

Sync Latch

Mutable Instruments MIDIpal app ‘sync latch’ re-conceived for the **Expert Sleepers** Eurorack module *Disting NT*, and implemented via Lua script. Same minimal vibe. Cool new features.

This Lua script was inspired by the *MIDIpal* app ‘sync latch’ which was apparently prior inspired by the **Mungo Sync**, which allows a slave device to be synchronized and started/stopped in sync with a master device playing a loop.

Version 2 enhancements:

- Option to use Run/Start signal instead of Reset pulse
- New **Reset Out** signal; which can pulse at Start/Reset or (if in Run/Stop mode) at Stop or (Start+Stop) This is particularly useful for certain sequencers like **ALM Busy Circuits’ ASQ-1**
- When **Slave State at Reset** is set to ‘Idle’ a pending Arm Latch signal will force behavior to the same as setting parameter to ‘Run.’ Simple tactile performance / quality of life feature.

From the original *MIDIpal* manual :

“Consider the following situation: you have a drum machine playing a steady 2-bar loop, in 4/4, and generating a clock signal for a slave MIDI sequencer. You need to reload a pattern on the sequencer. You stop the sequencer, load the pattern, and then you need to hit the play button on the sequencer at the exact right moment so that it starts spot on the first beat of the 2-bar drum loop. Tricky isn't it? This is where the *MIDIpal* can be used. Insert it between the drum machine and the sequencer. Configure the duration/time signature of the loop (Here: 8 quarter notes). The *MIDIpal* will keep counting beats. When you press the *MIDIpal* encoder, it'll wait for the beginning of the loop and then send a start message to the MIDI slave. When you press again the *MIDIpal* encoder, it'll wait for the end of the loop and then send a stop message to the MIDI slave.”

MIDIpal’s brilliant, minimal user interface and vibe has been lovingly retained, with a few enhancement and features. Consider the simple status display:

01:01:00 .

This counter shows **Bar:Beat:Tick**. The clock tick count resolution is determined by the Clock PPQN parameter. (For example, when set to the default of 24 PPQN, the tick count runs from 0 to 23. for 4/4 time signature)

Slave transport status: a dot “.” is shown when the slave is stopped ; a “>” sign is shown when the slave is running. This reflects the status of this Lua algorithm’a **Slave Run** output signal.

When you press the *Disting NT*’s **Encoder no. 2** (Right encoder), the beat count is shown in brackets, to indicate that the Sync Latch is waiting for the beginning of the appropriate bar for slave state change. Optionally, the “Arm / Disarm” toggle can also be commanded via the **Arm input** signal.

For example, with the **Loop Bars** parameter set to 4:

[04:04:00] .

Indicates that the slave is currently stopped, and that Sync Latch is waiting for the beginning of the next bar to raise the **Slave Run** signal (or send a MIDI Start message, when routed to and 2nd subordinate Disting NT Clock with “send to breakout” parameter set.)

[03:02:12]>

Indicates that the slave is currently running, and that Sync Latch is waiting for the end of the loop length to send a Stop message.

Automatic and manual Fill-in features.

When the **Auto Fill** parameter is **On**, Sync Latch takes over the **Fill Gate** output signal - forced High - for a fixed number of beats (set with **Auto Fill Beats** parameter) leading up to the latch transition (Run or Stop). Note that the **Fill gate** is always automatically cleared (signal Low) at the End of Loop. The status of the Fill signal is indicated in the display as an exclamation mark “!” instead of the usual colon “:” as below

[04!04!23]>

This gate signal can be used to command a master-synced sequencer to switch its lane/channel for example an alternate drum pattern. Creative lead-in or lead out music.

The next master clock tick for the example (with four-bar loop setting) above will change state to:

01:01:00 .

In the case of *manual* Fill, the corresponding gate signal is cleared at the end of current bar. Think of it as a brief “Fill to the end of bar.” Beside the Fill input signal, this manual fill behavior can be alternately commanded by pushing *Disting NT*’s **Pot3** (Module’s Right-most Pot on panel.)

Inputs –

- Clock – connect master clock, minimum 24 PPQN required!

