Goblin-SPST-2.0

Installation, Configuration and Usage

This document describes the installation (page 2 - 4), configuration (page 5 - 7) and usage (page 8 - 9) of the Goblin-SPST MIDI mod version 2.0.1. For a description of the product, its features and further information visit https://oscillatordevices.com/goblin

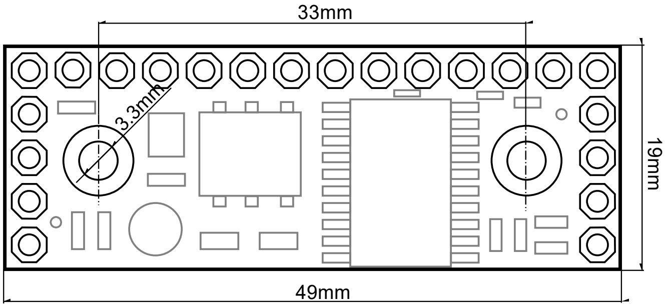
Electrical Properties

Electrical Properties	Min	Typ	Max	
Supply Voltage (+9V Terminal)	7	9	20	VDC
Current Consumption	3	10	30	mA
Maximum voltage at RLY, LED and SW terminals			5	VDC
Output voltage at SW terminals			5	VDC

The Goblin-SPST has reverse polarity protection at the V+ terminal. Be careful nonetheless to not reverse polarity upon installation, as long as other wires are connected. Negative currents could flow through the microcontroller and damage it.

Mechanical Properties

Height is approx. 8mm. If using the Goblin Screw set, account at least for 15mm height.



Caution: Please read this manual carefully before making any connections. Be careful to not exceed any voltage limits.

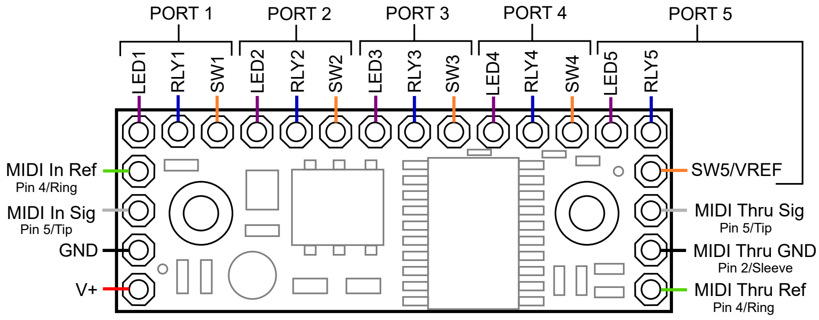
Attention: Digital signals, such as MIDI signals, can lead to crosstalk on other lines. This particularly applies to effects with multiple gain stages (distortion, fuzz, etc.). Pay attention to keep the MIDI wires as far away from the analog circuit as possible. Otherwise it can happen that a click can be heard in the audio signal with every MIDI command. To further reduce possible crosstalk, use shielded wires for audio and MIDI.

Overview

The main function of the ***Goblin-SPST-2.0*** is to take control over effects devices that are controlled with non latching SPST (or (ON)-OFF) footswitches. Additionally the 2.0 version is able to control connections for external footswitches, expression controls and – in special cases – potentiometers. All these things can then be controlled with MIDI.

The Goblin-SPST has five ports, with 3 connections each, to control five functions of the effects device. Here's a quick overview over the connections:

- **V+/GND**: This is the power supply. Connect them directly to the DC jack of the guitar pedal.
- **MIDI Sig**: This is the active MIDI signal. It is connected to **Pin 5** of a DIN 5-PIN MIDI Connector, or **Tip** if a TRS connector according to MIDI standard (Type A) is used.
- **MIDI Ref**: This is the reference MIDI signal. It is connected to **Pin 4** of a DIN 5-PIN MIDI Connector, or **Ring** if a TRS connector according to MIDI standard (Type A) is used.



- **MIDI GND:** This is the GND signal for MIDI Thru. It is connected to **Pin 2** of a DIN 5-PIN MIDI Connector, or **Sleeve** if a TRS connector according to MIDI standard (Type A) is used. Note that only MIDI Outputs have a connection to GND. Do not connect GND to the MIDI Input.

The connections of the ports have different functions depending on the ***Role*** of the port. The first port has always the role ***Switch*** and is meant to be used for the bypass footswitch and to set the MIDI channel.

	Default Role	Switch/Tap Tempo	External Footswitch	Expression/Potentiometer/CV
Port 1	Switch	Yes	No	No
Port 2	Disabled	Yes	Yes	Yes
Port 3	Disabled	Yes	Yes	Yes
Port 4	Disabled	Yes	Yes	Yes
Port 5	Disabled	Yes	Yes	Yes

Installation

The installation consists of two parts. The actual wiring, which is described in this chapter, and the configuration described in chapter Configuration.

Ports and Roles

The installation depends on how the five ports of the Goblin are used, i.e. what ***Role*** they have.

- Role ***Switch***: A switch is, for example, a conventional bypass footswitch, but can be any type of SPST non latching switch. In that role the state set by the footswitch (not MIDI) is remembered after powering off.
- Role ***Tap Tempo***: The same as role *Switch*, but without the status monitoring.
- Role ***TRS:*** In the role of *TRS*, the Goblin imitates an external switch with “Ring” and “Tip” (often marked with Footswitch, Ext-Ctl, CTL, EXT). The two lines can be open or closed to GND. This way, external switch connections can be operated from inside the device.
- Role ***Pot/CV***: In the role *Pot/CV*, the Goblin generates a control voltage between 0V and 5V. It also can receive a voltage, for example from a potentiometer. The control voltage is then either the same voltage than the received voltage, or any voltage set by MIDI. This can either be used for an expression port or to emulate a potentiometer that is used for a control voltage.

All five ports are able to take on any role, with the exception of port 1, which is always a *Switch*.

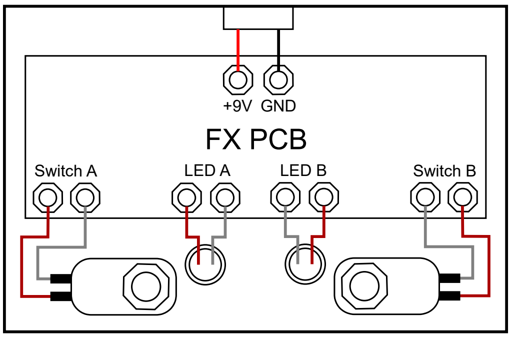
Which Port for Which Role?

