Erae

用户手册

Embodme

2026-10-29

Contents

1 简介 **1**

1.1 什么是 Erae 2? . . . 1
1.2 主要特性 . . . 2
1.3 本测试版的新功能 . . . 2
1.4 本手册的适用对象 . . . 3
1.5 包装清单 . . . 3
1.6 手册约定 . . . 3

2 快速入门 **5**

2.1 开机与启动流程 . . . 5
2.1.1 低功耗模式 . . . 5
2.2 USB 连接（兼容 Class-Compliant MIDI） . . . 6
2.2.1 在 DAW 中将 Erae 与 Erae Sound 配合使用 . . . 7
2.3 MIDI 输入与输出 . . . 7
2.4 USB Host 端口 . . . 7
2.5 您的第一个 Layout . . . 8
2.6 独立模式与 Lab 连接模式 . . . 8
2.6.1 独立模式 . . . 9
2.6.2 Lab 连接模式 . . . 9
2.7 后续步骤 . . . 9

3 第 3 章：触控表面与触控 **10**

3.1 触控板网格 . . . 10
3.2 XYZ 感应 . . . 11
3.3 10 指多点触控 . . . 11
3.4 力度曲线设置 . . . 12
3.5 灵敏度 . . . 12
3.6 物理保养 . . . 12

4 Layout **14**

4.1 什么是 Layout? . . . 14
4.2 每个项目的 Layout 数量 . . . 14
4.3 在 Erae 2 上切换 Layout . . . 15
4.4 LED 指示灯 . . . 15
4.5 在 Erae Lab 中编辑 Layout . . . 16

5 元素 **17**
5.1 5.1 Iso Keyboard . . . 18
5.2 5.2 Chroma Keyboard . . . 20
5.3 5.3 Drumpad . . . 21
5.4 5.4 Fader 1D . . . 22
5.5 5.5 Fader 2D . . . 23
5.6 5.6 Button {#56-button} . . . 25
5.7 5.7 Key . . . 26
5.8 5.8 Ableton Launchpad . . . 27
5.9 5.9 API Zone {#59-api-zone} . . . 28
5.10 5.10 Pedal . . . 28
5.10.1 Switch . . . 28
5.10.2 Kick . . . 29
5.10.3 Sustain (Binary) . . . 29
5.10.4 Expressive . . . 30
5.10.5 Sustain Continuous . . . 30

6 MIDI 配置 **32**
6.1 USB MIDI 端口 . . . 32
6.1.1 2.x 更新后重新映射三端口设置 . . . 33
6.2 MIDI 通道与组 . . . 33
6.3 逐元素输出目标 . . . 33
6.4 琴键元素：表现力参数 . . . 34
6.4.1 压力 . . . 34
6.4.2 揉弦（Pitch Bend） . . . 35
6.4.3 滑音（Glissando） . . . 35
6.4.4 音色明亮度（CC 74） . . . 36
6.5 CC 映射 . . . 36
6.6 高分辨率 CC 和 NRPN . . . 37
6.7 MIDI 1.0 与 MIDI 2.0：您实际读取到的数值 . . . 37
6.8 MPE 配置 . . . 38
6.8.1 让滑移和压力进入您的 DAW . . . 38
6.9 MIDI 路由矩阵 . . . 39
6.10 相关主题 . . . 40

7 音阶与调音 **41**
7.1 内置音阶库 . . . 41
7.2 根音 . . . 42
7.3 八度 . . . 42
7.4 显示非音阶音（Show Offscale） . . . 43
7.5 音阶如何影响各类键盘 . . . 43
7.6 全局音阶 . . . 43
7.7 自定义音阶 . . . 44

7.8 各元素独立的音阶设置 44

8 琶音器与 Looper 45
8.1 琶音器 45
8.1.1 启用琶音器 45
8.1.2 速率（Rate） 45
8.1.3 演奏方式（Style） 46
8.1.4 八度范围（Octave Range） 46
8.1.5 压力转力度（Pressure to Velocity） 46
8.1.6 量化（Quantize） 47
8.2 Looper 47
8.2.1 主屏幕 Looper 显示 47
8.2.2 录制与回放 48
8.2.3 Looper 控制 48
8.2.4 时钟同步 48
8.2.5 演奏型长度 49
8.2.6 量化网格 49
8.2.7 击拍设速（Tap Tempo） 49

9 第 9 章：步进音序器（Step Sequencer） 50
9.1 进入与退出音序器 50
9.2 可以音序化的内容 50
9.3 表面：三个区域 51
9.3.1 Live 区域 51
9.4 一个单元格意味着什么：音高网格 52
9.4.1 相对音阶的录制方式 52
9.5 在网格上编辑步 53
9.5.1 选择并移动一组音符 54
9.6 在长演奏型中移动 54
9.7 Sequencer 按钮快捷操作 54
9.8 音序器 LCD 55
9.8.1 顶栏：泳道选择器 55
9.8.2 网格：全局地图 56
9.8.3 底栏：细分与长度 56
9.8.4 本画面中的编码器与 Return 57
9.9 切换演奏型 57
9.10 Sequencer Settings 57
9.11 自动保存 59
9.12 时钟 59

10 CV 输出 60
10.1 硬件概览 60
10.2 输出模式 60
10.3 按元素分配 CV 61

10.4 每个 Voice 的额外 CV 输出 . . . 61
10.5 Eurorack 快速上手 . . . 62
10.5.1 第一个 Voice 设置（Pitch + Gate） . . . 62
10.5.2 USB Host MIDI 转 Eurorack . . . 62
10.5.3 24 ppqn 的 CV Clock . . . 63
10.5.4 推荐的 Eurorack 模块配置 . . . 63
10.6 校准 . . . 63
10.7 CV 时钟输出 . . . 63
10.7.1 访问 CV Clock 设置 . . . 64
10.7.2 CV Clock 参数 . . . 64

