Introduction

Chaos Matrix is a programmer for an Oberheim Matrix 1000 analog synthesizer built by Patrick Manderson. It's capable of manipulating all 126 voice parameters including 10 programmable matrix modulations buses, ramp generators, 5 point tracking generator, 3 five stage envelopes, 2 oscillators, 2 LFOs, and much more.

It features edit buffer synchronization for visual feedback and inspection of all Matrix 1000 voices parameters, patch compare, patch init, advanced performance options for some parameters as well as containing all 1000 original patch names. Seeing patch names instead of numbers alone, makes it a lot easier to remember your favorites.

Chaos Matrix is currently a work in progress and is nearly complete. It was built with help from the MBHP (MidiBox Hardware Project) using the MIOS operating system. It uses 1 DIN module, 1 AIN module, 1 DOUT module and 1 core with a PIC18F4620 microprocessor along with a custom application built with the C skeleton interface.

Forum announcement here: http://www.midibox.org/forum/index.php/topic,14240.0.html

YouTube video demonstration.

Current Status (updated 20091212)

- The electronics need to be mounted inside the wooden enclosure.
- Rear panel with MIDI & power jacks
- A MIDI merge has to be built, its necessary for the edit buffer sync feature without the need of a PC.
- Software complete. Beta source code available on this post
- How-To currently being documented

Features

Almost all communication from the Chaos Matrix to the M1000 is made via MIDI system exclusive messages. Non-registered parameter numbers were not implemented in the Matrix 1000 firmware despite what the manual states. Some communication from the M1000 to the Chaos Matrix is necessary however.

Real-Time Modulation Control

All button clicks and knob twists on the live panel send sysex voice parameter changes by default. Oscillator 1 and 2 frequency is controlled by rotary encoders to improve tuning accuracy. The rest of the knobs are potentiomenters. The pulse width potentiometers are dual purpose. In one mode, they adjust the pulse width to a fixed duty cycle, in another mode, the same potentiometer adjust the modulation depth of the pulse width controlled by LFO2. These two modes can be toggled by clicking on the pulse waveform button. The sync button cycles through 4 sync modes, LFO waveform buttons cycle through the 7 available waveforms. Oscillator click toggles click on/off as well as enabling the noise waveform.

Patch Initialization

Voice parameters can be initialized to a default setting. Voice edit initialization is a good starting point to begin creating a new patch from scratch. The initialized parameters are sent to the M1000 so the display on the Chaos Matrix matches what is heard on the Oberheim.

Edit Buffer Synchronization

When selecting a new patch, the edit buffer is requested from the M1000 and processed by the Chaos Matrix, updating the status of all LEDs and parameters in the soft panel. When the edit buffer is received, you can inspect the matrix modulation, ramps and tracking generator parameters for all the patches in the M1000. When you tweak any of the live panel controls, the original value as well as the newly adjusted value is displayed. This feature requires additional MIDI cables to route the MIDI out of the M1000 back to the MIDI in of the Chaos Matrix.

Patch Compare

When editing patches, the patch compare feature lets you compare your current edit to the original patch.

Saving Voices

In the edit menu, select the 2nd page and select a patch location (0-199) to save your current patch.

Voice Names

The M1000 doesn't handle voice names like the Matrix 6/6R, they are simply ignored and never stored. However, each and every patch name (published when the M1000 was released) was painstakingly included in the Chaos Matrix firmware and is displayed while browsing patches. Seeing patch names on the LCD instead of numbers alone, makes it a lot easier to remember your favorites.

Enhanced Performance Options

Due to the M1000 processor lag when adjusting parameters by sysex, the filter frequency control can optionally send continuous controller messages which are then handled by a matrix modulation bus. When this feature is enabled, the first free bus is used to route lever 3 (breath controller) to filter frequency. The frequency knob will then send continuous controller messages on the current MIDI channel.

History

The project started in 2006 with a bold design: build a voice programmer for an Oberheim Matrix 1000 with a dedicated control for every parameter.

Unfortunately there was a physical limit to the number of analog and digital inputs I could use with one core. So I compromised by layering groups together, like the DCO, LFO and ENV controls with select buttons.

The prototype, containing all the surface controls was built and mounted in a pair of cardboard boxes. 90% of the software was completed and all the panel controls worked as designed.

The project didn't go much beyond that for a few reasons.

- The cost of manufacturing the aluminum panel, along with the engraving for each and every control (104) exceeded $450. Way too much money than I was willing to spend.
- The aesthetics of all the controls on a 6u rack mountable panel was rather unattractive and and difficult to read. I also realized for a lot of controls, I rarely used them (ramps, tracking parameters, modulation switches). These were taking up valuable space on the panel and getting everything arranged neatly on the panel was difficult.
- I ran out of flash storage on the PIC18F452 microprocessor and had a lot more features to code for the LCD panel and the matrix modulation parameters. This was later resolved by getting another PIC with more storage.