Always use port 1, 4 and 5 first. Only when these are occupied, move to port 2 and 3.

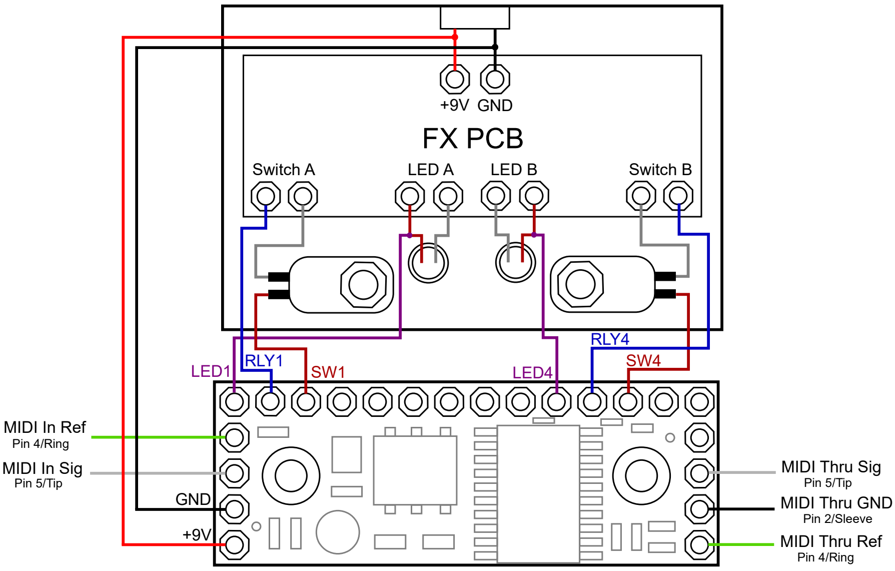
The reason is, that there is a small filter at RLY2 and RLY3. This leads to a slower rise and fall time. The reason is that the Pot/CV function on these ports are driven by PWM instead of a DAC like port 4 and 5.

Installation for Role Switch and Tap Tempo

In the images below you can see an example of a typical dual footswitch, non latching SPST, guitar effects, such as EQD, Walrus, or Wampler. The first image depicts the device before installation, the image after that shows the wiring for that particular case.



Dual Footswitch Effects Device



Dual Footswitch Effects Devices with Goblin-SPST-2.0 installed, using Port 1 and Port 4

Switch Connection (SW)

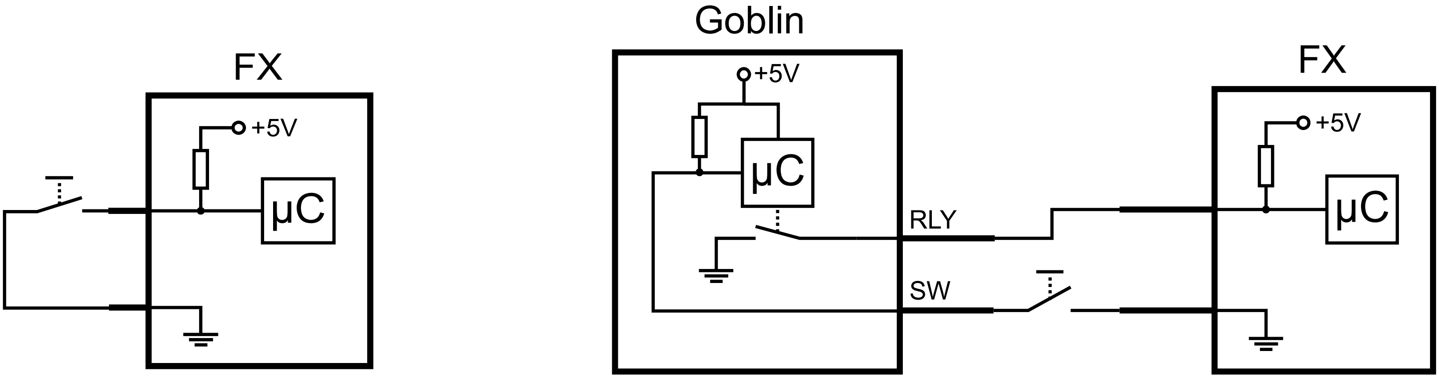
The connection SW is for connecting a non latching SPST switch. All SW terminals have internal 5V pull ups. The other side of the switch has to be connected to GND, so the switch connects SW to GND, when closed.

For regular cases, like the one above, the active (or high) side of the switch has to be disconnected from the effects device and connected to the Goblin, while the GND side stays connected. To find the active (or high) side, measure both sides with a DMM, do not rely on wire colors.

Most switches are normally open. In rare cases there are normally closed switches. Refer to chapter Configuration for further information.

PCB connection (RLY)

The connection RLY emulates a switch. For that it connects to GND for a brief moment. In the picture below, left side, you can see a typical effects device switching scheme. A microcontroller (µC) detects the switching, when the 5V line is pulled to GND, which happens when the switch closes. On the bottom right you can see how the Goblin connects to that scheme. Instead of the switch, the internal microcontroller of the Goblin pulls the line to GND.



Before connecting the Goblin it is important to control, that the voltage on the effects device does not exceed 5V. Everything below is fine.

Status monitoring connection (LED)

With the connection SW and RLY the FX device can be switched by the Goblin. If you're dealing with a Tap Tempo switch, or anything else that doesn't have a particular state, like on or off, you can skip this step.

If you're dealing with say, a bypass switch, simple switching is not enough. The state is important and the Goblin needs to know that state to carry out an "on" or "off" command. For that the Goblin can measure a voltage and determine the current state with that. The only thing that is needed for that is a voltage that changes with the state of the effects device and is not higher than 5V. In most cases the voltage of the LED is perfect for that.

To find out to which pin of the LED the Goblin has to be connected to, proceed as follows:

1. Switch the effects device **on** and measure the voltage on both pins of the LED relative to GND.
2. Switch the effects device **off** and measure the voltage on both pins of the LED relative to GND.
3. On one of the pins the voltage changes. Connect this pin to LED on the Goblin. Write down the measured voltages, we need them later.