11 第 11 章：LCD 界面 65
11.1 11.1 编码器导航模型 . . . 65
11.2 11.2 主界面 . . . 65
11.3 11.3 设置菜单{#113-settings-menu} . . . 66
11.4 11.4 琶音器设置界面 . . . 67
11.5 11.5 音阶 / 调式选择器{#115-scale–key-selector} . . . 68
11.6 11.6 信息界面（状态浮层） . . . 68
11.7 11.7 路由界面 . . . 69
11.8 11.9 CV 时钟界面 . . . 70
11.9 11.10 踏板输入配置{#1110-pedal-input-configuration} . . . 71
11.9.1 11.10.1 格式化 SD card 确认对话框 . . . 71
11.9.2 11.10.2 踏板输入校准向导 . . . 72
11.10 11.11 力度曲线编辑器 . . . 72
11.11 11.11b Sequencer 画面 . . . 73
11.12 11.12 元素路由界面{#1112-element-routing-screen} . . . 73
11.13 11.13 映射界面 . . . 74
11.14 11.15 保存 / 另存为 / 加载界面 . . . 74
11.15 11.16 自定义音阶界面{#1116-custom-scales-screens} . . . 75

12 第 12 章：开关与编码器 77
12.1 面板开关 . . . 77
12.1.1 编号开关（N1 到 N8） . . . 77
12.1.2 功能开关 . . . 77
12.2 编码器 . . . 78
12.2.1 滚动 . . . 78
12.2.2 按下 . . . 78
12.3 Return 开关 . . . 78
12.4 开关组合 . . . 78

13 第 13 章：LED 反馈 80
13.1 RGB 配色系统 . . . 80
13.1.1 亮度控制 . . . 80

13.2 各元素的可视化 . . . 80
13.2.1 琴键 . . . 81
13.2.2 按钮 . . . 81
13.2.3 推子（1D 与 2D） . . . 81
13.2.4 API 区域 . . . 82
13.3 动画 . . . 82
13.3.1 FingerGlow . . . 83
13.3.2 Ripple . . . 83
13.3.3 Halo . . . 83
13.3.4 ColorMorph . . . 83
13.4 按压反馈 . . . 83
13.5 状态指示 . . . 83
13.5.1 自动保存指示 . . . 84
13.5.2 项目已保存确认 . . . 84
13.5.3 时钟同步脉冲 . . . 84
13.5.4 开关 LED . . . 84

14 第 14 章：连接到 Erae Lab **85**
14.1 USB 连接 . . . 85
14.1.1 自动检测 . . . 85
14.2 连接指示 . . . 85
14.3 项目同步 . . . 86
14.3.1 冲突解决 . . . 86
14.4 通过 Erae Lab 进行固件更新 . . . 86
14.5 哪些功能需要 Erae Lab，哪些可独立使用 . . . 87

15 第 15 章：设置 **88**
15.1 全局设置 . . . 88
15.1.1 亮度 . . . 88
15.1.2 灵敏度 . . . 88
15.1.3 力度曲线 . . . 89
15.2 项目设置 . . . 89
15.2.1 速度 . . . 90
15.2.2 时钟源 . . . 90
15.2.3 节拍器 . . . 90
15.2.4 切换 Layout 时发送 CC . . . 90
15.3 自定义音阶 . . . 91
15.4 MIDI 路由 . . . 91
15.5 CV 时钟输出 . . . 91
15.6 踏板输入配置 . . . 92
15.6.1 类型 . . . 92
15.6.2 全局启用 . . . 93
15.6.3 路由 . . . 93

15.6.4 锁定 . . . 93
15.6.5 通道 . . . 93
15.7 校准流程 . . . 93
15.7.1 CV 校准 . . . 93
15.7.2 编码器校准 . . . 93
15.7.3 踏板输入校准 . . . 94
15.8 项目管理 . . . 94
15.8.1 MIDI 2.0 . . . 94
15.9 存储状态与恢复 . . . 95

16 第 16 章：故障排除 96
16.1 连接问题 . . . 96
16.1.1 Erae Lab 无法识别 USB . . . 96
16.1.2 DAW 中不出现 MIDI 端口 . . . 96
16.1.3 macOS 上不出现 MPE 端口（设备条目过期） . . . 97
16.2 MIDI 路由问题 . . . 97
16.2.1 音符未在预期的输出上发声 . . . 97
16.2.2 意外的音符或重复触发 . . . 97
16.3 触摸反应迟钝 . . . 98
16.4 电源问题 . . . 98
16.4.1 Low Power 消息 . . . 98
16.5 LED 无响应 . . . 98
16.5.1 表面 LED 不亮或部分点亮 . . . 98
16.6 出厂、备份和重置流程 . . . 99
16.7 引导加载程序恢复 . . . 99
16.8 孤立引导加载程序检测 . . . 99
16.9 SD card 错误 . . . 100
16.10联系支持 . . . 100

17 第 17 章：Erae 兼容性 101
17.1 什么是 Erae Touch? . . . 101
17.2 项目与 Layout . . . 101
17.2.1 Main/Alt Layout 配对 . . . 101
17.3 五按键导航系统 . . . 102
17.4 Erae Touch 上的步进音序器 . . . 102
17.5 Erae Touch 上的力度曲线 . . . 102
17.6 仅 Z 轴压力感应 . . . 103
17.7 无 LCD . . . 103
17.8 无 CV 输出 . . . 103
17.9 无 Looper，也无琶音器画面 . . . 104
17.10无编码器 . . . 104
17.11两阶段固件更新 . . . 104
17.12功能对比：Erae Touch 与 Erae 2 . . . 104

18 附录 A：参数参考 **106**
18.1 通用参数 . . . 106
18.2 控制器槽位 . . . 106
18.3 Key（琴键） . . . 107
18.3.1 基本参数 . . . 107
18.3.2 Pressure Tune（压力调校） . . . 107
18.3.3 Vibrato Tune（揉弦调校，Pitch Bend） . . . 108
18.3.4 CC 分配（可选） . . . 108
18.3.5 Control Voltage（控制电压） . . . 108
18.4 Button（按钮） . . . 109
18.4.1 通用按钮参数 . . . 109
18.4.2 Button：Note 类型 . . . 109
18.4.3 Button：Control Change 类型 . . . 110
18.4.4 Button：Program Change 类型 . . . 110
18.4.5 Button：Control Voltage 类型 . . . 110
18.4.6 Button：Tap Tempo 类型 . . . 110
18.5 Fader 1D（Fader 1D） . . . 111
18.6 Fader 2D（Fader 2D） . . . 111
18.7 Keyboard（键盘：Keyboard、Keygrid、Drumpad） . . . 112
18.7.1 键盘布局参数 . . . 112
18.7.2 Glissando Tune（滑音调校） . . . 113
18.7.3 Sound Brightness（CC 74） . . . 113
18.7.4 Arpeggiator（琶音器） . . . 114
18.8 Ableton Launchpad（LivePad） . . . 114
18.9 API Zone（API 区域） . . . 114
18.10 Non-Visible Elements（不可见元素） . . . 115
18.10.1 Pedal（踏板） . . . 115
18.11 Project Settings（项目设置） . . . 116
18.11.1 CV Clock Output（CV 时钟输出） . . . 117
18.11.2 MIDI Routing（MIDI 路由） . . . 117