- And finally, the compromises I made layering some of the most important parameters made sound programming non intuitive.

After all my hard work, I didn't want to invest any more money and time on something that was ugly and costly and didn't work as intended. For the next few years the prototype sat in my office. Occasionally I would turn it on and play it a while. Eventually I got tired of looking at an incomplete project and decided to give it another shot. This time, sticking to the basics... keep it simple, stupud!

Panel Design



The redesign of the panel began with inspiration from a few classic analog synths: The panel to the Oberheim OBX, Roland Jupiter 8 and Jupiter 6. I liked how simple the two rows of controls were laid out on the Jupiter 8 and how easy it was to see the flow of synth parameters. I knew that the Chaos Matrix would have to have no more than two rows of controls and they would have to fit in 24 inches of the “Live Panel” and the rest of the parameters would be menu driven from the “Soft Panel”. FrontPanel Express file is attached to this post. Please note: I mistakenly didn't engrave the OSC-LFO pots with the negative, center and positive markings. The software currently treats those parameters in the positive ranges only. If you build one of these, let me know if I can assist in updating the firmware to accommodate both positive and negative. The cost of the panel through FrontPanel Express was approximately $175 with shipping in the USA.

Live Panel

The most important parameters would have a dedicated control here. Note: the Matrix 1000 actually has 3 envelopes. The 1st is almost always modulating the filter cutoff, the second envelope modulated the main VCA and the 3rd was sometimes used to modulate the FM amount or the DCO pitch. I decided the 3rd envelope would be accessed through the soft panel when tweaking the preset patches.

Soft Panel

The rest of the parameters would have to be controlled from a menu driven interface using a 2×20 CLCD, 1 encoder and 8 buttons with LEDs.

LFO Section

2 LFOs, each with a Speed and Depth potentiometer, and a button to toggle 1 of 7 waveforms. The last waveform “Sample Modulation” can further be edited on the soft panel with its rotary encoder.

The speed values are also indicated by blinking the LEDs on the waveform buttons. The slower the LFO setting, the slower the blinking.

Oscillator Section

All the knobs on the Live panel are potentiometers except for DCO1 and DCO2 frequencies. The prototype originally used potentiometers but I found it difficult to dial in an exact frequency. I decided to use detented encoders instead.

Pulse waveform can be enabled by toggling the pulse button and altering its shape with the “Width” knob below it. Click the pulse button a 2nd time the button blinks and any adjustments to the “Width” knob instead adjusts the pulse width modulation depth to LFO 2.

DCO2 Click/Noise button is a 3 state button allowing you to enable the noise wave without needed an additional button. When noise is enabled, the LED blinks.

The Sync button toggles the 4 oscillator sync modes.

Filter Section

The filter to envelope modulation modulates the cutoff frequency to envelope 1 by default when editing preset patches. If a user patch is initialized, the modulation is routed to envelope 3 instead.

Some of the parameters sent as system exclusive messages causes the Matrix 1000 to hang. If you're playing notes and change some of the parameters, the notes will hold for a brief moment. Most of the hanging is overcome by simply controlling the interval of messages being transmitted over MIDI for each parameter. Most behave fine and the message can be sent at normal speed, but for others, the transmit interval was reduced in order to compensate for the hanging.

The effect is ideal for slow pot movements, but for quick changes, a lot of the values in between the 1st and last movement are dropped. Despite this, the solution is better than having the Matrix 1000 hang.

Envelope Section

Instead of naming the envelopes “Envelope 1”, “Envelope 2” etc… I simply named them as they are

on the Oberheim OBX synthesizer. The filter envelope adjusts envelope 1 or 3 depending on which patch is being edited.

Panel Construction

Some of the panel electronics, including the LCD module have to be mounted on the underside of the panel. I used stripboards from FutureLec to mount the buttons together and applied Wilba's method of attaching countersink machine screws with JB-Weld epoxy. The epoxy has to be applied in two steps to ensure a good weld.

Step one: Simply dab a bit of the epoxy on the flat head of a machine screw and place the head down on the aluminum panel (be sure to clean the panel surface thoroughly). Use a pair of clamps to provide pressure between the screw and the panel. Wait at least 24 hours to allow the epoxy to cure.

Step two: Apply an additional amount of epoxy on the underside of the head of the counter-sunk screw and spread it over to the aluminum. You want to maximize the contact between the screw and aluminum surface. Allow another 24 hours to cure.