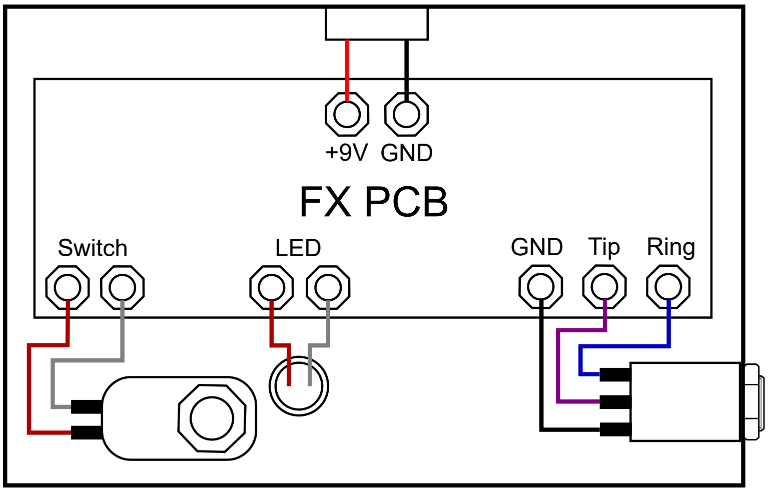
Here is a real world example of the EQD White Light. Two pins of the LED (called A and B) are measured to GND:

	Effects device on	**Effects device off**
Pin A	0V	0V
Pin B	1.8V	0V

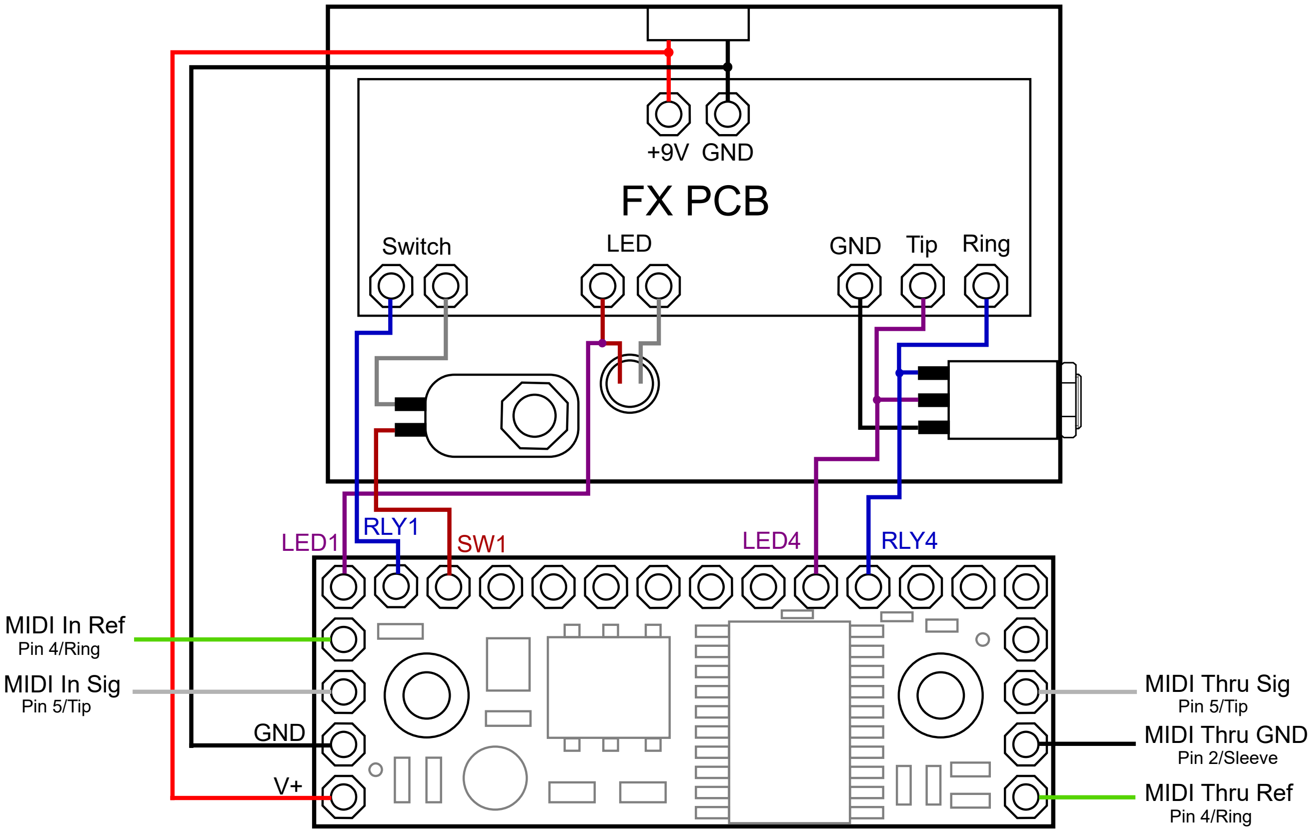
For this example, Pin B would be used for the LED connection.

Installation for Role TRS

Besides foot switches, the Goblin-SPST-2.0 can control external foot switch connections, like connections for external tap tempo switches, Strymon FAV-Switches etc. In the image below is a typical effects pedal with a bypass foot switch (left) and a TRS socket for an external foot switch (right).



Effects Device with Foot Switch and External TRS Connector



Effects Device with Foot Switch and External TRS Connector with Goblin-SPST

For a Port in the Role TRS the wiring is very simple. The corresponding lines are soldered directly to the pins of the socket. To operate the switch, the Goblin either leaves the lines open (switch open) or closes them to GND (switch closed).

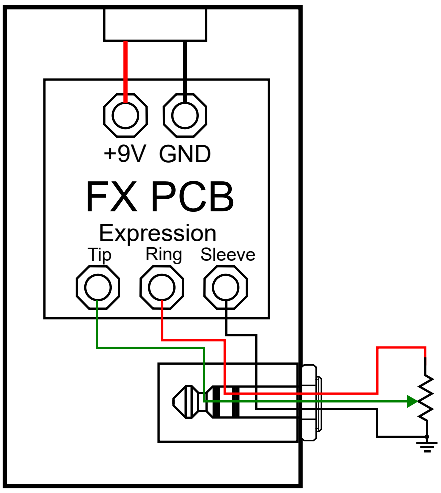
TRS can be configured as *Normally Open* and *Normally Closed*. With *Normally Open*, the two lines are open at system start and pulses are executed as Open → Closed → Open. With *Normally Closed*, the lines are closed to GND at system start and pulses are executed as Closed → Open → Closed.

