes have a **"Plus"** level, which is activated with a **short** pushdown. Then the green LED is ON and the LCD display shows another set of items. When the encoder knob is pushed down again, the **"Basic"** level is activated again and the green LED is off.

When the **encoder is pushed down for one second**, the **"Play" mode** is activated, which allows fast access to the most essential featured while playing the synthesizer. Another long push on the encoder switches back to **"Configuration" mode**, which is default after power on. Differing from common use, the cursor is blinking left hand of the parameter to be changed for better readability on the display.

## THEMES and their ITEMS

### Configuration mode

#### General:

This theme is opened at synthesizer start

V-6	S:0	Py	G1:0
VL1:50	B+32	T±0	

#### **V:**

sets the final output volume between -46dB (i.e. output is off) and max. +6 dB. Keep in mind, that the output level has a limited "headroom". Especially when played heavily polyphonic or filters are highly resonant, it may be advised to use lower Volume. For this reason, -6dB is chosen as default volume.

#### **S:**

selects the active SoundSet. Range 0..59. Defaults: SoundSet #0 uses only Osc.1. SoundSet #1: very elementary piano style sound. All else Osc1, Osc1 primitive organ style.

#### **Py or Mo:**

Py activates polyphonic playing (default), Mo activates monophonic playing style. Usually Polyphony works up to 10 voices, may be limited in some cases.

#### **G1:**

sets Glide (Portamento). Very simple implementation, works only on "legato" played notes.

#### **Vib:** or **VL1:** or **VL2:**

sets the Vibrato depth between 0 and 100%. If Vibrato is off, "Vib" is displayed. If the encoder is turned clockwise, "VL2" is displayed. Then Vibrato depth is modulated by Lfo2. If the encoder is turned counter clockwise, "VL1" is displayed and Vibrato depth is controlled by Lfo1. Range 0...100%. (100% is displayed "^").

#### **B:**

controls the Bass tone level. Default state is neutral. Turning the encoder clockwise, Bass is increased. Turning the encoder counter clockwise Bass is decreased. Range -64...+63. Technically this feature modifies Peak values of filter#3. In "Easy" routing configuration, this detail is not relevant for the user. In "Flex" routing configuration the Bass control works on the Peak setting of filter #3, whatever the filter configuration is, though B is displayed.

## T:

controls the Treble tone level. Default state is neutral. Turning the encoder clockwise, Treble is increased. Turning the encoder counter clockwise Treble is decreased. Range -64...+63. Technically this feature modifies Peak values of filter#4. In "Easy" routing configuration, this detail is not relevant for the user. In "Flex" routing configuration the Treble control works on the Peak setting of filter #4, whatever the filter configuration is, though T is displayed.

**Plus State** (short push on encoder button, green LED is ON)

Ds: Kick	L-3	EAS
FOM2: 0	G: 1	FOTO

## Ds:

selects the drum for adjustment of relative level.

Drums are triggered by a Note On message with the corresponding note value of the "General MIDI Drum Kit". In addition to the explicitly named drum types, these drums are triggered if MIDI messages for similar drum types are received.

By default the MIDI channel for Drums is 10 - but can be changed to any. If the MIDI channel for chromatic notes is the same, music has priority. Drums will cause hanging notes then.

"All" modifies the level of the complete drum set, else the level of the selected drum type is modified.

## L:

adjusts the relative level of the selected drum. Drum level can be changed between -40dB (=Drum Off) and +12dB. The dB values are relative, because each drum level is modified by "All" and global output volume "Vol" too.

## EAS or FLX:

Selects a specific set of filters and routing configuration ("Easy" or "Flex", see Routing theme). Setup data of both configurations are stored independently. Further details see Routing theme.

## FOM2 or FOM1:

sets the amount of filter#0 MidFreq modulation by Lfo. Default state is modulation OFF. Turning the encoder clockwise, Lfo2 is used as modulation source. Turning the encoder counter clockwise, Lfo1 is the source. Range 0...99.

## G:

sets the amplification factor of the Gain stage following filter#0, no direct influence on filters. Range 1...50. **With setting "1" (default) the Gain stage is passed by, ie. amplification = 1,** but no extra effects (saves CPU time). Higher values amplify the sound, a **soft level clip** works behind the amplification, the sound is distorted softly towards a square wave and the **Gain ADSR** works behind the soft clip stage. This feature is sensitive and takes care for good results!

A more or less smooth transition may be achieved with a proper combination of the Oscillator(s) ADSR setting and the Gain factor. Because the clipped sound is always very loud, this is regulated with the Envelope and Sustain of the Gain Adsr, preferably with short Attack. The effect sounds harsh and should not be overstressed. Could be used as "Heavy Metal" guitar simulation.

### **F0T or F1T:**

provides a white noise on the cascade of filters#0 or #1 and 3,4 instead of discrete note sounds. If the encoder is turned clockwise, the noise is fed through filter#0, turned counter clockwise, the noise is fed through filter#1. Playing the keyboard has no influence on the noise level, but ADSRs working on filter#0 are triggered.

Position 0 (default) disables this white noise and activates normal playing of notes.

Intended for experimental filter optimization.

The filter shape can be viewed and optimized with a spectrum analyzer. (I am using "DL4YHF Spectrum Lab" (free download, see Google) with "Scarlett 2i2" audio interface)

On headphones and loudspeakers the sound is ugly, turn them off!

Not to be confused with the Noise channel which is implemented for normal playing.

### **Oscillators:**

Osc1 Tr±0 Dt±0
Mod: OL1 FM Sine

Osc2 Tx+12 Dx±0
Pwm32L2    Sq25%

**Osc1 or Osc2.** (Lfo1,2,3 see below) selects the oscillator which is actually configured

#### **if Osc1 is selected:**

##### **Tr:**

note frequency transposition by semitones equally for Osc1 and Osc2.

Transposition can only be changed while the note is OFF

##### **Dt:**

fine pitch detune equally for Osc1 and Osc2.

Detune can be changed at any time while a note is playing-

##### **Mod or Pwm and L1 or L2:**

Frequency Modulation or Pulse Width Modulation exclusively of Osc1 by Lfo1 or Lfo2.

Different or inadequate setup of this feature for Osc1 and Osc2 may cause strange sound.

(Because Osc1 and Osc2 may work differently with square waves/PWM or not, this feature works **only on Osc1**. This differs from the ADSR "FreqMod Osc1&2", which handles frequency modulation of Osc1 and Osc1 commonly, no matter if they produce square waves or not)

Encoder turned **clockwise**, **Frequency Modulation** is selected ("Mod").

Exclusively if the waveform of the Oscillator is square wave (50, 30, 25 20 or 14%), and the encoder is turned **counter clockwise**, **Pulse Width Modulation** gets active ("Pwm").

Which Lfo is used, is selected by item L (Lfo 1 or 2).

After the waveform of Osc1 has changed from Square to another waveform or vice versa, this setting possibly has to be readjusted.

Due to limited CPU speed,FreeSquare waves are always modulated with vertical gradient(slope=0)

##### **|| or FM:**

if || is displayed (default), Osc1 operates free oscillating with the selected wave form.

If FM is displayed, Osc1 is frequency modulated by Osc2 (simple FM synthesis).

#### **if Osc2 is selected:**

##### **Tx:**

additional note frequency transposition by semitones exclusively for Osc2.