19 附录 B：MIDI 实现表 **118**
19.1 主实现表 . . . 118
19.2 MPE 详情 . . . 119
19.3 Universal Identity Request . . . 119
19.4 SysEx 协议 . . . 120
19.4.1 API Zone 命令 . . . 120
19.5 默认 CC 映射 . . . 120

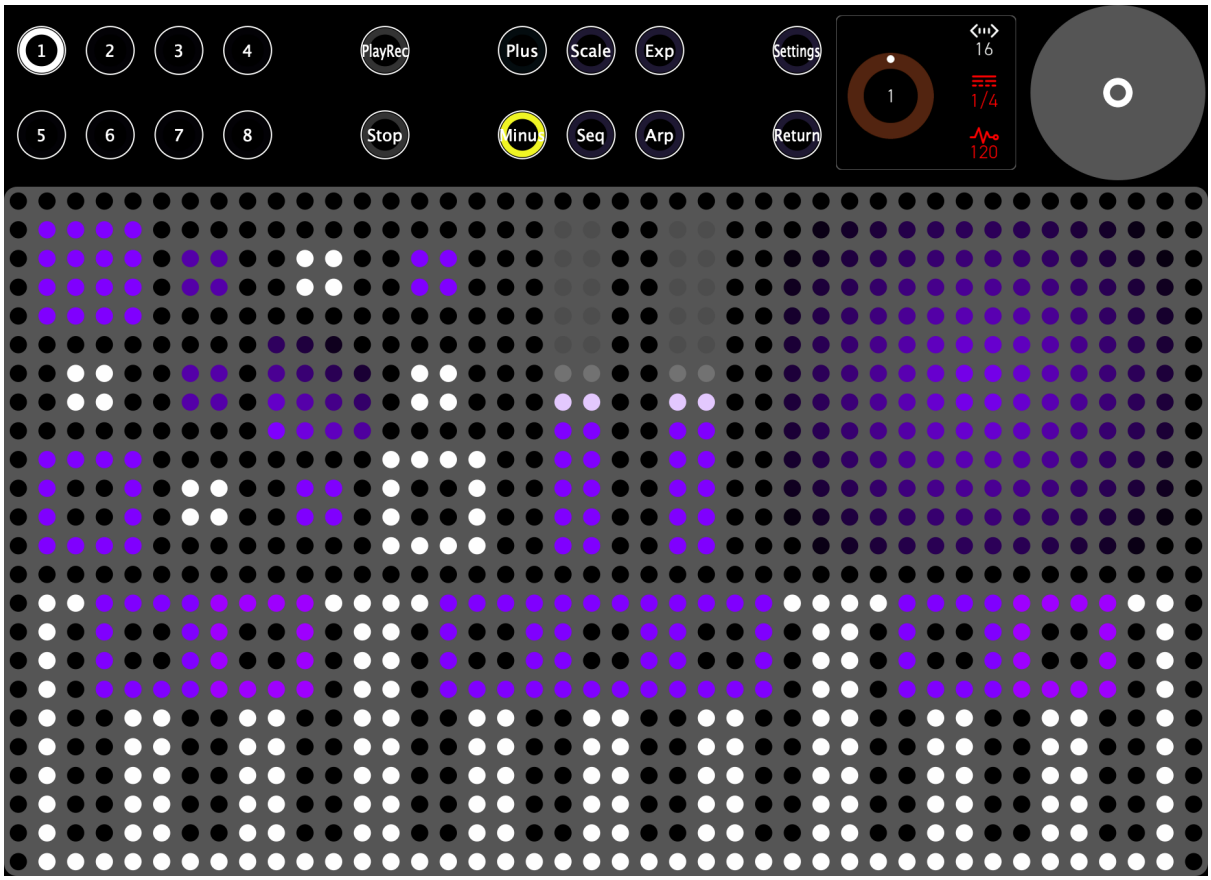
20 附录 C：DAW 设置指南 **122**
20.1 通用 MIDI 设置 . . . 122
20.1.1 USB MIDI 端口 . . . 122
20.1.2 在 DAW 中使用 Erae Sound . . . 123

20.1.3 macOS：移除陈旧的设备条目 . . . 123
20.1.4 Windows：确认端口可见性 . . . 123
20.2 Ableton Live . . . 124
20.2.1 第 1 步：启用 MIDI 端口 . . . 124
20.2.2 第 2 步：设置标准 MIDI 乐器轨道 . . . 124
20.2.3 第 3 步：设置 MPE 乐器轨道 . . . 124
20.2.4 第 4 步：安装 MIDI Remote Script（LivePad 元素） . . . 125
20.2.5 MIDI 时钟同步（Ableton -> Erae 2） . . . 125
20.3 Logic Pro . . . 125
20.3.1 第 1 步：在 Audio MIDI Setup 中启用 MIDI 输入 . . . 126
20.3.2 第 2 步：配置 Logic 的 MIDI 偏好设置 . . . 126
20.3.3 第 3 步：创建标准 MIDI 轨道 . . . 126
20.3.4 第 4 步：逐轨道启用 MPE . . . 126
20.3.5 第 5 步：将轨道指向 MPE 端口 . . . 126
20.3.6 Logic Pro：“Reset MIDI Drivers” 变通方法 . . . 127
20.3.7 MIDI 时钟同步（Logic -> Erae 2） . . . 127
20.4 FL Studio . . . 127
20.4.1 第 1 步：启用 MIDI 端口 . . . 127
20.4.2 第 2 步：将 MIDI 路由到乐器 . . . 128
20.4.3 第 3 步：使用 Multi-Link 进行 CC 映射 . . . 128
20.4.4 第 4 步：在 FL Studio 中近似实现逐通道 MPE . . . 128
20.4.5 MIDI 时钟同步（FL Studio -> Erae 2） . . . 128
20.5 Bitwig Studio . . . 129
20.5.1 第 1 步：配置控制器 . . . 129
20.5.2 第 2 步：在乐器轨道上启用 MPE 输入 . . . 129
20.5.3 第 3 步：验证 Note Expression . . . 129
20.5.4 第 4 步：使用 Generic Flexi 映射 CC . . . 130
20.5.5 MIDI 时钟同步（Bitwig -> Erae 2） . . . 130
20.6 Cubase / Nuendo . . . 130
20.6.1 第 1 步：启用 MIDI 端口 . . . 130
20.6.2 第 2 步：创建标准 MIDI 或乐器轨道 . . . 130
20.6.3 第 3 步：将 Erae 2 声明为 Poly Expression 设备（Cubase 12+） . . . 131
20.6.4 第 4 步：创建 MPE 乐器轨道 . . . 131
20.6.5 第 5 步：逐 Layout 的 MPE 区域配置 . . . 131
20.6.6 MIDI 时钟同步（Cubase -> Erae 2） . . . 131
20.7 Reaper . . . 131
20.7.1 第 1 步：启用 MIDI 设备 . . . 132
20.7.2 第 2 步：创建用于标准用途的 MIDI 轨道 . . . 132
20.7.3 第 3 步：在轨道上启用 MPE 输入 . . . 132
20.7.4 第 4 步：MPE Pitch Bend 范围匹配 . . . 132
20.7.5 第 5 步：MIDI 时钟同步（Reaper -> Erae 2） . . . 132
20.8 连接到硬件合成器 . . . 133
20.8.1 TRS MIDI 输出 . . . 133