Installation for Role Pot/CV (Expression)

Following is explained how to wire the Goblin-SPST for a single expression input without further explaining how it works. For that refer to the next chapter which is very long and technically.

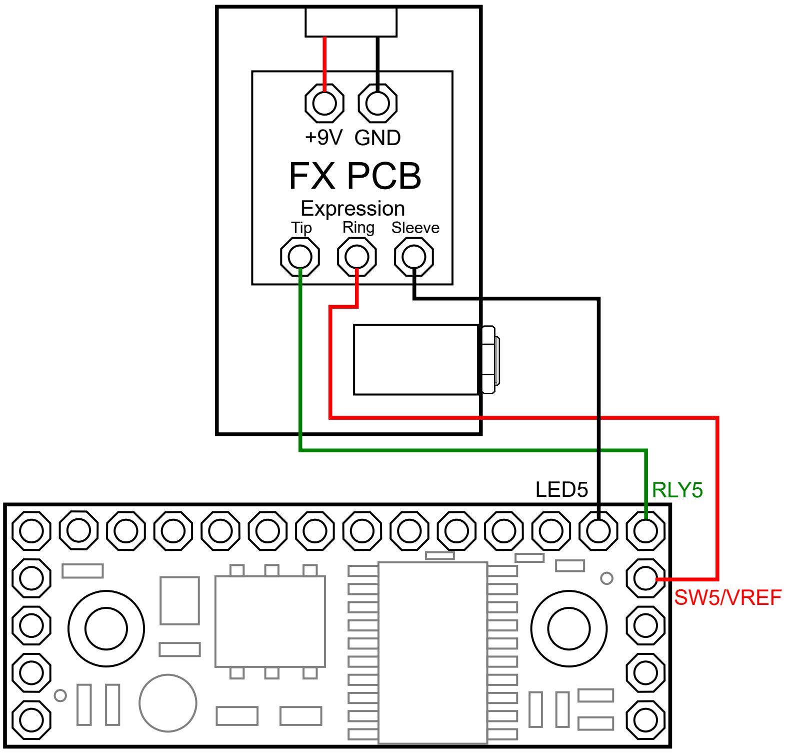
So, for a single expression input and only if you don't use any other ports with the role *Pot/CV,* use port 5 for expression.

This is how the expression port looks before installation, with an expression pedal (represented by the potentiometer) plugged in.



Effects device with expression Input

Connect the Goblin directly to the pins of the expression socket. Use a dummy jack plug, so the pedal thinks, an expression pedal is plugged in.



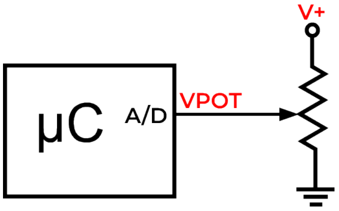
Effects device with expression input and installed Goblin-SPST

The function of LED5 in this case, is to tell the effects device which position is its default position at startup. If that's not right for your case, try to connect it to SW5/VREF.

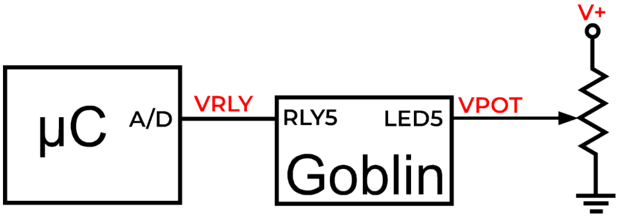
Installation for Role Pot/CV (Potentiometer)

I wish this was easy. But it isn't. To understand how the ***Pot/CV*** function works, we have to go through some basics.

First, let's have a look how the role *Pot/CV* works theoretically. The role *Pot/CV* generates a voltage dependent on either a voltage input, or a MIDI command. So, for example, you configure port 5 for role *Pot/CV* and apply 2.5V on LED5 you'll get 2.5V output on RLY5. Apply 5V on LED5 and you'll get 5V output on RLY5. Now, let's say, with 5V on LED5 you send MIDI command CC 90 00 (CC90 is the command for port 5) then the output on RLY5 will be 0V, MIDI command CC 90 127 and you get 5V on RLY5. When you now change the voltage on LED5, RLY5 will again change it's voltage to the voltage on LED5, overriding previous MIDI commands.



OK, that's great, but what does this have to do with my guitar pedal? Well, let's have a look how potentiometers on digital pedals work. On the right side we see a typical system. There is a voltage V+ applied to one side of a potentiometer with the other side tied to GND. When the potentiometer is turned, the voltage VPOT changes from anywhere between 0V to V+. The Microcontroller uses its ADC (Analog to Digital Converter) and convert this voltage to a digital value which it uses to change whatever value is tied to the potentiometer, like volume for example.



We can now use the above explained behaviour of the Goblin to mimic the behaviour of the potentiometer without losing the actual potentiometer. Let's see what happens, when we wire it like in the picture on the right. As long as no MIDI commands are sent, the voltage on RLY5 is equal to the voltage on LED5. The Goblin is transparent, so to say. VRLY = VPOT. The effects device can be operated as before. If we now send a MIDI command, VRLY changes independently from VPOT. The microcontroller doesn't know anything about Goblins and all and thinks, the potentiometer has been turned and changes its internal value. If the potentiometer is turned and VPOT changes, the MIDI command is overridden and again VRLY = VPOT. So, this gives us the ability to use the potentiometer AND control the effects device with MIDI.