**Dx:**

additional fine pitch detune exclusively for Osc2.

**Mod or Pwm and L1 or L2:**

same as described above for Osc1.

**|| or SY:**

if || is displayed (default), Osc2 operates independently.

If SY is displayed, the wave phase of Osc2 is reset when Osc1 starts a new wave loop (Osc2 synchronized by Osc1). The sound gets more compact, less interference at high frequencies.

**if Osc1 or Osc2 is selected:****Waveform:**

The waveform types are shortened as follows:

"Sine" = Sinewave	"Trian" = Triangle
"aTriS" = Sawtooth with flattened shape (soft)	"aTrH" = Sawtooth with flattened shape (hard)
"Saw" = Sawtooth	"Trapez" = Trapez, 50% square w. flattened shapes
"Sq50%" = 50% symmetric square wave	"Sq25%" = 25% square wave
"Sq20%" = 20% square wave	"FrSq1" = Square/Trapez with adjustable % and slope
"FrSq2" = Square/Trapez with adjustable % and slope	"Spike" = Sawrooth with freq*3, else zero
"Z+Si2-" = one sine wave freq*2, else zero	"Z+2S2" = 2 halfwaves sine freq*2, else zero
"Si2- -" = 2 waves sine freq*2, level shrinking	"S+Si3-" = superposition sine + 2 waves sine freq*3
"S+Si4-" = superposition sine + 2 waves sine freq*4	"SiTrp-" = sine halfwave + trapez halfwave
"Strat" = simulation of Strat humbucker B string	"ExpS3" = 3rd harmonic sine w. exponential decay
"ExpS5" = 5th harmonic sine w. exponential decay	..."User0" = Userwave no.0
"User1" = Userwave no.1	"User2" = Userwave no.2
"User3" = Userwave no.3	..."S5th" = superposition of sine and fifth interval
"Si~2S" = sequence of sine and 2 waves sine freq*2	"Si~Tp" = sequence of Sine wave and Trapez wave
"S~Str" = sequence of Sine wave and Strat wave	"S~E5S" = sequence of Sine wave and ExpS5 wave
"Us0~1" = sequence of Userwaves 0 and 1	"Us0~2" = sequence of Userwaves 0 and 2
"Us0~3" = sequence of Userwaves 0 and 3	"Us1~2" = sequence of Userwaves 1 and 2
"Us1~3" = sequence of Userwaves 1 and 3	"Us2~3" = sequence of Userwaves 2 and 3

..."Noise" = white noise as oscillator waveform ( don't confuse with Noise channel !)

Mainly intended for frequency modulation of Osc1 by Osc2 (gives smoky to raw sounds). Else no great effects. Following waveforms are exclusively available for Osc1:

Osc2 Noise is generated at runtime, Osc1 Noise generated 1x when selected (always slightly different sound).

..."Si4th" = superposition of sine and fourth interval	..."SM3rd" = superposition of sine and major third sine
"UsAll" = sequence of all 4 Userwaves	"S~3S" = sequence of silence, sine and sine freq*2
"4Si- -" = seq. of 4 sine waves with shrinking level	"Tp~3S" = sequence of 1 trapez and 3 sine waves

**Osc Plus State** (short push on encoder button, green LED is ON)

FS1Hi 33.3 SL 16.4
FS2Hi 14.0 SL 1.6

This is the only plus state item and always displayed, anyway from where selected:

**FS1Hi or FS2Hi**

sets the shape (high range percentage) of Free Sqare waveform FrSq1 or FrSq2 (0..50%).

**SL**

Slope (rise/fall gradient) of Free Sqare waveform FrSq1 or FrSq2 (0.. hi Range/2).

Transforms the square wave into an asymmetric trapezoid wave.

Automatically scaled down to (hiRange / 2) if it would exceed this limit.

if Lfo1, 2, 3 is selected:

```
Lfo2 Freq: 1.0Hz  
Trian Slp+ Sym50
```

**Lfo1 or Lfo2 or Lfo3:**

selects the Lfo to be configured.

**Freq:**

sets the Lfo frequency between 0 (=OFF) and 24.8Hz. Frequencies up to 0.6Hz are selected in steps of 0.1Hz, above selection is made in steps of 0.2Hz.

**Sine or Trian or Squar:**

Selects the Lfo waveform

**Slp:**

Start slope of the Lfo (+ or – selectable). Particularly at low Lfo frequency, this will result in different audible changes of sound.

**Sym:**

Shapes Trian asymetry between 10%(fast rising sawtooth), 50%(symm. triangle) and 90%(slow rising sawtooth).

Shapes Squar between 10 and 90% high phase.

Not used with Sine.

**ADSRs:**

Instead of selecting ADSR resources in a routing matrix, a bigger number of ADSR structures for selected features is implemented. **Each ADSR is triggered by a MIDI Note On message. The ADSR is switched off, if Env=0 and Velo=0.** This saves CPU runtime if the specific ADSR is not needed for sound shaping.

```
ADSR: Osc1  
Velo±0 Env: 64
```

```
ADSR: Osc2  
Velo±0 FMi: 16
```

```
ADSR: Trebl  
k1 Velo+72 Env+20
```

The most basic ADSR setup is used for Osc1, Osc2 and Filter0 mid frequency shift. These ADSRs have a separate instance for any pressed key.

**ADSR:**

selects which ADSR shall be configured.

Following ADSR types are implemented. Technically all of them can be used simultaneously, though this may result in confusing sounds and is not recommended.

Osc, Osc2, Noise Envelope,

FreqMod Osc1&2 simultaneously (works in addition to constant detune and modulation by Lfo),

Lfo1FreqMod, Vibrato depth, Bass, Treble, Gain, Filter0MidFreq, Filter0Peak, FreeAdsr.

If turned almost fully clockwise, the "**Free ADSR**" is selected. It can be assigned to one effect at time. Specific details see below.

**Vel:**

(Velocity) is added to the Env level corresponding with the NoteOn velocity (range -28...+99)

**k:**

limits the number of polyphonic played voices which will trigger special effects

Not applied for Osc1, Osc2, F0mid frequency, which have instances for up to 10 voices.

All ADSRs else only have only 1 instance.

Default and most recommended is k=1, ie. only the first key after silence triggers the effect.

Higher values may cause irritating sounds. Range 1...9.

The k parameter is global, i.e the choice is always the same, anyway where modified.

**Env: or FMi:**

the fixed part of the max. Envelope level. May be increased or lowered by the Vel part.

Range 0...127 (Osc1, Osc2, Noise, Gain) or -64...+63 else

When Oscillators **work in FM Synthesis mode** (see **option || or FM** in theOsc1 theme), **for Osc2 the modulation index FMi is selected and displayed** (subjective scale - not identical with "academic" FM index.)

Oscillator ADSR settings have great influence on FM sound. Only simple Osc1 sound if FMi = 0 or ADSR is totally faded down !!

**For filter ADSR's:** the max frequency shift can be set to about +1 or - 1/2 octave.

**Free ADSR:**

```
Free:NoisF!Vibra
k1 Vel±0 Env+12
```

selects one of some less important ADSR features.Following features are available:

Additional FreqMod of Osc2 (FmO2+ works in addition to constant detune and modulation by Lfo),

Lfo2FreqMod, Noise Filter MidFreq, Chorus, Dry level (not delayed total volume),

Delay Level, Feedback amount of delayed signal.