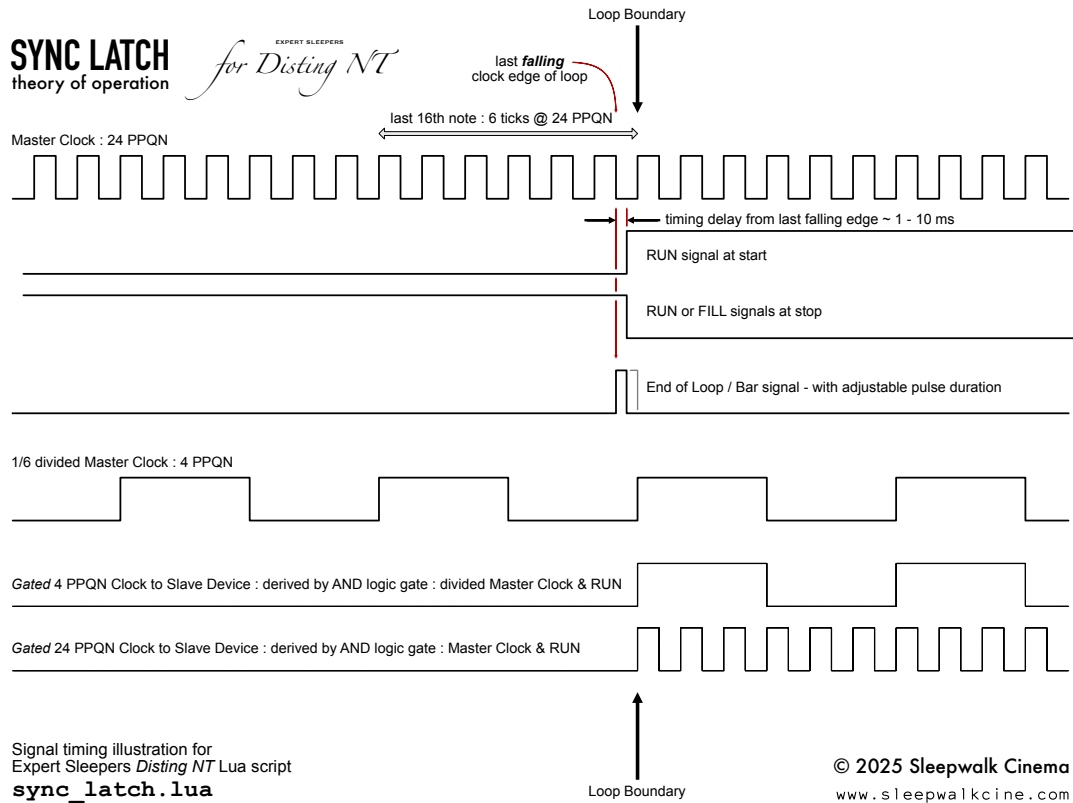
Rising edge triggers:

- Reset – reset the internal state, Latch and Slave Run state to stopped.
- Arm – Arm the Latch, for either Slave’s Run or Stop
- Fill – new feature. Immediate manual

Outputs –

- Slave Run gate – signal to Run / Stop slave device or secondary clock
- Fill gate – Fill signal
- End of Bar trig
- End of Loop trig

The **End of Bar** and **End of Loop** signals are provide Eurorack users needing to design custom logic or deal with external gear's various Reset behaviors.
 For example **ALM Busy Circuits' ASQ-1** sequencer requires that its **Reset** input is triggered **PRIOR to first Clock tick!** Otherwise there is sequencer lag, i.e., poor sync.
 Please see the timing illustration below.



A note regarding Synch Latch's **Makeup Reset Clock Tick** parameter; this is set to On by default, as it is presumed that most operators will use a *Disting NT* Clock algorithm's **Clock** and **Reset** outputs (the 'Master Clock') as to inputs for this Lua program. Since the NT algorithm simultaneously sends First Tick and Reset signal. Devices and their manufactures have different methods and thinking around Reset signal timing. Sorry, you'll have to be your own detective on in this matter and experiment. Some devices like **Reset** prior to first clock tick. If you're using an external analog clock/reset sources and assume this behavior, then this parameter can be set to Off.

Happy Synchronizing!

- Michael Lauter / Sleepwalk Cinema

© 20025 Sleepwalk Cinema <http://www.sleepwalkcine.com>

This software is offered with no warranties.

Manual vers. 2

Bug reports are welcome, but for actual fixes please send me a gift via PayPal.

paypal.me/kinotonik

Sleepwalk Cinema is the moniker of filmmaker Michael Lauter, based in Denver, Colorado USA