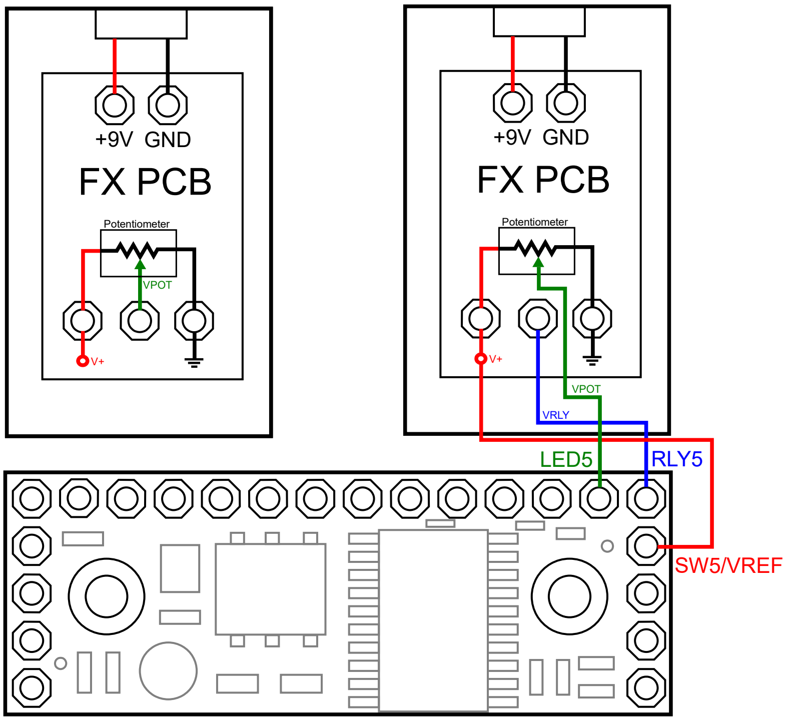
One last problem is there, though. How much is V+? Effects devices run on different voltages. The Goblin runs on 5V and the EHX Pitch Fork for example runs on 3.3V. This is not so much a problem, when operating the effects device with the potentiometer, since VRLY=VPOT. But what happens when I send CC 90 127. The Goblin will go to maximum and that's 5V and the Pitch Fork won't be happy about that. Because of that, there is the possibility to tell the Goblin what the *Reference Voltage* is. That means the voltage that is maximum. To achieve that, this reference voltage is applied to a special connection: *SW5/VREF.* As soon as one of the ports is used in the role *Pot/CV*, SW5 is used as the reference voltage **for all Pot/CV ports.**

But, believe it or not, there are mixed systems. The EHX Pitch Fork uses 3.3V for its potentiometers and 5V for the expression port. For that reason every Pot/CV can be configured to use the internal supply voltage of the Goblin (5V) as reference. See chapter Configuration.

Let's see how this looks in reality. On the left there is the FX PCB with a potentiometer. We see three connections of the potentiometer and how they are wired in a real digital pedal. Again we see the voltages V+ and VPOT.

On the right we see how the Goblin is wired for that case. The middle leg of the potentiometer is removed from the connection to the PCB and instead wired to the Goblin's LED5 connection. Where the middle leg was connected to the PCB before, there is now RLY5 connected. The reference voltage V+ is connected to SW5/VREF. Note that this voltage is used for all Pot/CV ports. Therefore the connection has to be made only once.

The action of removing the middle leg of the potentiometer from the PCB is difficult. Most of the time it is necessary to desolder the POT, cut off the middle leg and solder it back in.



Configuration

In order to be able to adapt the Goblin to as many effects devices as possible, various parameters can be set via the configuration procedure. **The configuration is always carried out via MIDI CC messages and consist of 4 messages per parameter.** The actual configuration message, two passcode messages and a save message. The configuration messages are described below. For passcode and save message, see chapter Saving the Configuration.

Roles

The role of a port describes the function that the Goblin performs. For a detailed explanation, see chapter Ports and Roles.

CC	#	Function
Port 1: 27 **Port 2: 47** **Port 3: 67** **Port 4: 87** **Port 5: 107**	0 1 2 3 4 5 6 7	Disabled (Default setting port 2...5) N.A. Switch (Default setting port 1) Tap Tempo (Like Switch, but w/o defined state) TRS Normally Open TRS Normally Closed N.A. Pot/CV

LED Polarity

In the S*witch* role, the polarity must be set according to the measured values from chapter Status monitoring connection (LED).

- Low Active: Voltage below the LED threshold is recognized as "On".
- High Active: Voltage above the LED threshold is recognized as "On".

In the example above, the LED polarity is “High Active” because when the effects device is active, the voltage is high.

CC	#	Function
Port 1: 28 **Port 2: 48** **Port 3: 68** **Port 4: 88** **Port 5: 108**	0 1	Low Active High Active (Default setting)

LED Threshold

The Goblin monitors the voltage on the active side of the LED for the S*witch* role. Depending on whether the threshold is exceeded or not, the status is recognized as "on" or "off". Since LED voltages can vary greatly, the threshold can be set for each port.

The Threshold must be set according to the measured values in chapter Status monitoring connection (LED). Set it to half way between the on and off value. In the example above the LED threshold would be half way between 0V and 1.8V, that’d be 0.9V.

CC	#	Function
Port 1: 29 **Port 2: 49** **Port 3: 69** **Port 4: 89** **Port 5: 109**	n	LED threshold in 0.05V steps. Default is 24 (=1.2V)

MIDI Channel via MIDI command

The MIDI channel can easily be set with one of the footswitches. If that’s not possible, it can be set with MIDI commands.

CC	#	Function
6	0	Set MIDI channel to Omni (reacts to every channel)
6	1...16	Set MIDI channel to channel 1...16

Pulse Length for Role Switch/Tap Tempo and TRS

To emulate the press of a switch, the Goblin pulls the RLY lines to GND for a brief moment. What “brief” means can be set with this command. In 10ms steps. Default is 8 (80ms).

CC	#	Function
7	0...127	Pulse length for switch, tap tempo and TRS from 0ms to 1.27s

Startup Delay

Some effects devices need some time to power up. This ranges from a few milliseconds to several seconds. So the Goblin can correctly restore the last state and the boot process of the effects device is not disturbed, a delay at the beginning can be useful.

CC	#	Function
8	n	Startup delay n*100 ms (Default n = 5)

Please note that a pressed button is immediately processed as soon as the Goblin is connected to the supply voltage, regardless of the startup delay, but depending on the polarities set.

Sensitivity of the MIDI Clock Detection

For commands that use automatic MIDI Clock detection, the sensitivity for resending the pulses can be set or the feature can be deactivated entirely. Raising the sensitivity is only necessary for systems with very high MIDI clock jitter.