Big and rapid ADSR level changes for any delay related ADSR may cause crashing or hopping sounds, which is a natural effect caused by rapid phase changes of the resulting output signal.

**The selection directly following "Free"** marks the instance whose parameters are

**adjusted currently.** The selection following the **"!" mark** selects the FreeADSR instance

which is **active during instrument play.** Must be adjusted, too, if the changed feature shall be active for a test,e.g.

**Plus State** (short push on encoder button, green LED is ON)

```
Att0.02 Dec0.00
SusHo100 Rls0.02
```

```
Att0.02 Dec1.00
Sustn:0 Rls0.02
```

The left default setup is shown at Osc1, Osc2 and Gain. All else have the right default config.

**Att: , Dec: Rls:**

configures Attack, Decay and Release. The transition time is displayed in seconds and parts of seconds.



For times up to one second, the raster is 20 milliseconds. Above the raster is 50 milliseconds, providing a max effect time of 4.85 seconds (based on the internal range 0..127).

### **SusHo or Sus1x:**

The Sustain level (range 0..100) is the percentage of the sum of Env and Velo, but cannot be negative.

If Sus is **followed by H**, the the envelope is of "**hold**" **type**, which means that the sound is held at Sustain level as long as the key is pressed. When the encoder is turned **counter clockwise, the "Ho" changes to "1x"**. Then the envelope is a "**one shoot**" **type**. This means that the level decays to Sustain level and then immediately is faded to zero during the Release phase, even if the key is pressed longer. Key release starts the Release phase immediately.

Care has to be taken, if an ADSR is activated with Attack = zero or Decay = zero and Sustain < 127. Though internal mechanisms for cross zero switching are implemented, this may cause a click, which is a natural result of this configuration. A similar effect may be experienced when Sustain is high and Release is = zero.

**For filter ADSR's:** the max frequency shift can be set to about +1 or - 1/2 octave.

### **Delay:**

DLv: -33	T: 150.7
ChorL3: 16	Fb: ±0

### **DLv:**

Sets the **level of the delayed signal** when it is mixed with the undelayed signal in %. The actually played signal is fed together with the scaled Feedback into the digital delay chain. When the encoder is turned **counter clockwise**, the displayed level gets a negative sign. In this case, the delayed signal (output of the delay chain) is mixed with **180 degree inverted phase**, but the amount is the same in %.

### **T:**

sets **DelayTime** in 1/10ms units up to 300.0ms. This feature **takes 2 steps of Key\_FWD** and Key\_BAK. The left position changes the delay time coarsely in 10ms steps, the right position does fine tuning in 0.1ms steps. This way, rather elementary comb filter effects are possible.

### **Chorus: or ChorL2: or ChorL3:**

**Chorus modulates the DelayTime** in a subjective scale driven by Lfo. With default display "Chorus" the Delay Time is not modulated. If the encoder is turned clockwise, the display changes to ChorL2, in this case Chorus may be modulated by Lfo2. If the encoder is turned counter clockwise, the display changes to ChorL3, in this case Chorus may be modulated by Lfo3. Range in both cases 0...99.

Because the Chorus effect modulates the length of the delay buffer, the delayed signal has an abrupt phase change between consecutive samples. Beyond a certain level, this may cause an audible click, depending on several factors: Chorus width, Lfo frequency, delay level, audio signal frequency. For this reason, the Chorus effect should be used carefully.

### **Fb:**

Sets the portion of the delayed signal (in %, before the delayed signal is scaled by the setting of "DLv"), which is **fed back into the delay** chain together with the actually played signal. But the signal read at the end of the delay chain (delay plus feedback) is controlled by "DLv".

When the encoder is turned **counter clockwise**, the displayed level gets a negative sign and the feedback is inserted with **180 degree inverted phase**, but the amount is the same in %.

### Filters:

**Filter#0, filter#1(alternative path if Osc1 and/or Osc2 is not fed through filter#0), filter#2(Noise) are cascaded in parallel. Followed serially by filters #3, #4 which are cascaded serially, too.**

In the Routing theme is selected if the output of Osc1 or Osc2 is fed into filter#0 or filter#1. Filter#0 is more sophisticated and has a separate ADSR clone for each played key. Filter#0 is followed by an optional Gain stage. Filter#1 is a simple alternative without any modulation features.

EA#0 F1.04 Tr:0 T6 Bwi0.88 P0.00
-------------------------------------

#### **EA# or FX#:**

selects the filter set to be configured (number 0,1,2,3,4). "EA" or "FX" displays the selected filter set for information only (see Routing theme) and cannot be modified here.

#### **F::**

sets the filter mid frequency or border frequency, depending on selected filter type

**Tr** (displayed only when filter0 is configured):

**Keytrack Value** setting: The resonance of filter0 follows the note value of the triggering key by the Keytrack value (approx. in percent). If F0=440Hz and Keytrack=100%, F0 follows exactly the played note.

#### **T:**

selects the filter type / characteristic of the actually configured filter

0 filter OFF, does not modify the sound, saves CPU time.

1 = **first order (6dB) Lowpass**

2 = **first order (6dB) Highpas**

3 = **2nd order (12dB) Lowpass**

4 = **2nd order (12dB) Highpass**

5 = **Bandpass(6dB).**

6 = **Parametric Filter.** Mid Freq, Bandwidth and Peak can be modified.

negative values it works as adjustable notch filter. With positive values it works similarly like a bandpass – but the level outside the bandwidth region is not suppressed to zero, instead the filter input level is kept. So, the peak is added to the input level.

7,8 = **LowShelf and High Shelf Filter.** These filter types are combined 12dB Highpass and Lowpass filter with Resonance setting. Their Highpass/Lowpass characteristic is adjustable with Peak in a similar way like the Parametric Filter. LowShelf filter modifies levels below border freq, High ShelfFilter modifies levels above border freq.

#### **Res: or Bwi:**

Setting of **Resonance** (filter types 3,4,7,8) or **Bandwidth** (filter types 5 and 6).Not applicable for filter types 1 and 2.

If **Res**: high values result in a flat shape, low values give a sharp and high peak. Res = 0.71 is default, which configures a "Bessel" filter, which is regarded as the optimum stopband transition without peak.

If **Bwi**: Internally, the corresponding bandwidth is stored as MIDI note value. For this reason exclusively specific values are selectable (range 20Hz to 4.97kHz).

**P:**

**Peak** setting, different meaning for different filter types:

**Filter types 1,2,3,4,5:** adds additional amplification or attenuation to the standard filter behaviour (ca. 1 dB per step). Range -10...+15.

**Filter types 6, 7,8:** default setting 0 configures a flat filter behaviour.

**Positive Peak** value increases level in the bandpass region (types 7, 8 get highpass characteristic). **Negative Peak** value decreases level in the bandpass region (type 6 gets notch behaviour, types 7, 8 get lowpass characteristic). Range -1.0 ...+10.0

In **Easy routing configuration, parameters of filters#3 and #4 are fixed**, whereas they can be configured freely in Flex routing configuration

```
EA#3
T7 par ams fixed
```

```
FX#3 F0.49
T7 Res1.00 P2.10
```

**Routing:**

```
R:Easy B±0 T-48
Osc1>F0 Osc2>F1 +N
```

```
R:Flex B+32 T±0
Osc1<FM>->F0
```

**R:Easy / Flex:**

Selects a specific set of filters and routing configuration.

