XAOC GERA 1989 Model Binary Product Commander Instruction Manual

Contents

XAOC GERA 1989 Model Binary Product Commander

module explained

SALUT

Thank you for purchasing this Xaoc Devices product. Gera ['gɛra] is a component of the Leibniz Binary Subsystem. It works by processing data in the Leibniz bus and requires a connection to other Leibniz modules. It features eight individual gate inputs that affect the corresponding individual bits of the Leibniz data. Eight illuminated tact switches allow for manual inverting of each gate input. The primary function of Gera is masking individual bits of the digital data through the AND operation. For example, when connected to Drezno, which is processing a waveform or voltage, masking individual bits yields various forms of quantization. However, Gera does not need Drezno to work. The AND operation is also a basic building block for sequence automation, chaos and rhythm generation, and various cybernetic modular patches. Bit processing logic in Gera is hardware-based; hence there is virtually no latency, and the binary signals may change at extreme rates.
To better understand the device and avoid common pitfalls, we strongly advise the user to read through the entire manual before use.

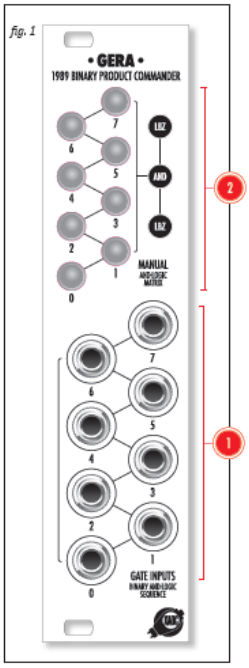
INSTALLATION

The module requires 6hp worth of free space in the Euro rack cabinet. Always turn the power off before plugging the module into the bus board using the supplied 10-pin ribbon cable, paying close attention to the power cable pinout and orientation. The red stripe indicates the negative rail and should match the dot or 12V mark on the bus board as well as the unit. Gera is internally secured against reversed power connection; flipping the 16-pin header may cause serious damage to other components of your system because it will short-circuit the +12V and +5V power lines. Always pay close attention to the proper orientation of your ribbon cable on both sides!
note: do not plug a power cable into the Leibniz headers as this will damage your unit and may also jeopardize other Leibniz modules connected to it!
Besides power, you need to connect Gera to other components of your Leibniz subsystem. Gera ships with one 10-pin ribbon data cable. To complete the connection, use the 10-pin cables included with your other Leibniz modules. Using the cable, connect the 10-pin un-shrouded header labeled out to the in header of your next Leibniz-compatible module (e.g., Drano, Jena, Erfurt, Odessa, etc.). Then connect the incoming data from the module preceding Gera (e.g., Stanchion, Drezno, or Erfurt) to the 10-pin unshrouded header labeled in. Please observe the marks of pin #1 (red stripe) on all connected modules. We advise you to plan ahead as to how you wish to incorporate Gera into your modular setup.
The module should be fastened by mounting the supplied screws before powering up.

MODULE OVERVIEW

The front panel of Gera is shown in fig. 1. As
with other modules in the Leibniz series, Gera features a bank of eight binary gate inputs 1 expecting gate signals, where an active 5V gate signal represents a logic 1, and an inactive gate is logic 0. Although Gera will work with smoothly changing signals within the entire range of Euro rack voltages, we recommend plugging standard Euro

rack gates and square waveforms for the best results.
Above the input jacks is a bank of eight illuminated manual tact switches 2 , each corresponding to a gate input of the same number. Pressing the button toggles between taking the input gate signal as-is and taking the inverted state of it. The result of this gate (inverted or not) is shown in the red backlight of the button. The eight gate signals affect the corresponding eight bits of the Leibniz data utilizing the logic function AND. In other words, every bit from the Leibniz bus is passed through only if the corresponding gate is active, which is indicated by the illumination of the switch.



PRINCIPLE OF OPERATION

There are eight binary AND gates inside Gera. These gates have two inputs: one (a) is connected to the individual data bit from the Leibniz interface, and the second input (b) is connected to the jack and button on the front panel. Thus, each gate may affect one bit of the binary data according to the truth table of the AND function (fig. 2).
For example, if the gate signal in the input jack #7 (the most significant bit) is active (+5V), the backlight turns on. This state is sent to the AND gate together with the input on the Leibniz bus (see: fig. 3). The logic AND operation performed on the binary data on the Leibniz bus passes the bit #7 unchanged. However, if button #7 is switched to invert while the gate