CC	#	Function
5	0-16	Sensitivity (default = 12) 0 = Deactivated 1 = Very low sensitivity 16 = Very High sensitivity

Configuration bits

The following configuration bits are only necessary in extremely exceptional cases. They allow more in-depth configuration. Each port has a set of configuration bits. Some settings are available as single commands, too. The following 5 parameters are set with a 7-bit word:

CC	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2-0
	POL-LED	FX-DRIV	POL-FX	POL-SW	ROLE
Port 1: 19 **Port 2: 39** **Port 3: 59** **Port 4: 79** **Port 5: 99**	Polarity of the LED 0 = Low Active 1 = High Active (*)	Driver function of port 0 = Push Pull 1 = Open Drain (*)	If Role Pot/CV: Direction of the Pot 0 = Pot direction inverted 1 = Pot direction non inv. (*) Any other Role: Polarity of FX 0 = Normally Closed 1 = Normally Open (*)	If Role Pot/CV: VREF 0 = Override VREF with 5V 1 = VREF = SW5 (*) If role Switch/Tap Tempo: Polarity of Switch 0 = Normally Closed 1 = Normally Open (*)	Role of the port 0 (000): Disabled 2 (010): Switch (*) 3 (011): Tap Tempo 4 (100): TRS Normally Open 5 (101): TRS Normally Closed 7 (111): Pot/CV

(*) = Default setting

- **ROLE:** The role of the port. See Roles
- **POL-SW**: If role Pot/CV: Source of VREF. Wether SW5 or internal 5V is used as VREF for CV.
  If role Switch/Tap Tempo: Switch polarity. *Normally open* switches are used most of the time.
- **POL-FX**: If role Pot/CV: Wether the voltage is falling or rising with rising MIDI CC values.
  If role Switch/Tap Tempo: Polarity of the FX. Usually this value is set to the same value as *POL-SW*.
- **FX-DRIV**: The FX side (connection RLY) usually expects a positive voltage of 5V or lower, that is briefly pulled to GND in order to switch the effect. If this is not the case (e.g. OBNE Dark Star) the Goblin can drive the + 5V high state itself. To do this, set this bit to *Push Pull*. Only for role *Switch* and *Tap Tempo.*
- **POL-LED**: The LED Polarity. See LED Polarity

It should be noted that these settings must always be made together. The entire word is always used.

There is a spreadsheet to help with calculation of the configuration bits and LED threshold.

Preconfigured Mods for Popular Pedals/Factory Reset

There are some popular pedals that the Goblin knows. The complete configuration for these pedals can be set with a single configuration command, or it can be used for a factory reset. In some cases it even has some special functions implemented.

CC	#	Function
4	0 1 2 3 4	Factory Reset (Default) Sets all ports to role *Switch* Universal Audio pedals Electro-Harmonix Pitch Fork Electro-Harmonix NanoPOG (Coming Soon)

Saving the Configuration

In order to save the above configurations, the following three commands must be called immediately one after the other. If another command is sent in between, the saving sequence is aborted. The effects device must be restarted for the changes to take effect.

CC	**#**	**Function**
9	18	1. passcode message for the saving sequence
9	52	2. passcode message for the saving sequence
9		Save messages:
	0	Saving the **Startup delay**
	1	Saving the **role, led polarity and configuration bits** of port 1
	2	Saving the **role, led polarity and configuration bits** of port 2
	3	Saving the **role, led polarity and configuration bits** of port 3
	4	Saving the **role, led polarity and configuration bits** of port 4
	5	Saving the **role, led polarity and configuration bits** of port 5
	6	Saving the **LED threshold** of port 1
	7	Saving the **LED threshold** of port 2
	8	Saving the **LED threshold** of port 3
	9	Saving the **LED threshold** of port 4
	10	Saving the **LED threshold** of port 5
	11	Saving the **MIDI channel** from command CC 06
	12	Saving the Sensitivity of the **MIDI Clock detection**
	13	Saving the **Pulse Length**
	14	Saving the **Preconfigured Mod**

Example 1: To set an LED threshold of 1.8V for port 2, the following four commands are sent back-to-back:
CC 49 36 -> CC 09 18 -> CC 09 52 -> CC 09 07

Example 2: Set the Goblin to factory reset:
CC 04 00 -> CC 09 18 -> CC 09 52 -> CC 09 14

Dual Color LEDs

If the indicator LED is dual color, the LED inputs of port 2 and port 3 can be used together on port 2. One color is then connected to *LED2* and the other to *LED3*. To activate this feature, the role of **port 3 must be *Disabled*** and the **LED threshold value of port 3 must be set to 0**.

Recommended installation order

Here's a short guide how to get started with the installation:

1. Disassemble the effects device, find the positive and negative pins on the supply voltage and connect the V+ and GND connections to it.
2. Measure the connections of the foot switches and find the "active side" i.e. the pin of the switch that changes its voltage when the switch is pressed. Desolder the wire from the active side of the foot switch. Connect the loose end to RLY and connect the free pin of the foot switch to SW. The effects device should work now as before.
3. Measure both pins of the LED in the on and off state. Find the pin, that changes it's voltage when changing state, connect the LED connection of the Goblin there. Determine LED polarity and LED threshold like in chapter ***Status monitoring connection (LED)***.
4. Connect the MIDI sockets. Test if MIDI Thru works. Send a toggle command (CC 10 02) and check if there is any reaction. MIDI has to work before proceeding to the next step.
5. With the collected data figure out the configuration (if necessary) and send the configuration commands. Test.
6. Mount the Goblin and assemble the pedal.

Troubleshooting

The most common mistake during installation is the wrong configuration of the LED threshold value and/or the LED polarity. If this is not set correctly, the effect can be operated with the switch, but via MIDI the on and off command doesn't work properly. In this case, check the configuration and repeat the measurement if necessary. Also make sure that you have measured against GND!

To help with the configuration you can use the spreadsheet at:
https://oscillatordevices.com/doc/oscillator_devices_configuration_helper.xlsx

Usage

MIDI Channel

The ***Goblin*** comes in omni mode (I.e. it responds to every channel). Please keep it that way until you have finished installation, because it takes the MIDI channel out of the equation, when configuring and testing. To change the MIDI channel, proceed as follows

1. Disconnect the device from the power supply
2. Press the button of port 1 and restore the power supply while it is pressed. The device starts to flash its LED after the startup delay has elapsed.
3. Press the button according to the number of the desired channel (e.g. twice for channel 2). The Goblin acknowledges this by emitting short flashing impulses according to the number of the channel.
4. Once the desired channel is set, press the button and hold it down until the Goblin switches off completely.
5. Disconnect supply voltage. The next time the Goblin is started, it reacts to the selected MIDI channel.

To put the ***Goblin*** in omni mode skip step 3.

Presets

The Goblin-SPST-2.0 is able to save up to 20 presets. Saving a preset always saves the current state. That includes Pot/CV value either set by the Pot or MIDI command, state of the switches and state of the TRS ports. To save a preset, set the Goblin-SPST into *preset save mode* first. When the Goblin is in *preset save mode* the LED flashes.