20.8.2 USB Host 端口 . . . 133
20.8.3 硬件设置的 MIDI 路由矩阵 . . . 134
20.9 Eurorack 集成 . . . 134
20.9.1 原生 CV 输出 . . . 135
20.9.2 USB Host 连接到 Eurorack MIDI-CV 模块 . . . 136
20.9.3 示例补丁：完整的表现力声部 . . . 137
20.10相关主题 . . . 137

21 附录 D：开发者 API 139
21.1 D.1 传输 . . . 139
21.2 D.2 SysEx 头部 . . . 140
21.3 D.3 约定 . . . 140
21.3.1 D.3.1 坐标轴 . . . 140
21.3.2 D.3.2 颜色编码 . . . 141
21.3.3 D.3.3 Bitize-7（图像和手指数据编码） . . . 141
21.3.4 D.3.4 流式传输激活与回复封装 . . . 142
21.4 D.4 宿主 -> 设备命令 . . . 142
21.4.1 D.4.1 StartFingerDataStreaming（0x01） . . . 142
21.4.2 D.4.2 EndFingerDataStreaming（0x02） . . . 142
21.4.3 D.4.3 ZoneBoundaryRequest（0x10） . . . 143
21.4.4 D.4.4 Clear（0x20） . . . 143
21.4.5 D.4.5 SetPixel（0x21） . . . 143
21.4.6 D.4.6 DrawRectangle（0x22） . . . 143
21.4.7 D.4.7 DrawImage（0x23） . . . 144
21.4.8 D.4.8 VersionRequest（0x7F） . . . 144
21.5 D.5 设备 -> 宿主消息 . . . 145
21.5.1 D.5.1 Zone Boundary Reply . . . 145
21.5.2 D.5.2 Version Reply . . . 145
21.5.3 D.5.3 Finger Stream . . . 145
21.5.4 D.5.4 Max Num Fingers 与 Finger Data Rate . . . 146
21.6 D.6 完整示例 . . . 146
21.6.1 D.6.1 将像素 (3, 5) 点亮为红色 . . . 146
21.6.2 D.6.2 清空区域 . . . 147
21.6.3 D.6.3 以前缀 7D 01 启动手指数据流 . . . 147
21.6.4 D.6.4 在 (0, 0) 处绘制一张 [红, 绿] 的 2x1 图像 . . . 147
21.7 D.7 错误处理与限制 . . . 147
21.8 D.8 整合应用 . . . 148

1 简介



Erae 2

1.1 什么是 Erae 2?

Erae 2 是 Embodme 推出的压感 MIDI 控制器。其 42×24 的 LED 网格下方搭载了一块高分辨率的力敏电阻（Force-Sensitive Resistor）矩阵，可同时捕捉每根手指的 X、Y 和 Z（压力）数据，将整个表面变成一件由您亲手设计、充分富有表现力的乐器。无论您将它当作键盘、推子组、现场 Looper、步进音序器，还是介于其间的任何形态来演奏，Erae 2 都能适应您的工作流程。

Erae 2 专为追求超越力度（velocity）表现的音乐人打造。它支持 MPE，能够发送 CV，内置琶音器、Looper 和步进音序器，每个项目可存储八个相互独立的 Layout。配置完成后，无需连接电脑即可演奏。

1.2 主要特性

- **42×24 触控表面：** 1,008 个 LED 单元位于高分辨率 FSR 矩阵之上，每个 LED 单元配有 16 个原始感测点位，使整个表面拥有超过 16,000 个压力感测位置。
- **每根手指的 XYZ 表现力：**压力、水平滑移和垂直滑移可针对每一次同时触控独立追踪，最多支持 10 根手指。
- **10 种元素类型：** Key、Keygrid、Keyboard、Drumpad、Button（Note、Control Change、Program Change、Control Voltage、Tap Tempo）、Fader 1D、Fader 2D、Ableton Live 会话控制、API Zone 以及 Pedal。
- **每个项目 8 个 Layout：**存储并切换八个完全独立的表面配置。项目保存在 SD card 上。
- **MPE 支持：**通过 MIDI Polyphonic Expression 实现逐音符的 Pitch Bend、压力和滑移，带来富有表现力的复音演奏。
- **CV 输出：** 12 个双路插孔上共 24 路控制电压输出，用于连接模块化和模拟合成器。
- **LCD 界面：**一块 280×240 的彩色显示屏，让您无需电脑即可调整音阶、元素映射、Looper、音序器和各项设置。
- **琶音器：**一款可配置的琶音器，可作用于当前 Layout 中的任意键盘类元素。
- **Looper：**一款演奏型 Looper，可录制您的触控手势（连同其压力和滑移），并与项目时钟同步回放。
- **步进音序器：**直接在表面上绘制、录制和编辑演奏型，每个 Layout 有八个演奏型。
- **自定义音阶：**为 Keygrid 元素提供微分音和替代调律，可在 Erae Lab 或设备上编辑。