In **Easy mode**, setting of filters #3 and #4 is fixed in a configuration, which provides a simple "Bass" and "Treble" control like is known from stereo appliances. This mode is intended for simply music playing without much experimentation. Filter#0 default is optimized to provide a simple increase or decrease of the middle frequency "presence" area. Apart from this, Filter#0 is freely configurable.

In **Flex mode**, filters #3 and #4 are freely configurable, a 24dB high- or low-pass. e.g. Setup data of both filter sets and routing configurations are stored independently.

**B:**

controls the Bass tone level. Default state is neutral. Turning the encoder clockwise, Bass is increased. Turning the encoder counter clockwise Bass is decreased. Range -64...+63.

Technically this feature modifies Peak value of filter#3. In "Easy" routing configuration, this detail is not relevant for the user. In "Flex"routing configuration the Bass control works on the Peak setting of filter #3, whatever the filter configuration is, though B is displayed.

**T:**

controls the Treble tone level. Default state is neutral. Turning the encoder clockwise, Treble is increased. Turning the encoder counter clockwise Treble is decreased. Range -64...+63.

Technically this feature modifies Peak value of filter#4. In "Easy" routing configuration, this detail is not relevant for the user. In "Flex" routing configuration the Treble control works on the Peak setting of filter #4, whatever the filter configuration is, though T is displayed.

**Os1>, Os2>, or Osc1(FM), +N**

Handles the routing of Osc1 and Osc2 output to filter#0 or filter#1.

If Osc1 is **not modulated** by Osc2 ( see "|| / FM" in the General theme), the left example box is displayed. Else, if Osc1 is modulated by Osc2, the right example box is displayed. In this case, only Osc1 can be routed.

**Additionally noise can be routed through filter#2 into filters#3 and #4** (if+N is selected).

The **Noise ADSR** must have positive envelope, too !

Plus State (short push on encoder button, green LED is ON)

**Easy / Flex** and **Tp** (type of filter#0) is displayed for information and cannot be changed here.

```
F1.04Bw0.88P0.00
Tp6 Easy actFilt
```

**F, Bw, P:**

Filter#0 can be configured here. Parameters are the same as described in Filter theme.

**actFilt:**

This is a little bit a gadget for users who want to configure filters and not needed for normal operation. The Back key works specially with this item.

```
Confirm KeyBK+FW
Tp6 Flex Default
```

```
Confirm KeyBK+FW
Tp6 Flex -->Memo
```

```
Confirm KeyBK+FW
Tp6 Flex LoadMem
```

```
Confirm KeyBK+FW
Tp6 Flex Undo
```

Usually "actFilt" is displayed here, this simply means that the active filter set is in action. No change will happen if any key is pressed. When each of the operations described below is performed, "actFilt" is displayed again, which means that the new configuration is active now. If the encoder is turned **counter clockwise**, "actFilt" is replaced by "Default". If first the Back key is pressed and kept held and the Fwd key is pressed too, all filters of the selected Easy or Flex set are reset to their default values. This is the "actFilt" set now and displayed as such.

If the encoder is turned **clockwise**, first "actFilt" is replaced by "Memo". If confirmed with Back and Fwd key, the active filter data are stored temporarily in a special memory array. Else nothing is changed.

If the encoder is turned further clockwise, "LoadMem" is displayed. If confirmed with Back and Fwd key, the content of the "Memo" array is reloaded as "actFilt".

If the encoder is in the last clockwise position, "Undo" is displayed. If confirmed with Back key first, held and Fwd key added, the last change of the "actFilt" is cancelled, the previous setup is restored. Only a single undo level is provided.

### **System Configuration:**

In this theme, the Back and Fwd key operation differs from the other themes.

When the theme is selected, first this display appears:

```
Configure System  
select w. KeyFWD
```

Each time the Fwd key is pressed, the display changes and different options are proposed. This procedure works cyclic and returns to the start display. The Back key cannot be used for stepping through the menu.

```
Burn Activ>Flash  
Confirm KeyBK+FW
```

```
CopySnd#?Enc: 0  
Confirm KeyBK+FW
```

Following actions are selectable: load all Flash presets to SRAM, burn complete SRAM user configuration into Flash, copy the actual sound set into another, dump sound set parameters to PC, change MDI channels, change USB COM and MIDI; change USART COM and MIDI. If the corresponding DIP switch is OFF, selection with the operator panel overrides it. But if the DIP switch is ON, no change by operator panel is possible, even if displayed.

For some options, a more detailed command is selected with the encoder. If the selected operation shall be performed, first the Back key is pressed, kept held and the Fwd key is pressed too. If only the Back key is pressed, no action is performed and the display changes back when the Back key is released. The new setup is stored in SRAM (except "burn to flash" and "sound dump" operation, which don't modify the active setup) but is lost at power cycle or reset.

**Plus State** (short push on encoder button, green LED is ON)

```
Vid: 0483Pid: 5740  
select w. KeyFWD
```

In this state, each push of the Fwd key shifts the cursor to the next digit, which can be modified with the encoder. Fwd key shift is cyclic. If confirmed with Back and Fwd key, the new Vid/Pid is stored in SRAM. To make it active, the complete setup must be stored in Flash and a reset must be performed. Please notice, that an appropriate driver for the new Vid/Pid must be installed on the PC, too.

**Please note: the default Vid/Pid is the one of the ST-LINK provided with the Nucleo64 board. In this context, its use is only allowed for test and setup. For any (public) use else, you are responsible for the rights of the Vid/Pid you are using !!**

Be careful when changing Vid/Pid not to shoot your USB access. In most cases, different drivers have to be installed

## Play mode

During "Play" mode the **UP/DOWN keys don't change theme**

--- **but select and activate SoundSets** in cyclic manner. The red LED is ON on PLAY mode.

The red LED is blinking when data are received via USB or USART.

V-6	S:0	Py	G1:0
VL1:50	B+32	T±0	

**V:**

sets the final output volume between -46dB (i.e. output is off) and max. +6 dB. Keep in mind, that the output level has a limited "headroom". Especially when played heavily polyphonic or filters are highly resonant, it may be advised to use lower Volume. For this reason, -6dB is chosen as default volume.

**S:**

selects the active SoundSet in addition to UP/DOWN keys. Range 0..59

**Py or Mo:**

Py activates polyphonic playing (default), Mo activates monophonic playing style.

Usually Polyphony works up to 10 voices, may be limited in some cases.

**GI:**

configures Glide (Portamento).The implementation is rather simple, it works only on "legato" played notes.

**Vib:** or **VL1:** or **VL2:**

sets the Vibrato depth between 0 and 100%.If Vibrato is off, "Vib" is displayed. If the encoder is turned clockwise, "VL2" is displayed. Then Vibrato depth is modulated by Lfo2. If the encoder is turned counter clockwise, "VL1" is displayed and Vibrato depth is controlled by Lfo1. Range 0...100%. (100% is displayed "^^").

**B:**

controls the Bass tone level. Default state is neutral. Turning the encoder clockwise, Bass is increased. Turning the encoder counter clockwise Bass is decreased. Range -64...+63.