CC	#	Function
119	127	Enter preset save mode
	0	Leaving preset save mode

When in *preset save mode* simply set all the controls how you like it and then send the PC command you want the preset to be saved to. Repeat until done. To leave *preset save mode* just send the MIDI command above, or disconnect from the power supply. To call up the preset just send the PC command it was saved to.

MIDI Commands for Role Switch and Tap Tempo

The ports can be controlled individually. “Hold”, or MIDI clock synchronous commands, are interrupted by pressing the switch.

CC Port 1	CC Port 2	CC Port 3	CC Port 4	CC Port 5	#	Function		#	Function
10	**30**	**50**	**70**	**90**	00	Port off		15	Toggle to the beat of the MIDI clock in 1/32 notes
					01	Port on		16	Toggle to the beat of the MIDI clock in 1/2 notes
					02	Port toggle (e.g. Tap Tempo)		17	Toggle to the beat of the MIDI clock in whole notes
					03	Port hold (Corresponds to a pressed and held switch)		18	Toggle to the beat of the MIDI clock every 2nd whole note
					04	Port release (Release held switch)		19	Toggle to the beat of the MIDI clock every 3rd whole note
					10	Toggle to the beat of the MIDI clock in 1/4 notes		20	Toggle to the beat of the MIDI clock every 4th whole note
					11	Toggle to the beat of the MIDI clock in 1/8 notes		21	Toggle to the beat of the MIDI clock every 5th whole note
					12	Toggle to the beat of the MIDI clock in triplet notes		22	Toggle to the beat of the MIDI clock every 6th whole note
					13	Toggle to the beat of the MIDI clock in 1/16 notes		23	Toggle to the beat of the MIDI clock every 7th whole note
					14	Toggle to the beat of the MIDI clock in dotted 1/8 notes		24	Toggle to the beat of the MIDI clock every 8th whole note

Some effects devices react strangely when a tap tempo signal is constantly being sent. There is the possibility to only give a limited number of impulses. Enough for the effect to take over the beat. With these commands, an automatic MIDI clock detection is carried out. If the MIDI clock changes by more than approx. 1%, the pulses are sent again. The sensitivity of the MIDI clock detection can be configured (see chapter Sensitivity of the MIDI Clock Detection).

CC Port1	CC Port2	CC Port3	CC Port 4	CC Port 5	#	Function		#	Function
11	**31**	**51**	**71**	**91**	0-19	1-20 times toggle to MIDI clock in 1/4 notes		100-104	1-5 times toggle to MIDI clock 1/2 notes
					20-39	1-20 times toggle to MIDI clock in 1/8 notes		105-109	1-5 times toggle to MIDI clock every whole note
					40-59	1-20 times toggle to MIDI clock in triplet notes		110-114	1-5 times toggle to MIDI clock every 2nd whole note
					60-79	1-20 times toggle to MIDI clock in 1/16 notes		115-119	1-5 times toggle to MIDI clock every 4th whole note
					80-99	1-20 times toggle to MIDI clock in dot. 1/8th note		120-124	1-5 times toggle to MIDI clock every 8th whole note

MIDI Commands for Role TRS

If the port is in the role of an external switch (TRS, EXT, CTL etc.), this port then has two lines, referred to as "Tip" and "Ring", which emulates a plugged in external switch. This switch can be *Normally Open (NO)* or *Normally Closed (NC),* which is set during configuration (see chapter Roles). This table is valid when wiring *Tip* to *LED* and *Ring* to *RLY*.

Line-specific Commands

CC Tip	CC Ring	#	Function	#	Function
Port2: 30 Port3: 50 Port4: 70 Port5: 90	**Port2: 40 Port3: 60 Port4: 80 Port5: 100**	00	**Set „Open“**		
		01	**Set „Closed“**		
		02	Single pulse		
		03	Toggle		
		10	Pulse MIDI clock 1/4	30	Toggle MIDI clock 1/4
		11	Pulse MIDI clock 1/8	31	Toggle MIDI clock 1/8
		12	Pulse MIDI clock triplets	32	Toggle MIDI clock triplets
		13	Pulse MIDI clock 1/16	33	Toggle MIDI clock 1/16
		14	Pulse MIDI clock dotted 1/8	34	Toggle MIDI clock dotted 1/8
		15	Pulse MIDI clock 1/32	35	Toggle MIDI clock 1/32
		16	Pulse MIDI clock 1/2	36	Toggle MIDI clock 1/2
		17	Pulse MIDI clk every whole note	37	Toggle MIDI clock every whole note
		18	Pulse MIDI clk every 2nd whole note	38	Toggle MIDI clock every 2nd note
		19	Pulse MIDI clk every 3rd whole note	39	Toggle MIDI clock every 3rd note
		20	Pulse MIDI clk every 4th whole note	40	Toggle MIDI clock every 4th note
		21	Pulse MIDI clk every 5th whole note	41	Toggle MIDI clock every 5th note
		22	Pulse MIDI clk every 6th whole note	42	Toggle MIDI clock every 6th note
		23	Pulse MIDI clk every 7th whole note	43	Toggle MIDI clock every 7th note
		24	Pulse MIDI clk every 8th whole note	44	Toggle MIDI clock every 8th note

Pulse

It is also possible to send a certain number of pulses, e.g. to select a preset.

CC-Tip	CC-Ring	#	Function
Port2: 31 Port3: 51 Port4: 71 Port5: 91	**Port2: 41 Port3: 61 Port4: 81 Port5: 101**	0	1 pulse
		1	2 pulses
		n	n+1 pulses
		127	128 pulses

MIDI Clock Pulse

Some Tap Tempo effects react strangely when the Tap Tempo Pulse is sent continuously. It is therefore possible to only send a limited number of pulses until the effects device has recognized the tempo. With these commands, an automatic MIDI clock detection is carried out. If the MIDI clock changes by more than approx. 1%, the pulses are sent again automatically. The sensitivity of the MIDI clock detection can be configured (see chapter Sensitivity of the MIDI Clock Detection).

CC-Tip	CC-Ring	#	Function	#	Function
Port2: 32 Port3: 52 Port4: 72 Port5: 92	**Port2: 42 Port3: 62 Port4: 82 Port5: 102**	0-19	1-20 pulses MIDI clock 1/4 notes	100-104	1-5 pulses MIDI clock 1/2 notes
		20-39	1-20 pulses MIDI clock 1/8 notes	105-109	1-5 pulses MIDI clock every whole note
		40-59	1-20 pulses MIDI clock triplet notes	110-114	1-5 pulses MIDI clock every 2nd whole note
		60-79	1-20 pulses MIDI clock 1/16 notes	115-119	1-5 pulses MIDI clock every 4th whole note
		80-99	1-20 pulses MIDI clock dotted 1/8 notes	120-124	1-5 pulses MIDI clock every 8th whole note

Pulse Length

The standard length of a pulse is approx. 80 ms. If this is too short for some devices, the pulse length can be set in 10 ms steps.

