# GOLT !

RED LIGHT DISTRICT

USER MANUAL v2.1

# How RLD Works

## Signal Flow

TIMING PROCESSOR

The timing processor receives all clock signals and start stop commands. Its main function is to determine which timing signals to obey (MIDI, Internal Clock or External Cock) and then distribute the correct signals to the 32 step sequencer.

Secondary functions include synchronising all clock dividers to the current clock signal, and switching the MIDI ports between MIDI OUT and MIDI THRU.

Transmitted to 32 step sequencer:
- Run Signal
- 24 pulses per quarter note Clock Signal (Din Sync and MIDI standard)

32 STEP TIMING SEQUENCER

The 32 step sequencer controls the timing of the notes.
It first applies a divide ratio to the selected clock and the result is used to control the speed of the sequence.
A pattern is either 16 or 32 steps long and may take either 1, 2 or 4 MIDI bars to complete; at which time the pattern restarts automatically in time with the start of the next occurring MIDI bar. The sequence can run in either forward or reverse, and when in 16 step mode two patterns are available (A & B).
Its primary function is to send Note On and Note Off triggers to the 12 Step sequencer.

Transmitted to 12 step sequencer:
- Note On trigger
- Note Off trigger
- Reset trigger (auto)
- Mute gate (auto)

12 STEP CV SEQUENCER

The 12 step sequencer controls the voltages that are applied to the CVs and Gates.
What makes it unique is its ability to kill the voltage on a step without immediately moving to the next step. This means each step has its own note on and note off times (the space between the last Note Off and the next Note On equates to a “rest”). This makes it equivalent to a 24 step analogue sequencer with 12 notes and 12 rests.

When a Note On is received the sequence moves to the next step and voltage is applied. When a Note Off is received the voltage to the current step is killed. This kills all gates and *optionally* kills all CVs (switch HOLD to ON to keep a CV at its last known value between notes, useful to hold the pitch stable during the release phase of an ADSR).

The 12 step sequence can run in 8 different modes (forward, reverse, alternate, mute, skip, voltage addressed stepping, brownian random and last step only). The sequence is reset at the end of the pattern unless the Reset signal (from the 32 step sequencer) is blocked (switch EOP Reset to Block).

# RLD Inputs & Outputs

## TIMING PROCESSOR

**CLOCK IN**

Positive edge triggered input.

Converts an external clock signal into the 24 pulses per quarter note Din Sync / MIDI standard. Reverse voltage protected. Over voltage protected (clamping occurs at -0.6V and +5.6V). Over current protected. (From this point onwards, these protections are referred to as “fully protected input”.)

**CLOCK DIVIDERS**

Six clock divider outputs with 50% pulse width.

Positive leading edge synchronised to MIDI and the 32 step sequencer. 220 ohms output impedance. Short circuit current limited (16mA).

**MIDI IN**

Interprets MIDI clock only. Everything else is ignored.

**MIDI OUT/THRU**

Auto switching between OUT & THRU. The second jack can be reconfigured as a Din Sync output. See the DIY manual for details.

## 32 STEP TIMING SEQUENCER

**PATTERN** VC

Only active in 16 step mode. Plugging in causes the Pattern A/B switch to be disabled. 0V = Pattern A, +5V = Pattern B. Fully protected input.

**DIRECTION** VC

Active in all modes. Plugging in causes the Direction switch to be disabled. 0V = Reverse, +2 to +3V (approx) = Alternate, +5V = Forward. Fully protected input.

**PAUSE**

Positive edge triggered input.

Causes the 32 step sequencer to stop at the current step and wait until the voltage is returned to 0V. Operation then continues as normal. The warning sign under the jack indicates that using this option will cause the sequencer to lose sync with with the start of the MIDI bar. Fully protected input.

**RESET**

A positive edge triggered input.

Causes the 32 step sequencer to reset to step 1 when the next step to due to occur. This is not an immediate reset, the sequencer may delay the reset until the last Din Sync / MIDI clock pulse before the next step is received. Using waveforms from your modular as the reset trigger may cause the sequencer to lose time with MIDI (hence the warning symbol). But when the source waveform is derived from or synchronised to the RLD clock dividers the result usually stays in time with MIDI. Fully protected input.

12 STEP CV SEQUENCER

**VAS**

Voltage Addressed Stepping Input.