Technically this feature modifies Peak values of filter#3. In "Easy" routing configuration, this detail is not relevant for the user. In "Flex" routing configuration the Bass control works on the Peak setting of filter #3, whatever the filter configuration is, though B is displayed.

**T:**

controls the Treble tone level. Default state is neutral. Turning the encoder clockwise, Treble is increased. Turning the encoder counter clockwise Treble is decreased. Range -64...+63.

Technically this feature modifies Peak values of filter#4. In "Easy" routing configuration, this detail is not relevant for the user. In "Flex" routing configuration the Treble control works on the Peak setting of filter #4, whatever the filter configuration is, though T is displayed.

**Plus State** (short push on encoder button, green LED is ON)

D±0	S59	G:1	EAS
DL-100	T:60	F±0	

**D:**

adjusts the relative level of all drums. Drum level can be changed between -40dB (=Drum Off) and +12dB. The dB values are relative, because each drum level is modified by global output volume "Vol" too.

**S:**

selects the active SoundSet. Range 0..59

In Play mode, the UP and DOWN key additionally increases/decreases the selected SoundSet number in cyclic manner.

**G:**

sets the amplification factor of the Gain stage following filter#0. Range 1...50. With setting "1"(default) the Gain stage is passed by. Further details see General theme.

**EAS or FLX:**

Selects a specific set of filters and routing configuration ("Easy" or "Flex", see Routing theme). Setup data of both configurations are stored independently. Details see Routing theme.

**DL:**

Sets the **level of the delayed signal** when it is mixed with the undelayed signal in %. The actually played signal is fed together with the scaled Feedback into the digital delay chain. When the encoder is turned **counter clockwise**, the displayed level gets a negative sign. In this case, the delayed signal (output of the delay chain) is mixed with **180 degree inverted phase**, but the amount is the same in %.

**T:**

Simplified setup of delay time in milliseconds. Range 0(=delay OFF)...300.

**F:**

Sets the portion of the delayed signal (in %, before the delayed signal is scaled by the setting of "DL"), which is **fed back into the delay** chain together with the actually played signal. But the signal read at the end of the delay chain (delay plus feedback) is controlled by "DL".

When the encoder is turned **counter clockwise**, the displayed level gets a negative sign and the feedback is inserted with **180 degree inverted phase**, but the amount is the same in %.

## Short Reference of ASCII commands

Letters can be entered upper or lower case. 'i' is written lower case here for better readability

**!** (exclamation mark) Panic OFF, everything audible is terminated immediately

**?or.**(dot) send a **List of parameters** of the actually loaded/modified SoundSet

**S** select and activate **SoundSet** (0...59)

**&** followed by: 0=**polyphonic** 10 voices (default), 1=**monophonic**

**V** set **global volume** (final output stage) -46dB (= switched Off) ... +6dB). Default = -6dB.

**B** **Bass** (Easy) or **filter#3 Peak** (Flex):

64 is neutral setting: no changes of basic filter characteristic

0...63 Bass is suppressed, depending on difference from 64

65...127 Bass is enhanced, depending on difference from 64.

**H** **Treble** (Easy) or **filter#4 Peak** (Flex):

64 is neutral setting: no changes of basic filter characteristic

0...63 Treble is suppressed, depending on difference from 64

65...127 Treble is enhanced, depending on difference from 64.

**W** **Waveform Osc1** (0=Sine, 1=Triangle, 2,3=asym.Tri, 4=Saw, 5=Trapez, 6=Sq50%, 7=Sq33% , 8=Sq25%, 9=Sq20%, 10=Sq14%, and more. Detailed description see page 6)

**X** **Waveform Osc2** (0=Sine, 1=Triangle, 2,3=asym.Tri, 4=Saw, 5=Trapez, 6=Sq50%, 7=Sq33% , 8=Sq25%, 9=Sq20%, 10=Sq14%, and more. Detailed description see page 6)  
**%** followed by: 1= **Osc2 wave** is restarted at Osc1 wave start ("**synchronized**"), or  
0= Osc1 and Osc2 **waves are independent** (default)

**T** **Transpose of Osc1 and Osc2** commonly (-36...+36 in semitone steps)  
**U** additional **Transpose of Osc2** (-36.. +36 in semitone steps)  
**Y** **global-Detune** (of Osc1 and 2) (-64...+63 ca. linear)  
**Z** additional **Detune of Osc2** (-64...+63 ca. linear)  
**/** set high range (%) and slope of FreeSquare1&2 waveforms (Osc theme)  
**P** set **Glide(Portamento)** speed of Osc1 und Osc2 commonly. 0=OFF  
**L** init or readjust **Lfo1,2,3**. Max frequency 24.8 Hz in 0.1 / 0.2 Hz steps.  
Enter frequency with decimal dot. Waveform is asked too.  
**i** **Frequency Modulation of Osc1** or PWM (square wave only) by Lfo1  
(0..127 ca.+100/-50% symmetric linear frequency shift)  
**J** **Frequency Modulation of Osc2** or PWM (square wave only) by Lfo1  
(0..127 ca.+100/-50% symmetric linear frequency shift)  
**A** configure **ADSR** (Osc1, Osc2, Noise, (Gain,) FModOsc1+2, Lfo1,  
Tone Balance, Filter0MidFreq, Free assigned)  
**=** assignment to "**Free**" **ADSR** (Lfo2, Vibrato, Chorus, FModOsc2, Noise Filter MidFreq,  
Filter1MidFreq, Delay, Dry, Del<>Dry, Feedb)  
**D** set **Delay Time** (0-300.0 ms in steps of 0,1 milliseconds)..  
Enter tenths with dot.  
**E** **Delay Level** (127 ... -127, negative values = 180 degree phase inversion)  
**N** **Delay Feedback Level** (127 ... -127, negative values = 180 degree phase inversion.  
**C** configure **Chorus** depth (0..99, about peak delay shift in ms). 0 means Chorus off)  
**Q** **Vibrato Depth** (0...100%). Asks too if controlled by Lfo1 or 2  
**G** set **Gain** (0 ... 50) **0 means amplification = 1** - but ADSR and clip/limiter is **not** active  
Gain amplification works always after filter0 but ahead of the "soft clip" limiter. Finally  
the sound is shaped by the Gain ADSR (% of limiter output)  
**F** (re)configure **Filter** in the active SoundSet. Filter2=Noise  
**\*** **Filter Test** : 1-9 = Filter Test is active (confirm w. upper case Y),  
0 = back to standard filter operation  
In Filter Test mode, the noise signal is fed to filter 0 and filter1. Gain is always off in this case.  
The input level of the noise is regulated with the parameter: 1 means about max. undistorted level, 9  
means ca.22dB below max. level. To test filters with high resonance, it is recommended to use lower  
level (ca.4). Level is independent of any key pressed, but the filter modulation reacts on keys.  
Depending on routing, fractions of the unfiltered noise signal are added before or after filters3&4.  
The total spectrum is available at OUT. (**Be careful withyour ears, the output is LOUD and awful !!**)

**K** set filter **Keytrack of filter0** (0..127) in % relative to A3(440Hz)  
**M** **Frequency Modulation of filter0** by Lfo(max ca.+/-1 octave).  
Asks too if controlled by Lfo1 or2  
**R** configure **Filter Routing**. Details see p.11  
**[** change **USB Vid/Pid** (corresponding driver on PC must be installed/changed too!)  
**#** select **MIDI channel**:  
**first** for chromatic instruments Note, PgnCh, CtrlCh, **next** for drums  
**>n** **copy** actual SoundSet into SoundSet no.n , confirm with upper case Y  
**\** **load** all SoundSets from Flash into SRAM (new Flash->default), confirm w. upper case Y