> **提示：** Erae 2 在配置完成后可独立工作。一旦您的项目已保存到设备上，演奏期间就无需开启 Erae Lab。

1.3 本测试版的新功能

与当前公开发布版相比，本测试版新增了以下内容。

步进音序器（参见步进音序器）

- **跨 Layout 停留在音序器中：**在音序器中短按另一个 Layout 按键，会带您进入该 Layout 的音序器，并定位到您上次在该 Layout 中音序化的元素。若改为按住该按键，则可预览该 Layout 并选择其他元素。
- **看到正在播放的内容：**每个带有活动序列的元素都会显示一条细细的播放头，随演奏型的节奏扫过元素。当某个您未在查看的 Layout 上正在播放序列或循环时，其 Layout 按键会亮起。
- **Pitch 泳道：** Pitch 泳道取代了 Glissando 泳道。以 bend 状态开始的音符会在其起始音高处显示一个标记，并与音符头相连。
- **手势泳道：**键盘或独奏琴键的 X、Y 和压力手势可以作为各自独立的泳道进行音序化（每个元素一条），并且在您并轨一个循环时会被保留。
- **绘制的音符带压力演奏：**您绘制的音符会将其力度保持为稳定的压力，因此即使没有绘制曲线，也能在压感合成器上发声。
- **STEPS 上的 Plus 和 Minus：**在编辑 STEPS 字段时，Plus 和 Minus 会将步数加倍或减半，范围从 1 到 256。

- **序列上的琶音器：**当某个键盘的琶音器开启时，其序列中的和弦会以琶音方式演奏。关闭琶音器即可按绘制的原样听到音符。
- **Bounce Loop 遵循 Retrig：**当键盘上的 Retrig 关闭时，并轨后的循环中的滑音会保持为一个音符。
- **一次清除循环和音序：**按住一个或多个 Layout 按钮，按下 **Return** 并保持按住。在屏幕上确认后，这些 Layout 的循环和全部八个演奏型都会被清除。

音阶与音色

- **自定义音阶适用于 Keygrid 元素：** Drumpad 始终保持半音阶的打击垫。参见音阶与调音。
- 删除某个键盘正在使用的自定义音阶时，该键盘会被设为 `Major`。当您切换到音级更少或更多的音阶时，按键会按该音阶自身的音级着色。
- **Erae Sound 槽位：**Erae Sound 预设只能从 Erae Sound 放入槽位。在预览预设时，按下 Layout 按键会带您进入该 Layout。
- 在 Erae 上对琶音器所做的更改会随 Layout 保存，并同步到 Erae Sound。

1.4 本手册的适用对象

本手册面向希望充分发挥 Erae 2 潜力的音乐人、制作人和勇于尝试的探索者。它涵盖了设备的方方面面，从在 Layout 上放置您的第一个元素，到将 CV 路由至模块化机架。阅读本手册无需任何 Erae 2 的使用经验，但若您熟悉基本的 MIDI 概念（通道、音符、Control Change）会更有帮助。

1.5 包装清单

- Erae 2 控制器，已安装好 Fabric Skin
- 65 W USB-C 电源适配器
- 8 GB SD card
- 两根 2 m USB-C 线缆
- 两个 MIDI TRS 转 DIN 适配器

> **提示：**请从 Embodme 网站下载 Erae Lab，即可在电脑上设计和管理 Layout。Erae Lab 完全免费，支持 macOS、Windows 和 Linux。

1.6 手册约定

在本手册中：

- 参数名称以**粗体**显示（例如，**Scale**、**PB Range**）。
- 参数值以 `代码格式` 显示（例如，`Chromatic`、`50%`、`MPE`）。
- 提示以引用块的形式出现，与本章中的示例相同。
- 截图通过每张图片上方的说明文字标注。
- 交叉引用会链接到其他章节（例如，Layout）或特定小节（例如，CV 时钟输出）。

- 对 Erae Lab 功能的引用以纯文本形式书写：“参见 Erae Lab 用户手册第 X 章。”
- 音符名称遵循设备和 Erae Lab 中使用的约定：MIDI 音符 60 为 C3，最低的 MIDI 音符（0）为 C-2。

2 快速入门

本章将带您完成 Erae 2 的开机、首次连线，以及演奏您的第一个 Layout，无论您是通过 USB 与 DAW 协同工作，还是脱离电脑独立演出。

2.1 开机与启动流程

Erae 2 的后面板上有三个 USB-C 端口：

- **Power**：用于连接随附的 65 W 电源适配器。
- **Computer**：即 **USB Device** 端口，将 Erae 2 作为 MIDI 控制器连接到您的电脑。
- **Synth**：即 **USB Host** 端口，将 USB MIDI 设备直接连接到 Erae 2。

插入 SD card，将随附的电源适配器连接到 **Power** 端口，然后打开设备电源。**Computer** 端口也能单独为 Erae 2 供电，但电脑端口无法为所有功能提供足够的电力（参见下文的低功耗模式）。

开机后会执行以下启动流程：

1. LED 表面点亮，与此同时固件初始化触控硬件，并从 SD card 载入您上次使用的项目。如果没有可用的 SD card，Erae 2 会回退到存储在其内部存储器中的备份副本。
2. LCD 屏幕先显示启动画面，待系统就绪后切换到 **Home** 主屏幕。
3. Layout 按键点亮，表示进入正常工作状态。

一旦出现 Home 主屏幕，设备即可演奏。

2.1.1 低功耗模式

Erae 2 需要能提供超过 15 W 的电源才能完整运行，例如随附的 65 W 电源适配器。当它由无法提供该功率的电源（例如电脑的 USB 端口）供电时，Erae 2 会显示 **Low Power** 消息，并以精简的功能运行：