CC-Tip	CC-Ring	#	Function
Port2: 35 Port3: 55 Port4: 75 Port5: 95	**Port2: 45 Port3: 65 Port4: 85 Poert5: 105**	n	Pulse length in 10ms steps

MIDI Commands for Role Pot/CV

The voltage of the port can set with this command.

CC	#	Function
Port2: 30 **Port3: 50** **Port4: 70** **Port5: 90**	n	Voltage from 0v to VREF in 127 steps. 0 = 0V, 127 = VREF with non inverted pot direction 127 = 0V, 0 = VREF with inverted pot direction

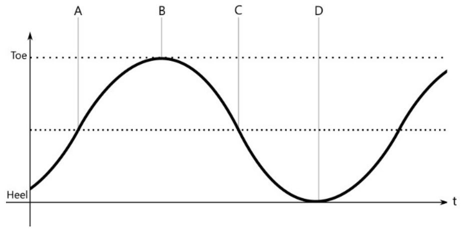
LFO-Waveforms

In addition, the ***Goblin-SPST*** starting with version 2.0.1 has an internal, MIDI clock synchronous, LFO engine with 3 waveforms to drive the Pot/CV. The LFO is stopped, as soon as there is a voltage change on the LED input (i.e. the potentiometer is turned).

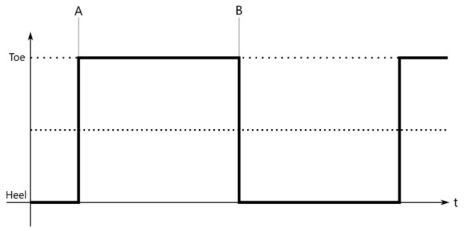
CC	#	Function
Port2: 36 **Port3: 56** **Port4: 76** **Port5: 96**	0	Stop LFO
	1	Restart LFO
	10	LFO sine wave with starting point A
	20	LFO sine wave with starting point B
	30	LFO sine wave with starting point C
	40	LFO sine wave with starting point D

CC	#	Function
Port2: 36 **Port3: 56** **Port4: 76** **Port5: 96**	50	LFO rectangle wave with starting point A
	60	LFO rectangle wave with starting point B
	90	LFO ramp forward with starting point A
	100	LFO ramp forward with starting point B
	110	LFO ramp backward with starting point C
	120	LFO ramp backward with starting point D

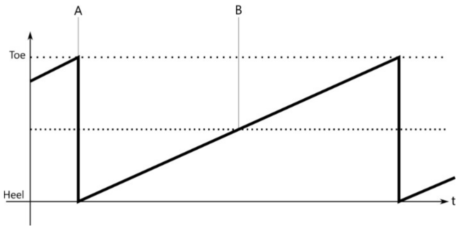
LFO Sine wave



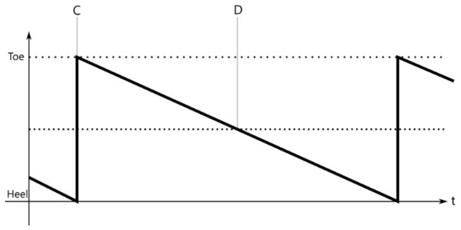
LFO Rectangle wave



LFO Ramp wave forward



LFO Ramp wave backward



The LFO starts immediately at the specified start point. Resending the commands will reset the engine to start over from the starting point.

LFO-Speed

The speed of the LFO relative to MIDI clock can be increased or decreased. The commands above perform one pass of the waveform per 1/4 note. The commands to change the speed are determined by adding to the basic command (CC 36 10, CC 36 20 etc.).

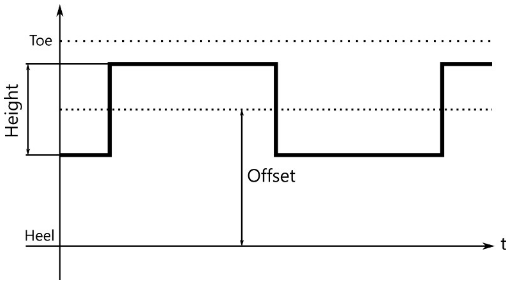
- Basic Command: Normal speed
- Basic Command+1: Half speed
- Basic Command+2: Double speed
- Basic Command+3: Quarter speed
- Basic Command+4: Quadruple speed

For example:

CC	#	Function
Port2: 36 **Port3: 56** **Port4: 76** **Port5: 96**	10	LFO sine wave with starting point A, speed normal (1/4 Note)
	11	LFO sine wave with starting point A, half speed (1/2 Note)
	12	LFO sine wave with starting point A, double speed (1/8 Note)
	13	LFO sine wave with starting point A, quarter speed (Ganze Note)
	14	LFO sine wave with starting point A, quadruple speed (1/16 Note)

This procedure can be applied to all waveforms and starting points. E.g. rectangle with starting point B in 1/8 notes on port 2: CC 36 62.

LFO-Parameter



The waveforms set this way always run through the entire range, from heel to toe. The range can be reduced and moved.

The waveforms are shifted in 13 steps, with step 6 being the waveform in the middle. This corresponds to the *Offset* in the graphic.

CC	#	Funktion
Port2: 37 **Port3: 57** **Port4: 77** **Port5: 97**	00	Offset 0. The middle of the waveform is at Heel
	60	Offset middle: The middle of the waveform is right between the heel and toe (default)
	120	Offset Max: The middle of the waveform is at Toe

All values in increments of 10 are valid.

- Basic Command: Normal Height
- Base Command+1: Half Height
- Base Command+2: Quarter Height
- Base Command+7: Height 1/128

For example:

CC	#	Funktion
Port2: 37 **Port3: 57** **Port4: 77** **Port5: 97**	60	No constraint on height, waveform sweeps full range (default)
	61	Waveform height halved
	62	Waveform height 1/4
	63	Waveform height 1/8
	...	
	67	Waveform height 1/128

The *Offset* and *Height* settings are global, so they persist across a new waveform. Reset with CC 37 60 for port 2.