- | (vertical line) **store**/burn all actual SoundSets in Flash, confirm with upper case Y
- ~ **delete all user data in Flash** = "Factory Rese"t, confirm w. upper case Y, next with – (minus) **Undo is impossible!** , so handle with care. Make a Dump before.
- } **Dump SoundSet** from SRAM formally as MIDI SysEx msg.  
Enter SoundSet to be dumped (00 to 59), If entered \*a\* or 'A', all SoundSets are dumped.  
confirm with upper case Y
- { **Upload SoundSet(s)** to Flash formatted as SysEx file,  
confirm with upper case Y , next with – (minus)  
The sound number is coded in the sound file, needs not to be entered explicitly  
The upload is terminated if no date appear after 4 seconds. This way any number of Sounds (e.g. combined in a single file) can be uploaded with one command
- @ **Upload User Wave** (0,1,2,3).  
The wave number is coded in the file, needs not to be entered explicitly
- , (comma) **simulate Note ON.** (for test or by special hard/software with serial ASCII interface)  
followed by the MIDI note value, entered as hex number (2 places!) in ASCII text.  
No SPACE, next the velocity is entered as hex number (2 places!) in ASCII text  
Input is echoed. E.g. enter text sequence ,450F to send note "A"(440Hz) with velocity=15
- (minus) **simulate Note Off.** (for test or by special hard/software with serial ASCII interface)  
or < followed by the MIDI note value, entered as as hex number (2 places!) in ASCII text
- \$ **Drum Volume:**  
A = all drums equally in addition to individual setting of drum type  
0 .. 7 = drum type as listed below
- 0..7 Trigger Drum:** (for test or by special hard/software with serial ASCII interface)  
0=Kick, 1=HiTom, 2=Snare, 3=Rim, 4=Clap, 5=HiHat, 6=Cymbal, 7=Cowbell

## Protocol of MIDI Messages

Exclusively Status Bytes with the selected "chromatic" or "drums" MIDI channel are evaluated.

A USART based MIDI-Scanner for a "legacy" MIDI IN and a separate USB based-MIDI-Scanner is implemented. **Both merge MIDI channel messages correctly**, so in fact two independent MIDI inputs are available, which can be played simultaneously. The first completely received Note, Pgm Change or CtrlChange message is executed first. SysEx messages (dump or userwaves) are not merged correctly. Other MIDI messages are not supported

Note Values below decimal 18(ca. 23 Hz) and above 120(ca. 8372Hz) are ignored.

### received PROGRAM CHANGE messages:

**(data bytes are via MIDI transferred values. Many user software needs input value+ 1 ! )**

n = active MIDI channel for chromatic instruments (user selected channel – 1 ! )

- 0xCn 0...0x3B(59) = select and activate SoundSet
- 0xCn 0x40(64)           select EASY filter set
- 0xCn 0x41(659)       select FLEX filter set
- 0xCn 0x50(80) 0x54(85) select Filter Nummer to be modified (in active SoundSet and configuration)
- 0xCn 0x60(96) OSC1, OSC2 phases work independently (default)
- 0xCn 0x61(97) OSC2 phase is synchronized with QSC1 phase
- 0xCn 0x70(120)       10 voices Polyphonic Mode (default)
- 0xCn 0x71(121)       Monophonic Mode
- 0xCn 0x7E(126)       request dump of one soundSet (the actually loaded)
- 0xCn 0x7F(127)       request dump all SoundSets

**received CONTROL CHANGE messages:**

0xBn 0x00	xxxxxxx	FilterRouting Filters 0 and Noise
0xBn 0x01	xxxxxxx	gNTrigger 3Bit number of keys trigger special ADSRs
0xBn 0x02	xxxxxxx	Bass ;7Bit
0xBn 0x03	xxxxxxx	Treble ;7Bit
0xBn 0x04	xxxxxxx	Vibrato modulated by Lfo2 ;7Bit
0xBn 0x05	xxxxxxx	Vibrato modulated by Lfo1 ;7Bit
0xBn 0x06	xxxxxxx	Glide(Portamento);7Bit
0xBn 0x07	xxxxxxx	Global Out Volume ;7Bit
0xBn 0x08	xxxxxxx	Gain ;7Bit
0xBn 0x09	xxxxxxx	OSC1 Waveform ;6Bit
0xBn 0x0A(10)	xxxxxxx	OSC1+2 Transpose ;7Bit
0xBn 0x0B(11)	xxxxxxx	OSC1+2 Detune ;7Bit
0xBn 0x0C(12)	xxxxxxx	OSC2 Waveform ;6Bit
0xBn 0x0D(13)	xxxxxxx	OSC2 add Transpose ,7Bit
0xBn 0x0E(14)	xxxxxxx	OSC2 add Detune ;7Bit
0xBn 0x10(20)	xxxxxxx	FreeSquare1 range 0..100. internally divided by 2, i.e.hiRange 0..50%)
0xBn 0x11(21)	xxxxxxx	FreeSquare1 slope 0..25. (internally limited to hiRange / 2)
0xBn 0x12(22)	xxxxxxx	FreeSquare2 range 0..100. (internally divided by 2, i.e.hiRange 0..50%)
0xBn 0x13(23)	xxxxxxx	FreeSquare3 slope 0..25. (internally limited to hiRange / 2)
0xBn 0x14(20)	xxxxxxx	LFO1 Waveform ;7 Bit (bit6 = Lfo for MOD)
0xBn 0x15(21)	xxxxxxx	LFO1 Freq ;7Bit
0xBn 0x16(22)	xxxxxxx	LFO2 Waveform ;7 Bit
0xBn 0x17(23)	xxxxxxx	LFO2 Freq 0-127 ;7 Bit
0xBn 0x18(24)	xxxxxxx	LFO3 Waveform ;7 Bit (bit6 = Lfo for MOD)
0xBn 0x19(25)	xxxxxxx	LFO3 Freq ;7Bit
0xBn 0x1C(28)	xxxxxxx	OSC1 Freq.Modulation ;7Bit
0xBn 0x1D(29)	xxxxxxx	OSC1 PWM(works only on Square Wave,else FM selected)
0xBn 0x1E(30)	xxxxxxx	OSC2 Freq.Modulation ;7Bit
0xBn 0x1F(31)	xxxxxxx	OSC2 PWM(works only on Square Wave,else FM selected)
0xBn 0x20(32)	xxxxxxx	OSC1 Attack ;7Bit
0xBn 0x21(33)	xxxxxxx	OSC1 Decay ;7Bit
0xBn 0x22(34)	xxxxxxx	OSC1 Sustain Hold type ;7Bit
0xBn 0x23(35)	xxxxxxx	OSC1 Sustain 1 Shoot type ;7Bit
0xBn 0x24(36)	xxxxxxx	OSC1 Release ;7Bit
0xBn 0x25(37)	xxxxxxx	OSC1 Volume;7Bit
0xBn 0x26(38)	xxxxxxx	OSC1 Velo ;7Bit
0xBn 0x28(40)	xxxxxxx	OSC2 Attack;7Bit
0xBn 0x29(41)	xxxxxxx	OSC2 Decay ;7Bit
0xBn 0x2A(42)	xxxxxxx	OSC2 Sustain Hold type ;7Bit
0xBn 0x2B(43)	xxxxxxx	OSC2 Sustain 1 Shoot type ;7Bit
0xBn 0x2C(44)	xxxxxxx	OSC2 Release ;7Bit
0xBn 0x2D(45)	xxxxxxx	OSC2 Volume ;7Bit
0xBn 0x2E(46)	xxxxxxx	OSC2 Velo ;7Bit
0xBn 0x30(48)	xxxxxxx	Noise Frequency Modulation Attack;7Bit
0xBn 0x31(49)	xxxxxxx	Noise Frequency Modulation Decay ;7Bit
0xBn 0x32(50)	xxxxxxx	Noise Frequency Modulation Sustain Hold type ;7Bit
0xBn 0x33(51)	xxxxxxx	Noise Frequency Modulation Sustain 1 Shoot type ;7Bit
0xBn 0x34(52)	xxxxxxx	Noise Frequency Modulation Release ;7Bit
0xBn 0x35(53)	xxxxxxx	Noise Frequency Modulation Volume ;7Bit
0xBn 0x36(54)	xxxxxxx	Noise Frequency Modulation Velo ;7Bit
0xBn 0x38(56)	xxxxxxx	OSC12 Frequency Modulation Attack;7Bit
0xBn 0x39(57)	xxxxxxx	OSC12 Frequency Modulation Decay;7Bit
0xBn 0x3A(58)	xxxxxxx	OSC12 Frequency Modulation Sustain Hold type;7Bit
0xBn 0x3B(59)	xxxxxxx	OSC12 Frequency Modulation Sustain 1 Shoot type;7Bit
0xBn 0x3C(60)	xxxxxxx	OSC12 Frequency Modulation Release ;7Bit
0xBn 0x3D(61)	xxxxxxx	OSC12 Frequency Modulation Fix Shift ;7Bit
0xBn 0x3E(62)	xxxxxxx	OSC12 Frequency Modulation Velo;7Bit