- LED 亮度降低。
- CV 输出被禁用。
- MIDI 输出 B 被禁用。
- 节拍器被禁用。
- **Synth**（USB Host）端口不供电。

将随附的电源适配器连接到 **Power** 端口即可恢复完整运行。

> **提示：**如果 LED 表面仍然不亮，请检查电源适配器是否连接到 **Power** 端口，以及电源开关是否已打开。

2.2 USB 连接（兼容 Class-Compliant MIDI）

Erae 2 会被识别为兼容 class-compliant 的 USB MIDI 设备：在 macOS、Windows 10/11 或 Linux 上均无需安装驱动。使用 USB-C 转 USB-C 或 USB-C 转 USB-A 线缆，将 **Computer**（USB Device）端口连接到您的电脑。

在常规的 MIDI 1.0 模式下，您的 DAW 或 MIDI 软件会看到两个面向用户的 USB MIDI 端口：

线缆	端口名称	用途
主（Main）	`Erae 2 MIDI`	标准 MIDI：用于非 MPE 乐器的音符输出以及常规 DAW 使用
MPE	`Erae 2 MIDI (MPE)`	用于兼容 MPE 的乐器（如 Equator2、Pigments、Omnisphere）的逐音符表

在 DAW 的 MIDI 输入设置中选择 **`Erae 2 MIDI`**（主），即可接收来自表面的音符、力度、Pitch Bend 以及连续控制器消息。对于支持 MPE 的乐器，请改为将乐器指向 **`Erae 2 MIDI (MPE)`**。

MIDI 2.0 是另一种 USB 模式，通过 **Settings** 中的 **MIDI 2.0** 一行控制。更改它时会要求确认并重启设备。除非收到 Embodme 的指示，或某个特定的 Erae Sound 或 Erae Lab 工作流程要求，否则大多数 DAW 用户应保持 MIDI 2.0 关闭。

> **提示：**在 macOS 上，打开 *Audio MIDI Setup -> MIDI Studio* 以确认设备已正确枚举。如果固件更新后设备带有数字后缀（如 `Erae 2 2`），请打开 Audio MIDI Setup，选中旧条目并将其删除。重新插拔 USB 即可恢复干净的枚举。

> **从固件 1.x 更新：端口布局变更。** 2.x 固件移除了专用的第三条 “Lab/API”MIDI 线缆。Erae Lab 现在使用其自己的 USB 连接，该连接不会作为 MIDI 端口出现在您的 DAW 中。更新后只有两个 MIDI 端口：
>
> | 2.x 之前（三个端口） | 2.x 之后（两个端口） |
> |---|---|
> | MIDI 端口 1：`Erae MIDI`（主） | `Erae 2 MIDI`（主，功能相同） |
> | MIDI 端口 2：`MIDI (Lab)` | （已移除） |
> | MIDI 端口 3：`MIDI (MPE)` | `Erae 2 MIDI (MPE)`（原端口 3，现为端口 2） |

> 如果您的 DAW 或 API 设置曾将旧的第三端口用于 MPE 数据，请将其重新映射到 **`Erae 2 MIDI (MPE)`**（第二个端口）。如果您曾按编号引用 Lab/API SysEx 端口，请改用 **`Erae 2 MIDI`**（第一个端口）：API SysEx 现在通过主线缆传输。

2.2.1 在 DAW 中将 Erae 与 Erae Sound 配合使用

Erae Sound 以 VST3/AU 插件形式提供。当您在 DAW 乐器轨道上加载它时：

- 推子手势和基础键盘音符开/关数据通过 **`Erae 2 MIDI`** 端口从 Erae 发送。请在插件轨道上启用该端口的输入（大多数 DAW 需要您将轨道的 MIDI 输入明确设置为此端口）。
- 要获得完整的 MPE 表现力（逐音符 pitch bend、压力和滑移），请在乐器轨道上启用 **MPE 模式**，并将其指向 **`Erae 2 MIDI (MPE)`**。滑移（CC 74）需要 MPE 端口，且要求宿主将全部 16 个通道传递给乐器。
- **不要**在使用插件的同时在 Erae Lab 中运行 Erae Sound 引擎。两个实例同时监听相同的 Erae MIDI 输入；音符和手势会同时到达两个引擎，产生双重触发和音频串扰。

> **仅 DAW 播放时静音内置引擎。** Erae Sound 作为内置音频引擎运行于 Erae Lab 之中。要仅通过 DAW 路由音频，请点击 Erae Lab 顶部标题栏中 **Sound** 标签页上的电源图标以停止该引擎。Layout 编辑和设备同步保持有效。再次点击该图标即可重新启动引擎。详情请参阅 Erae Sound 手册。

2.3 MIDI 输入与输出

Erae 2 在 3.5 mm TRS 插孔上配有一个 MIDI 输入（**MIDI In**）和两个 MIDI 输出（**MIDI A** 和 **MIDI B**），可在无需电脑的情况下连接硬件合成器、鼓机和效果器。MIDI B 在低功耗模式下被禁用。

使用随附的 TRS 转 DIN 适配器即可连接标准的 5 针 DIN MIDI 设备：将它们插入 **MIDI In** 和 **MIDI A**，然后用标准 5 针 MIDI 线缆连接到您的设备。

每个元素可选择由哪些输出承载其消息（参见逐元素输出目标），路由矩阵则在端口之间转发传入的 MIDI（参见MIDI 路由矩阵）。

2.4 USB Host 端口

Synth（USB Host）端口允许 Erae 2 充当 USB 主机，在无需电脑的情况下与兼容 class-compliant 的 USB MIDI 设备通信：例如带 USB MIDI 的硬件合成器、MIDI 控制器或 USB 转 DIN 适配器。该端口可为所连接的设备提供最高 5 W（5 V，1 A）的电力，且仅在电源适配器连接到 **Power** 端口时供电。

连接到 USB Host 端口的设备会在元素路由和 MIDI 路由矩阵中显示为 **USB Host** 目标。您可以将表面的触控事件直接发送到所连接的合成器，或将来自合成器的传入 MIDI 转发到您的电脑。

> **提示：**您可以同时使用 **Computer** 和 **Synth** 端口。

2.5 您的第一个 Layout

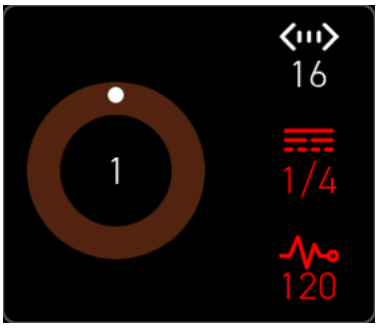
Erae 2 出厂时带有一个工厂项目，它填满了全部八个 Layout 槽位。每个 Layout 槽位都通过标号为 **N1** 至 **N8** 的 Layout 按键选择。