If VAS mode is selected for the 12 step sequencer this input becomes active. If nothing is plugged in, the input is pulled down to 0V and the 12 step sequencer will continue to retrigger the first step. When a voltage is applied it will be used to select the next step in the sequence. Higher voltages select higher steps, lower voltages select lower steps. Active input range is 0V to 5V. Higher voltages are OK but will only register as if they were +5V. Fully protected input.

**GATE OUTS**

These are the gate outs for the sequencer.

Usually between 8.5V to 9.1V but can be changed (chosen at build time, see DIY manual). Output short circuit is current limited (40mA). Strongly pulled down between pulses to allow triggering of modules that emit voltage from their inputs (Malekko Switch for example).

**CV OUTS**

These are the control voltage outputs for the sequencer.

Variable between 0V and the gate voltage (typically 8.7V). CV 1 also has a fine tune control which is approximately 10% of the range of the course tune. Short circuit protected (45mA). Output is enough to fan-out to a few oscillators, more than 3 or 4 and you may need a buffer (this is heavily dependent on the input impedance of the oscillator which can vary quite a bit).

**KILL SLEW**

Positive voltage input to this jack will remove the slew on CV 1 between steps. Effect is immediate but the input voltage must approximately equal to, or higher than, the CV 1 level. If the KILL SLEW input is much lower than CV 1 the slew may only be reduced not removed completely.

# RLD Switches

TIMING PROCESSOR

**CLOCK** switch - **MIDI** or **INT/EXT**

Switches the between the Internal/External clock and MIDI as the synchronisation source.

The Internal clock is normalled through the Ext Clock In Jack. Plugging a cable into this jack disables the internal clock and allows any external square wave source to act as the clock source. If you input a clock that is running faster than 230-240Hz the sequencer will start to drop clock pulses, resulting in slower running and uneven timing across the pattern. Mind you, this is approximately 600bpm at 24 pulses per quarter note MIDI/Din Sync standard. Deliberately running the clock to fast will result in random timing, not good for MIDI, but great for crazy stuff. It is unknown how fast the clock can be run while still converting every pulse to MIDI clock. It's guaranteed to 350bpm to it is likely to be as high as 600bpm.

**INT/EXT CLK** switch - **STOP** or **GO**

Switching to go puts a run signal on the Internal or External clock line.

When MIDI is selected as the clock source this switch is ignored. The sequencer listens for Start and Stop MIDI messages instead. When INT/EXT is selected as the clock source this switch is active and will start and stop the sequencer.

32 STEP TIMING SEQUENCER

**Step Switches** - **1** to **32**

These send note trigger events to the 12 step sequencer.

Flick a switch down to trigger a Note On event on a particular step (like pressing a key down on a keyboard). When you would like a note to finish, select a later step and flick that switch up (like lifting your finger off a keyboard). In the centre position no trigger is sent. If two Note On events occur consecutively, the first note is considered to have ended when the next one starts. The sequencer will place a 1mS space (at 0V) in between the two so that envelope generators know to retrigger.

**LENGTH** switch - **32** or **16**

Controls the length of a pattern.

The 32 step sequencer can run in either 32 or 16 step mode. In 16 step mode the sequencer will automatically resynchronise so that 16 steps take the same amount of time that 32 steps would have. This means you can use this switch on the fly and not lose time with MIDI. All changes are applied at the beginning of the next MIDI bar after the pattern completes.

**PATTERN** switch - **A** or **B**

Selects which set of step switches is used for the current 16 step sequence.

Timing sequencer has a total of 32 steps. When in 16 step mode either the first 16 steps can be used or the last 16 steps. The first 16 are referred to as pattern A and the second 16 referred to as pattern B. Select which 16 are active by using this switch. When the sequencer is in 32 step mode this switch is ignored.

## 32 STEP TIMING SEQUENCER

**SYNC RATIO** switch - **1:1, 1:2, 1:4**

Changes the ratio at which the 32 step sequencer tracks a MIDI bar.

The first number in the ratio the length of a pattern (ie. **1** pattern), the second how many MIDI bars the pattern takes to complete (**1**, **2** or **4**). The 32 step sequencer will run 4 times slower on 1:4 speed than it will when running at 1:1 speed. You can change this switch in real-time without losing sync with the start of the MIDI bar. All changes are applied at the beginning of the next MIDI bar after the pattern completes.

**DIRECTION** switch - **FWD, REV** or **ALT**

Selects the direction of the sequence.