How to build a Chaos Matrix

I realize some people may not want to build theirs using the same panel layout, same pots, buttons encoders etc, but I'll document my construction as much as possible so it can be used as a reference for those who want to build their own custom programmers for the Oberheim Matrix 1000. I would categorize the construction difficulty as medium-advanced.

Hardware

- 1 PIC based core with a PIC18F4620
- 1 AIN, 1 DIN, 1 DOUT
- 1 2x20 CLCD with a 116 x 37 PCB and a 83 x 16.6 (or more) viewable area. If using a different size, changes will have to be made on the front panel.

- 25 Alpha 10k linear potentiometers
- 3 Bournes detented encoders
- 16 Mountain Switch buttons with LEDs
- 1 small red LED T1 (.125“ Dia) 3mm
- 2 large stripboards (110 x 94mm) STPBRD1
- 5 small stripboards (94 x 53mm) STPBRD2
- 3 panel mount 5 pin common MIDI jacks from avishowtech
- 1 DC power jack
- 3 24 x 5 1/2 x 3/4 oak boards from local hardware store for the cabinet
- 8 L brackets with screws for the cabinet
- 2 pieces of trim stock for the cabinet
- wood glue
- wood stain
- poly coat for the wood

Building the Modules

Follow the standard instructions for building the 4 Midibox components and getting them connected to the core. You will also have to build a cable to connect the core to the character LCD module. I would highly recommend getting PCB kits from avishowtech.com instead of building and parting out your own. The PCBs are of excellent quality.

- PIC based core
- AIN module
- DIN module
- DOUT module
- wire up a LCD cable

The control surface components requires VSS and VDD power connections, I used one of the small stripboards as a common power rail to connect these components to, by jumping VSS and VDD from the PIC core. You may want to do it differently however.

Front Panel

All the pots can be mounted directly on the panel. Here is a simple guide on wiring pots.

AIN PIN wiring

- 0 LFO1 DEPTH
- 1 LFO1 SPEED
- 2 UNUSED (pin to ground)
- 3 OSC1 WAVE SHAPE
- 4 OSC1 LFO1 MOD
- 5 LFO2 DEPTH
- 6 OSC1 PLSE WIDTH
- 7 LFO2 SPEED
- 8 FREQ → ENV
- 9 RESONANCE
- 10 OSC2 LFO1 MOD

- 11 OSC2 PLSE WIDTH
- 12 FM → PRESSURE
- 13 FM AMOUNT
- 14 OSC2 WAVE SHAPE
- 15 FREQUENCY
- 16 FILTER RELEASE
- 17 FILTER SUSTAIN
- 18 ENV ATTACK
- 19 ENV SUSTAIN
- 20 ENV RELEASE
- 21 FILTER ATTACK
- 22 ENV DECAY
- 23 FILTER DECAY
- 24 DETUNE
- 25 OSC1←MIX→OSC2

Finally, clamp the remaining unused AIN pins to ground.

The Mountain Switch buttons can be mounted on a small PCB. 2 of its pins are for the button, the other two are for the LED.

DIN pin wiring for buttons and encoders

- 0 LFO2 WAVE
- 1 LFO1 WAVE
- 2 DCO1 ENCODER
- 3 DC01 ENCODER
- 4 OSC SYNC MODE
- 5 DCO1 PULSE
- 6 DCO1 WAVE
- 7 DCO1 CLICK
- 8 DCO2 ENCODER
- 9 DCO2 ENCODER
- 10 EDIT ENCODER
- 11 EDIT ENCODER
- 13 DCO2 CLICK
- 14 DCO2 WAVE
- 15 DCO2 PULSE
- 20 PAGE
- 21 MATRIX MODULATE
- 22 KEYBOARD
- 23 ENVELOPES
- 28 OSCILLATORS
- 29 FILTER
- 30 EDIT
- 31 PATCH SELECT

DOUT pin wiring for buttons

- 0 DCO2 CLICK
- 1 DCO2 WAVE
- 2 DCO2 PULSE

- 4 OSC SYNC MODE
- 5 DCO1 PULSE
- 6 DCO1 WAVE
- 7 DCO1 CLICK
- 8 ENVELOPES
- 9 KEYBOARD
- 10 MATRIX MODULATE
- 11 PAGE
- 16 PATCH SELECT
- 17 EDIT
- 18 OSCILLATORS
- 19 FILTER
- 21 LFO2 WAVE
- 22 LFO1 WAVE
- 23 MIDI ACTIVITY

From:
https://midibox.org/dokuwiki/ - **MIDIbox**

Permanent link:
https://midibox.org/dokuwiki/doku.php?id=chaos_matrix&rev=1261932552

Last update: **2009/12/27 16:49**