要选择一个 Layout，请按下 **N1** 至 **N8** 中的某个按键。LED 表面会重绘以显示该 Layout，相应按键以全亮度点亮以确认选择。

工厂 Layout 包括：

- 一个组合 Layout，包含鼓垫、钢琴式键盘、一个大型表现力琴键和若干推子
- 钢琴式键盘
- Keygrid（同构）Layout，其中部分设置为 MPE
- 一个调音台式 Layout，包含音符按钮、推子和一个 XY 面板
- 由独立表现力琴键组成的网格
- 一个大型鼓垫网格

各工厂 Layout 使用不同的 MIDI 通道，MPE Layout 通过 `Erae 2 MIDI (MPE)` 端口发送。每个工厂 Layout 在您切换到它时，还会请求 Erae Sound 载入一个匹配的预设。您无需连接 Erae Lab，即可立即在这些 Layout 上探索、演奏与演出。



LCD 主屏幕

LCD 上的 **Home** 主屏幕显示当前 Layout 的 Looper：一个显示 Layout 编号和循环播放位置的圆环，以及循环的 **Length**、**Quantize** 网格和 **Tempo**。参见琶音器与 Looper。

> **提示：**使用专用的 **N1** 至 **N8** Layout 按键，以便在现场进行可预期的切换。

2.6 独立模式与 Lab 连接模式

Erae 2 根据 Erae Lab 是否正在运行并通过 USB 连接，工作于两种模式之一。

2.6.1 独立模式

当没有连接电脑，或 Erae Lab 未打开时，Erae 2 运行于独立模式。在此模式下：

- 设备播放 SD card 上最后保存的项目。内部存储器保存的是备份副本和设备设置，而非常规的项目库。
- 全部八个 Layout 均可用且功能完整。
- 每个元素发送到其路由中所选的输出（默认为 USB Device 端口 `Erae 2 MIDI`）。
- 您在设备上对存储于 SD card 中的项目所做的编辑会自动保存。工厂项目和备份项目在您将其保存到 SD card 之前不会被保存（参见项目管理）。

独立模式专为无需笔记本电脑的现场演出而设计。用随附的电源适配器为设备供电，Erae 2 即可完全独立运行。

2.6.2 Lab 连接模式

当 Erae Lab 在您的电脑上打开并检测到 Erae 2 时，设备进入 Lab 连接模式。在此模式下：

- Erae Lab 可以向设备发送新的 Layout 和项目配置。
- 在 Erae Lab 中所做的编辑会立即反映到表面上，无需重启。
- 您可以将项目从 Erae Lab 保存到 SD card，以供日后独立使用。

> **提示：**当 Erae Lab 检测到设备时，Lab 连接模式会自动激活。打开 Erae Lab，并将 USB-C 线缆连接到 **Computer** 端口即可。

有关 Lab 工作流程、项目传输以及固件更新的完整细节，请参阅连接到 Erae Lab。

2.7 后续步骤

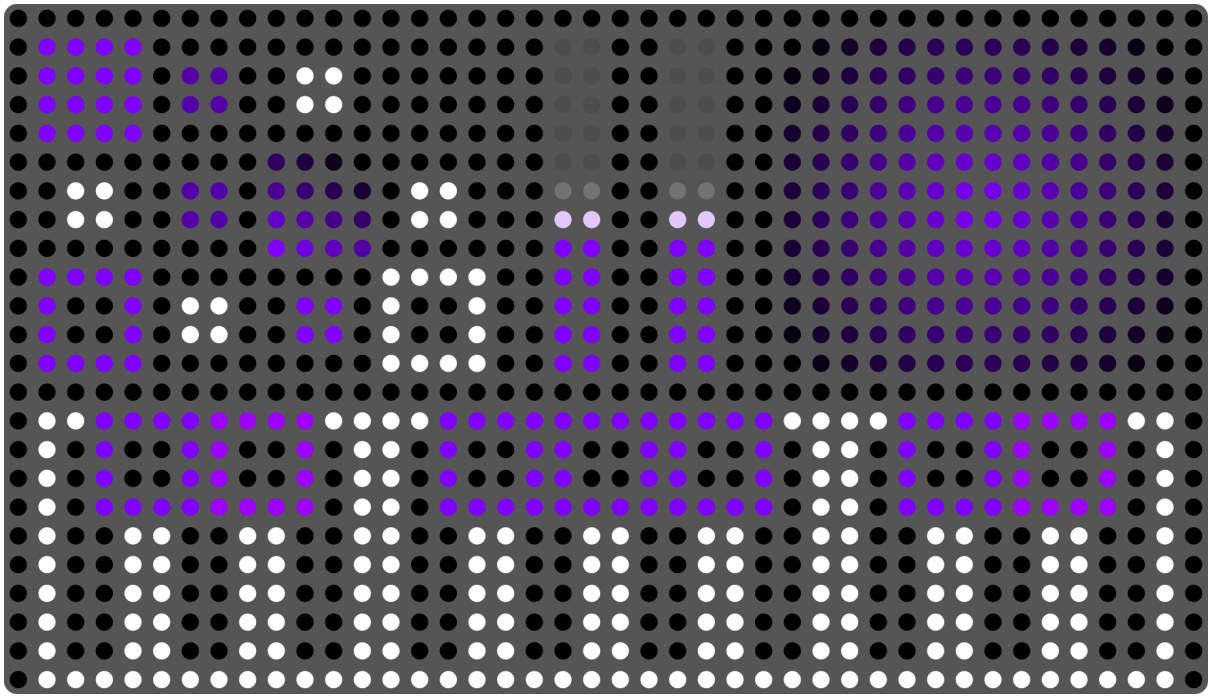
在您完成设备开机，并确认某个工厂 Layout 能够输出 MIDI 后，您就可以深入探索表面了。请继续阅读触控表面与触控，了解压力、位置和复音触控如何转化为富有表现力的 MIDI 数据。