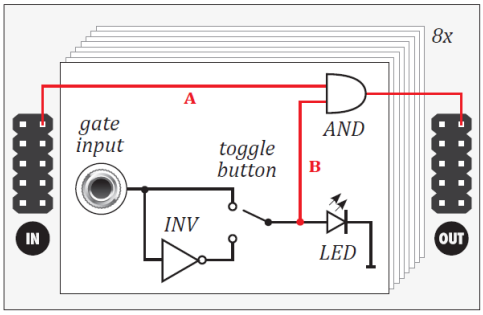


fig. 2: data processing in gear

is high, the resulting signal B will be logic 0 (button unlit), and it will mask bit #7 so that the resulting bit (available at the Leibniz header/bus) will be 0 regardless of the input state of this bit. The same masking of bit #7 occurs when the gate signal is inactive (0V), and is not manually inverted. note: masking all bits of the data (all buttons unlit) results in Gera not passing any data to its Leibniz out. This is in contrast to another Xaoc Devices module, Lipsky. In Lipsky, all buttons unlit indicate that the data is passed through unchanged.

EFFECT OF MASKING BITS

Data processed by the Leibniz subsystem modules may represent voltages or waveforms. For example, combinations of 8 bits represent 256 different values from 0 (binary 00000000) to 255 (binary 11111111), which are abstract in the digital domain but may be converted to a range of voltages by a DAC. What happens if a bit is masked (i.e., it is forced to be 0)? The set of 256 values is reduced because each of the two original values that differ only by this bit will be “glued” into one value with this bit equal to 0. For example, observe that 2 (binary 00000010) and 3 (binary 00000011) differ by the lowest

INPUT A	INPUT B	OUTPUT
0	0	0
0	1	0
1	0	0
1	1	1

fig. 3: LOGICAL 'AND' TRUTH TABLE

bit. Actually, all even and odd values differ by the state of this bit. Thus, masking the lowest bit will not change any even value but will con-vert all odd values to even; therefore, the set of possible values will shrink from 256 to 128. Masking more bits from the bottom end up re-duces it even further. For example, masking the lowest four bits leaves only the four upper bits active, which results in 24=16 possible combinations, from 0 (binary 00000000), through 16 (binary 00010000), 32 (binary 00100000), up to 240 (binary 11110000). The process of reducing the number of possible values is called quantization. Indeed, connecting Gera in a loopback of Durazno’s ADC and DAC and turning some of the lowest bits off allows one to create a crude quantizer. Passing a variable waveform through this setup results in a stepped waveform at the output (fig. 4a).
Masking high bits also reduces the set of values, but the impact on waveforms differs. Ob-serve that all values greater than 127 have the most significant bit set to 1, e.g., 213 (binary 11010101) because this bit represents a value of 27=128. Masking this bit will not affect any values lower than 128 but will subtract 128 from higher values; for example, 213 will be changed

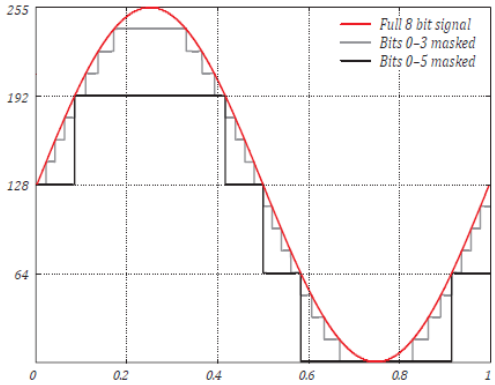


fig. 4a: EFFECTS OF MASKING THE LEAST SIGNIFICANT BITS

to 85 (binary 01010101). This is equivalent to the modulo operation. Masking more of the highest bits will decrease the range further; for example, masking the three most significant bits is equivalent to n mod 32, where n is the initial value. Processing a waveform will introduce a particular type of folding (fig 4b). note: the modulo operation significantly reduces the amplitude.

PATCH IDEAS

- When combined with Drezno, Gera may serve a variety of quantization and wave shaping functions, which can be automated by patching various gate signals into the module’s front panel jacks. An interesting variation of this concept is to chain Erfurt (either with no input or programmed with Lipsk) through Gera and then to the DAC section of Drezno. Using Drezno’s CV out to drive a VCO produces interesting glissandi. Similarly, when Erfurt is driven from Drezno to modulate its step value, adding Gera between the two modules reduces the

noise of low significant bits from Drezno, thus making

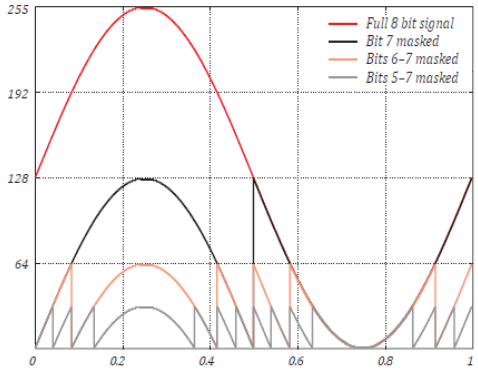


fig. 4b: EFFECTS OF MASKING THE MOST SIGNIFICANT BITS

the patch more stable, though at a price of losing precise control.