0xBn 0x40(64)	xxxxxxx	Lfo1 Balance Attack;7Bit
0xBn 0x41(65)	xxxxxxx	Lfo1 Balance Decay;7Bit
0xBn 0x42(66)	xxxxxxx	Lfo1 Balance Sustain Hold type;7Bit
0xBn 0x43(67)	xxxxxxx	Lfo1 Balance Sustain 1 Shoot type;7Bit
0xBn 0x44(68)	xxxxxxx	Lfo1 Balance Release ;7Bit
0xBn 0x45(69)	xxxxxxx	Lfo1 Balance Fix Shift ;7Bit
0xBn 0x46(70)	xxxxxxx	Lfo1 Balance Velo;7Bit
0xBn 0x48(72)	xxxxxxx	Filter0 Mid Frequency Modulation Attack ;7Bit
0xBn 0x49(73)	xxxxxxx	Filter0 Mid Frequency Modulation Decay;7Bit
0xBn 0x4A(74)	xxxxxxx	Filter0 Mid Frequency Modulation Sustain Hold type;7Bit
0xBn 0x4B(73)	xxxxxxx	Filter0 Mid Freq Modulation Sustain 1 Shoot type;7Bit
0xBn 0x4C(76)	xxxxxxx	Filter0 Mid Frequency Modulation Release;7Bit
0xBn 0x4D(75)	xxxxxxx	Filter0 Mid Frequency Modulation Fix Shift;7Bit
0xBn 0x4E(78)	xxxxxxx	Filter0 Mid Frequency Modulation Velo;7Bit
0xBn 0x50(80)	xxxxxxx	Selectable ADSR Attack;7Bit
0xBn 0x51(81)	xxxxxxx	Selectable ADSR Decay ;7Bit
0xBn 0x52(82)	xxxxxxx	Selectable ADSR Sustain Hold type ;7Bit
0xBn 0x53(83)	xxxxxxx	Selectable ADSR Sustain 1 Shoot type ;7Bit
0xBn 0x54(84)	xxxxxxx	Selectable ADSR Peak Release ;7Bit
0xBn 0x55(85)	xxxxxxx	Selectable ADSR Peak Fix Shift ;7Bit
0xBn 0x56(86)	xxxxxxx	Selectable ADSR Peak Velo ;7Bit
0xBn 0x57(87)	xxxxxxx	Select ADSR type (default:9 = Filter0Peak)
0xBn 0x58(88)	xxxxxxx	Free ADSR Attack;7Bit
0xBn 0x59(89)	xxxxxxx	Free ADSR Decay ;7Bit
0xBn 0x5A(90)	xxxxxxx	Free ADSR Sustain Hold type ;7Bit
0xBn 0x5B(91)	xxxxxxx	Free ADSR Sustain 1 Shoot type ;7Bit
0xBn 0x5C(92)	xxxxxxx	Free ADSR Release ;7Bit
0xBn 0x5D(93)	xxxxxxx	Free ADSR Fix Shift ;7Bit
0xBn 0x5E(94)	xxxxxxx	Free ADSR Velo ;7Bit
0xBn 0x5F(95)	xxxxxxx	select target of Free ADSR ;7Bit
0xBn 0x60(96)	xxxxxxx	DelayTime 10ms steps, 0...30
0xBn 0x61(97)	xxxxxxx	DelayTime 0.1ms steps, 0...99
0xBn 0x62(98)	xxxxxxx	Delay Level 7Bit (in %, delay phase = input phase)
0xBn 0x63(99)	xxxxxxx	Delay Level 7Bit (in %, delay phase inverted)
0xBn 0x64(100)	xxxxxxx	Delay Feedback 7Bit(in %,feedback phase=input phase)
0xBn 0x65(101)	xxxxxxx	Delay Feedback 7Bit (in %,feedback phase inverted)
0xBn 0x66(102)	xxxxxxx	Chorus Width, modulated by LFO2 ;0...99
0xBn 0x67(103)	xxxxxxx	Chorus Width, modulated by LFO3 ;0...99
//adjust actually selected filter(see PGM CHANGE 0x40,41 and 0x50,51,52,53):		
0xBn 0x68(104)	xxxxxxx	TYPE ;7Bit MIDI note value
0xBn 0x69(105)	xxxxxxx	FREQ ;7Bit MIDI note value
0xBn 0x6A(106)	xxxxxxx	QFACTOR ;7Bit MIDI note value or Zeta(Resonance)
0xBn 0x6B(107)	xxxxxxx	PEAK ;7Bit
0xBn 0x6C(108)	xxxxxxx	Filter0 Keytrack(%)
0xBn 0x6D(109)	xxxxxxx	Filter0 Modulation with/selects Lfo1 ;7Bit
0xBn 0x6E(110)	xxxxxxx	Filter0 Modulation with/selects Lfo2 ;7Bit
0xBn 0x6F(111)	xxxxxxx	Drums global volume; 7Bit, values 0..52
0xBn 0x70(112)	xxxxxxx	Kick drum volume; 7Bit, values 0..52
0xBn 0x71(113)	xxxxxxx	HiTom volume; 7Bit, values 0..52
0xBn 0x72(114)	xxxxxxx	Snare volume; 7Bit, values 0..52
0xBn 0x73(115)	xxxxxxx	Rim volume; 7Bit, values 0..52
0xBn 0x74(116)	xxxxxxx	Clap volume; 7Bit, values 0..52
0xBn 0x75(117)	xxxxxxx	Hihat volume; 7Bit, values 0..52
0xBn 0x76(118)	xxxxxxx	Cymbal volume; 7Bit, values 0..52
0xBn 0x77(119)	xxxxxxx	Bell volume; 7Bit, values 0..52
//Exception:		
0xBn 0x7F(127)	see above	emulate PROGRAM CHANGE with data1= this data2
//SysEx:0xF0 0x7D 0x20 Sound# upload SoundSet#		
//Sysex:0xF0 0x7D 0x40 UserWave# upload User Waveform#		

## User waveforms

Up to 4 user defined waveforms ("User0,1,2,3") can be uploaded.