When switching from forward to reverse the Note On and Note Off triggers occur in reverse order, completely inverting the relationship between off and on. The notes now become rests and the rests become notes. This also means that Note On triggers that were set for a down beat no longer occur exactly on the down beat. ALT (Alternate) mode alternates between forward and reverse. Changes are applied at the end of the pattern.

**EOP RESET** switch - **PASS** or **BLOCK**

Allows or prevents the **E**nd **O**f **P**attern Reset signal from reaching the 12 step sequencer.

At the end of the pattern the 32 step sequencer generates a reset signal which is sent to the 12 step sequencer. Blocking this signal will prevent the 12 step sequencer from resetting. The result will the 12 step sequencer does not return to the first step at the end of a pattern. Instead it will continue to increment until it overflows.

## 12 STEP CV SEQUENCER

**MODE** rotary switch

Changes the step order of the 12 step sequencer.

**FWD** - Forward
The sequence will start at the left-most step and progress one step at a time to the right. You must pass the EOP RESET signal to synchronise this mode with the 32 step sequencer. If the EOP RESET signal is blocked, the sequencer will overflow in the same direction as it started.

**REV** - Reverse
The sequence will start at the right-most step and progress one step at a time to the left. You must pass the EOP RESET signal to synchronise this mode with the 32 step sequencer. If the EOP RESET signal is blocked, the sequencer will overflow in the same direction as it started.

**ALT** - Alternate (Ping Pong)
The sequence will start at the left-most step and progress one step at a time until it reaches the end of the row. Any overflow will be in the opposite direction to which it was travelling when it overflowed. Also, an EOP RESET will cause the next step to be the either the first or last step, whichever is required to travel in the opposite direction to when the reset was received.

12 STEP CV SEQUENCER

**MUTE** - OFF
None of the 12 steps will be selected for output. This effectively mutes the sequencer. There is no overflow and the EOP RESET signal is ignored.

**SKIP** - Jump steps
The sequence will start at the left-most step and skip every other step thereafter. On the first pass all the odd steps are selected and on the second pass all the even steps are selected. Any overflow will continue the same way. Pass the EOP RESET signal to synchronise this mode with the 32 step sequencer.

**VAS** - Voltage Addressed Stepping
The voltage received at the VAS jack is used to select the active step. The input range for addressing is between 0V and 5V. Higher voltages will be treated as 5V. The input is tied to ground, so if nothing is plugged into the VAS jack, the left-most step will always be selected. There is no overflow and the EOP RESET is ignored.

**BROWNIAN** - A random walk with a bias towards selecting adjacent steps, while generally moving forward.
There is no overflow direction, the next step depends on the randomness of the selection process. It's recommended to BLOCK the EOP RESET signal when using this mode. If the reset signal is blocked the step selection will appear to be more random and cover the whole 12 step range. If the sequencer is continually reset it will keep relocating to the left-most step at the end of each bar, limiting the "walking" distance; which is less fun.

**LAST** - Last step only
The right-most step is the only step used in the sequence. This can be useful if you only need a single pitch. There is no overflow and the EOP RESET signal is ignored. Conversely, use VAS with nothing plugged into the VAS jack to select the left-most step only.

**MANUAL** switch - **STEP** or **RESET**

Allows a manual modification to the running sequence.

While a sequence is running, flick the manual switch up to insert a step, or flick if down to reset the sequence early. This allows you to mix things up a bit. When the sequencer is not running this which will have no effect, but there *is* a way to play the sequencer totally manually:

1. Remove all Note On and Note Off triggers from the 32 step sequencer.
2. BLOCK the EOP RESET signal.
3. Start the 32 Step Sequencer (no triggers will be sent).
4. Use the manual switch.

**HOLD** switches - **CV1** or **OFF**, **CV2** or **OFF**

Holds the CV1 / CV2 voltages until the next step in the sequence occurs.

When a Note Off is received the 12 step sequencer will remove the voltage from the current step. This will cause the pitch of the any oscillators tracking the CV outputs to fall in between notes. To prevent this, switch the HOLD on for that CV. The CV will then be held during the note off period. The result will be a stable oscillator pitch during the release phase of an ASDR for example.
Note: Try send sending CV2 to a filter cutoff, turn HOLD off and use lots of slew. The CV will be pulled down in between notes causing an ACID bounce.