- Xaoc Devices Odessa oscillator features a Leibniz input that controls the switching of groups of harmonic partials within banks of partials. While this may be animated from many Leibniz modules (using Moskva+ Ostankino is an attractive option), sometimes it is desirable to limit the animation to just a particular partial or a group of partials. Gera, installed between Stanchion and Odessa, allows just that by masking the activity of unwanted bits.

CONNECTIVITY

Gera connects to all modules compatible with the Leibniz binary subsystem: Drano, Lipsk, Jena, Erfurt, Poczdam, Stanchion, and Odessa.

ACCESSORY

Our Coal Mine black panels are available for all Xaoc Devices modules. Sold separately. Ask your favorite retailer.
•

WARRANTY TERMS

XAOC DEVICES WARRANTS THIS PRODUCT TO BE FREE OF DEFECTS IN MATERIALS OR WORKMANSHIP AND TO CONFORM WITH THE SPECIFICATIONS AT THE TIME OF SHIPMENT FOR ONE YEAR FROM THE DATE OF PURCHASE. DURING THAT PERIOD, ANY MALFUNCTIONING OR DAMAGED UNITS WILL BE REPAIRED, SERVICED, AND CALIBRATED ON A RETURN-TO-FACTORY BASIS. THIS WARRANTY DOES NOT COVER ANY PROBLEMS RESULTING FROM DAMAGES DURING SHIPPING, INCORRECT INSTALLATION OR POWER SUPPLY, IMPROPER WORKING ENVIRONMENT, ABUSIVE TREATMENT, OR ANY OTHER OBVIOUS USER-INFLICTED FAULT.

LEGACY SUPPORT
IF SOMETHING GOES WRONG WITH A XAOC PRODUCT AFTER THE WARRANTY PERIOD IS OVER, THERE IS NO NEED TO WORRY, AS WE'RE STILL HAPPY TO HELP! THIS APPLIES TO ANY DEVICE, WHEREVER AND
WHENEVER ORIGINALLY ACQUIRED. HOWEVER, IN SPECIFIC CASES, WE RESERVE THE RIGHT TO CHARGE FOR LABOR, PARTS, AND TRANSIT EXPENSES WHERE APPLICABLE.

RETURN POLICY
THE DEVICE INTENDED FOR REPAIR OR REPLACEMENT UNDER WARRANTY NEEDS TO BE SHIPPED IN

THE ORIGINAL PACKAGING ONLY AND MUST INCLUDE A COMPLETED RMA FORM. XAOC DEVICES CAN NOT TAKE ANY RESPONSIBILITY FOR DAMAGES CAUSED DURING TRANSPORT. SO BEFORE SENDING US ANYTHING, PLEASE CONTACT US AT SUPPORT@XAOCDEVICES.COM. NOTE THAT ANY UNSOLICITED PARCEL WILL BE REJECTED AND RETURNED!

GENERAL INQUIRIES
FOR USER FEEDBACK SUGGESTIONS, DISTRIBUTION TERMS, AND JOB POSITIONS, FEEL FREE TO CON-TACT XAOC DEVICES AT **INFO@XAOCDEVICES.COM**. PLEASE VISIT XAOCDEVICES.COM FOR IN-FORMATION ABOUT THE CURRENT PRODUCT LINE, USER MANUALS, FIRMWARE UPDATES, TUTORIALS, AND MERCHANDISE.

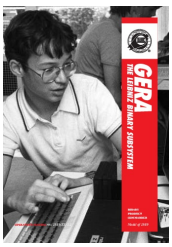
MAIN FEATURES

- Leibniz Binary
- Subsystem component
- Logical AND operation on Leibniz data
- Masking of individual bits controlled manually and by gate signals
- Visual indication by illuminated buttons

TECHNICAL DETAILS

- Euro rack synth compatible
- 6hp, skiff friendly
- Current draw: +60mA/-0mA
- Reverse power protection
- NOT protected against plugging power to Leibniz headers!

Documents / Resources

	XAOC GERA 1989 Model Binary Product Commander [pdf] Instruction Manual GERA 1989 Model Binary Product Commander, GERA, GERA Binary Product Commander, 1989 Model Binary Product Commander, Binary Product Commander, Binary Commander, Product Commander, GERA Commander, Commander

References

- Xaoc Modular Devices