The upload procedure can only be triggered externally by starting the upload with an external software (tested here with "MIDITERM", see <[www.midi-and-more.de/miditerm.htm](http://www.midi-and-more.de/miditerm.htm)> -- push F6 or F7) or "Dterm", see <[www.midi-and-more.de/more/dterm.htm](http://www.midi-and-more.de/more/dterm.htm)>), depending on the active communication mode.

Then the LCD display of any Operator Panel "theme" changes into an upload message.

For upload, the userwave must be formatted as a MIDI SysEx message (for external upload software preferably stored as .SYX file)

**The mandatory upload format** for one Userwave is as follows: (all byte descriptions in hex format)  
first byte 0xF0

second byte 0x7F (i.e. MIDI "private", or "experimental" 1 byte manufacturer ID)

third byte 0x40

fourth byte 0,1,2 or 3 (number of uploaded Userwave)

following **exactly 1024** integer 16 bit wide data words (with 14bit level data -8192...+8192), each transferred as 4 subsequent MIDI data bytes containing the word split into 4 nibbles (big endian, i.e. highest nibble first). **That means: exactly 4096 MIDI data bytes must follow!**

followed by final MIDI EOX (0xF7). The wave content is arbitrary, but should have the same positive and negative "volume" to avoid crashes of output DC level when played.

Any number of appropriate MIDI SysEx messages can be received in a sequence, the end of transfer is detected by timeout.

This highly redundant data format was chosen because it is most easy for the user to transform 14 bit integer data into the nibble based format.

After reception the user waveform is stored in SRAM and can be played, but will be lost when power is turned off or a reset is made. To save it permanently, **all user data** (60 SoundSets and 4 Userwaves) must be burnt into the microcontroller flash. Storage from SRAM to Flash must be triggered explicitly, see "System Configuration" theme or ASCII command '|'. Due to the microcontroller memory organisation, a more selectable storage is impossible. The microcontroller flash is specified for about 10.000 storage processes.

Before Userwaves are uploaded, some basic waveforms are stored there as dummies.

For tests, some example user waveforms are supplied at my website. Furthermore, **the source code for a Windows console app ("waves14.c") is provided there as a template to build your own waveforms mathematically** (unfortunately written in old devcpp 4.9.9.2).

## Organisation of Sound Dump and Upload

The **Sound Dump** (or vice versa Upload) is a MIDI SysEx conformant file **including SysEx specific control bytes and communication parameters like USB Vid/Pid, Baud rates and MIDI channels**. Detailed data structure see published source code.

To trigger the dump of one or all SoundSet(s), select the "System Configuration" theme, details see there above. Alternatively the ASCII command '}' may be used.

To **save the dump on a PC**, an appropriate software is necessary. Though dump is possible via MIDI (needs special MIDI connector, see DIY manual), dump via COM is recommended.

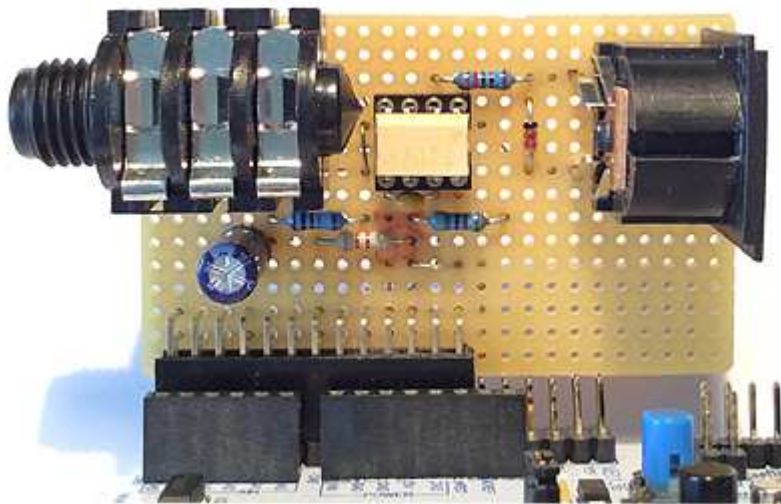
I am using "DTerm" (see <[www.midi-and-more.de/more/dterm.htm](http://www.midi-and-more.de/more/dterm.htm)>. Select ReceiveFile(F2), start reception with F9, terminate with F12). Intentionally the dump/upload contains no checksum, so a dump can be modified with a hex editor, but needs intimate knowledge of its structure.

The **upload of a dump** is triggered automatically when the adequate SysEx header is received via MIDI or COM. The sound number is coded in the SysEx file, needs not be entered. (DTERM: Select SendFile(F1) and start transmission with F5). Reception is signalled at the Operator Panel and there is no warning if runtime data are overwritten. Any number of appropriate MIDI SysEx messages can be received in a sequence, the end of transfer is detected by timeout.

Uploaded SoundSets are stored in SRAM and are lost when the power is cycled or a reset is performed. Storage from SRAM to Flash must be triggered explicitly, see "System Configuration" theme or ASCII command '|'. In this case, **all user data** (60 SoundSets and 4 userwaves) are written into the microcontroller flash. Due to the microcontroller memory organisation, a more selectable storage is impossible.

## Firmware for minimal hardware without operator panel

This firmware can be used with an unmodified STM32F446 Nucleo and a **minimal hardware design** as described at <[www.midi-and-more.de/nucleosynth/nucleosynth-hwdiyman.pdf](http://www.midi-and-more.de/nucleosynth/nucleosynth-hwdiyman.pdf)>



The onboard USB of the F446 is disabled, instead USART2 is activated and is seen by a PC (via the ST-LINK USB) as virtual COM port. (STM32F446 internally USART2). The 5pin DIN socket is always configured as legacy MIDI IN (STM32F446 internally USART1).

Configuration is possible with MIDI PgmChange and CChange commands (less recommended). **More comfortably configuration is handled with ASCII commands** via ST-LINK COM port and terminal software, e.g. Teraterm. Commands are the same as described for the standard hardware. Dump of SoundSets must be made via ST-LINK COM port, recommended for uploads too, but possible via MIDI.

**contact:** [wschemmert@t-online.de](mailto:wschemmert@t-online.de), <[www.midi-and-more.de](http://www.midi-and-more.de)>

---

\* Right of technical modifications reserved. Provided 'as is' - without any warranty. Any responsibility is excluded.  
\* This description is for information only. No product specification or useability is assured in juridical sense.  
\* Trademarks and product names cited in this text are property of their respective owners