## 1. Introduction

Module **A-160 (Clock Divider)** is a frequency divider for clock signals, designed to be a source of lower frequencies, particularly for rhythm uses.

The **Trigger input** will take clock signals from, eg., an LFO, MIDI sync, or the gate from a MIDI-CV interface. At the **outputs**, you have access to the **sub-divided clock signals**, from half the clock frequency down to 1/64.

The A-160 also has a **reset input**. Whenever a reset signal is sensed, all outputs are set to zero, until the reset voltage disappears.

The Clock Divider can be used in combination with the **A-161 Clock Sequencer** to produce stepped sequences with a length of from one to eight events.

2. Overview

A-160

CLOCK DIVIDER

Trig. In

Res. In

① / 2

② / 4

③ / 8

④ / 16

⑤ / 32

⑥ / 64

❶
❷
❸
❹
❺
❻
❼
❽
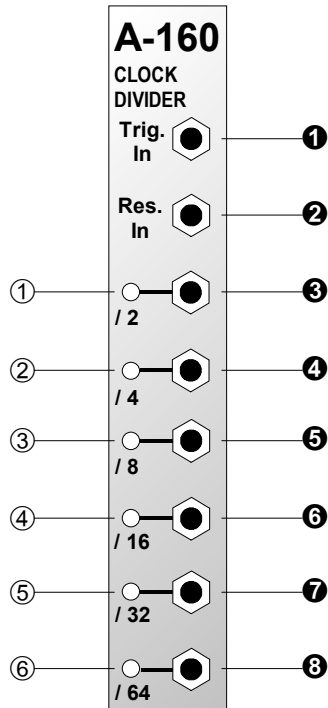
Indicators:

① **LED** :   Clock / 2 indicator
② **LED** :   Clock / 4 indicator
③ **LED** :   Clock / 8 indicator
④ **LED** :   Clock / 16 indicator
⑤ **LED** :   Clock / 32 indicator
⑥ **LED** :   Clock / 64 indicator

In / Outputs:

❶ **Trig. In** :   clock signal input
❷ **Res. In** :   reset input
❸ **/2** :   1/2 clock frequency output
❹ **/4** :   1/4 clock frequency output
❺ **/8** :   1/8 clock frequency output
❻ **/16** :   1/16 clock frequency output
❼ **/32** :   1/32 clock frequency output
❽ **/64** :   1/64 clock frequency output

3. Indicators

① **LED ... ⑥ LED**

LEDs ① to ⑥ indicate the status of each of the sub-divided frequencies at outputs ❸ to ❽.

4. In / Outputs

❶ **Trig. In**

Trigger input: patch the frequency to be divided in here.

❷ **Res. In**

Socket ❷ is the **reset input** for the A-160. When a reset voltage is sensed, all outputs go to zero, for as long as the reset voltage is present. (ie. a 'static reset')

☞   With a Clock Sequencer (A-161) connected, every time the A-160 senses a reset signal, the A-161 returns to output ❶ .

🔌   With the help of the reset input, you can produce sequences with less than eight steps. For instance, by patching output ❼ of the A-161 to the reset input socket ❷ of the A-160 you get a six-step sequence. (Instead of going to step seven, it resets to step one.)

❸ **/2 ... ❽ /64**

Sockets ❸ to ❽ are the A-160 **outputs**, from which the sub-divided clock signals are available.

5. User examples

"Ping-Pong" effect

The patch in Fig. 1 produces a "Ping-Pong" effect: with each note on the keyboard (gate signal) the audio jumps between left and right outputs - **Out$_L$** and **Out$_R$.**

The A-160 is halving the gate frequency, and triggering the A-150 VCS, so that each of its outputs is active only every other note.

ADSR envelope with re-triggering

In the patch in Fig. 2, an ADSR is re-triggered by a square wave LFO, sub-divided in the A-160 by a factor of four (output ❹). The original whole LFO oscillation is patched to the re-trigger input on the ADSR. In this way, a new repeating envelope is created.
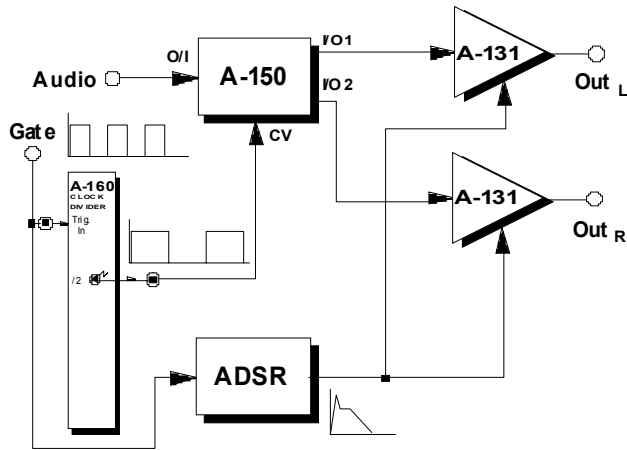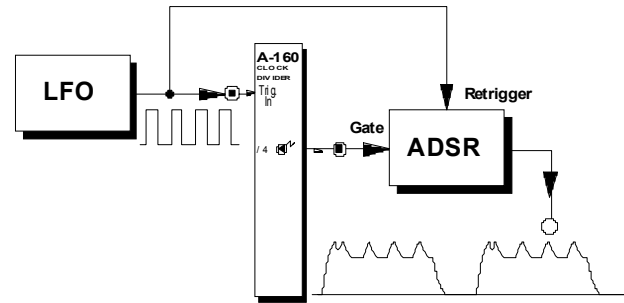
Fig. 1: Ping-Pong effect

Fig. 2: ADSR envelope with retriggering

By using different outputs on the A-160 it's possible to have envelopes re-triggering different numbers of times, from two to thirty-two.

In addition, by using an A-161, 3-, 5-, 6- or 7-step triggering is possible as well. You patch the A-161's ❹, ❺, ❻ or ❼ output to the gate input of the ADSR. Output ❺, ❻, ❼ or ❽ is patched to the A-160's ❷ reset input socket.

☞ Other suggestions for using the A-160 are in the A-161 Clock Sequencer's section of the manual.

6. Patch-Sheet

The following diagrams of the module can help you recall your own **Patches**. They're designed so that a complete 19" rack of modules will fit onto an A4 sheet of paper.

Photocopy this page, and cut out the pictures of this and your other modules. You can then stick them onto another piece of paper, and create a diagram of your own system.

Make multiple copies of your composite diagram, and use them for remembering good patches and set-ups.

•   Draw in patchleads with colored pens.