3 第 3 章：触控表面与触控

Erae 2 围绕一块大型压力感应触控板打造，能够捕捉您演奏中的每一处细微变化。本章介绍触控表面的工作原理、它产生的数据，以及如何针对您的演奏技巧调校其行为。

3.1 触控板网格

演奏表面以 **42 × 24** 的 LED 网格组织而成（横向 42 列、自上而下 24 行），其背后由一块高分辨率的力敏电阻（FSR）矩阵支撑。每个 LED 单元由 16 个原始感应点支持，使整个演奏区域拥有超过 16,000 个压力感应位置，同时让视觉反馈与物理表面保持对齐。



LED 触控反馈

物理网格是连续的：触控区、推子和琴键可以横跨任意数量的单元，因此 Layout 不受固定按键间距的限制。当手指落在单元之间时，固件会将位置插值到亚单元级的精度，从而带来平滑、高分辨率的控制。

> **提示：**更宽的元素（横跨更多单元）会给您更大的滑动空间，并产生更平滑的 X/Y 数据。只有在需要将许多区域塞进狭小空间时，才使用较窄的元素。

3.2 XYZ 感应

每一次触控都会同时报告三个相互独立的维度：

- **X 位置：**手指在其所在元素内的水平位置，归一化为 `0.0` 到 `1.0`（从左到右）。向左或向右滑动会产生 X 方向的变化。
- **Y 位置：**手指在元素内的垂直位置，归一化为 `0.0` 到 `1.0`（从下到上）。向上或向下滑动会产生 Y 方向的变化。
- **Z 压力：**施加在表面上的力度，归一化为 `0.0` 到 `1.0`。按得越用力，Z 值越大。

触控引擎还会根据手指在表面上的连续移动推导出**运动速度（Motion Speed）**。运动速度的范围为 `0` 到 `100 cm/s`，可用于驱动 CC。

Z 经过缩放，使得在舒适的演奏力度范围内，稳健的按压即可达到 `1.0`。

> **提示：**在琴键元素上使用 **X Position** 和 **Y Position** 可同时驱动 pitch bend 与调制：这是富有表现力的 MPE 演奏的核心。MPE 配置请参阅第 6 章。

表面以**每秒 1,600 次**的速率进行扫描，并被转化为稳定、富有表现力的 X/Y/Z 手指数据。

3.3 10 指多点触控

Erae 2 可在整个表面上追踪多达 **10 根同时触控的手指**。每根手指都会被独立识别，并在其接触期间被分配一个稳定的追踪 ID，因此密集的和弦与双手手势都能获得各自独立的连续 X、Y、Z 数据流。

表面会持续自我校准，因此无需等待任何预热。短暂的压力下降不会释放手指，从而防止意外的 note-off。

两根靠得非常近（距离小于约一个半单元）的手指会被视为一次触控，直到它们分开为止。

> **提示：**将双手铺在表面上即可演奏簇状和弦：只要触控点在物理上没有相互重叠，最多可独立追踪 10 根手指。

3.4 力度曲线设置

设备上的 **Velocity Curve** 编辑器位于 **Settings** 中，用于塑造触控动态如何转化为 MIDI 音符力度。它是一项全局演奏响应设置，而非逐元素的压力曲线。

- **阈值（Threshold）**设定在力度开始上升之前所需的接触量。
- **驱动（Drive）**改变曲线中段达到较高力度的速度。
- **压扩（Compand）**对响应形状进行压缩或扩展。
- **范围（Range）**设定曲线的整体灵敏度：数值越高，达到满力度所需的压力越小。它并不是对输出数值的封顶限制。

调整这些数值时，LCD 上的曲线预览会实时更新。借助它可将 Erae 2 与您的演奏风格相匹配：较轻的设置适合细腻的手指演奏，较硬的设置则适合您希望在达到高力度前获得更多阻力的情况。

> **提示：**如果音符过于轻易地跳到最大力度，可降低 **Range** 或提高 **Threshold**。如果音符感觉过于柔弱，可提高 **Range** 或 **Drive**。

3.5 灵敏度

Sensitivity 设置（位于 **Settings** 中）控制整块触控板对触控的整体响应难易程度。共有四种模式可选：

模式	说明
XSensitive	响应最灵敏：对最轻微的接触也会作出反应。最适合触感非常轻的演奏者。若不慎触碰表面可能产生误
Sensitive	灵敏：激活阈值低于默认值。
Safe	默认设置：阈值均衡，适合大多数演奏风格与环境。
XSafe	响应最不灵敏：需要更稳健的接触才会触发。可减少由衣物、线缆或轻微意外接触引起的误触发。

> **提示：**如果您在演奏时遇到幻影音符或误触发，请从 `Safe` 切换到 `XSafe`。如果表面对轻柔演奏反应迟钝，可尝试 `Sensitive` 或 `XSensitive`。对于初次接触基于 FSR 表面的演奏者而言，灵敏度是最有效的单项调校选项。

3.6 物理保养

Erae 2 可以安装不同的演奏 Skin。请将 Skin 视为乐器的一部分，而非可随意更换的鼓皮。

- **Fabric Skin：**专为细腻的手指演奏和富有表现力的表面控制而设计。请尽量保持双手清洁，避免刮擦或硬性撞击，并用蘸有异丙醇的柔软无绒布小心清洁污渍。请勿浸湿表面。
- **Drum Skin / 黑色 Skin：**用略微湿润的无绒布擦拭，存放前请将其擦干。这种 Skin 可以用鼓棒演奏，但 Erae 2 仍是一款电子控制器，而非原声鼓。大力击鼓、过于尖锐的鼓棒角度或反复的高强度撞击都可能在 Skin 上留下痕迹，并可能缩短产品的预期使用寿命。
- **所有 Skin：**让尖锐物体、戒指、拨片和磨蚀性材料远离表面。避免过热和长时间的直射阳光。不使用时请将乐器存放在保护箱中。
- **演奏之前：**移除搁在表面上的线缆、工具和其他物品，让触控引擎从干净的接触状态开始工作。

4 Layout

Layout 是项目的基本构建单元。它定义了 Erae 2 表面的行为方式：哪些元素放置在哪里、它们如何响应触摸，以及发送哪些 MIDI 消息。切换 Layout 让您能够瞬间改变整个表面的配置，从一个 Layout 中的半音键盘，切换到下一个 Layout 中一组压力感应的打击垫。

4.1 什么是 Layout?

每个 Layout 都是表面的一个独立快照。它存储了：

- 放置在表面上的每一个**元素**（键盘、推子、按钮等等）
- 每个元素的**位置**和**尺寸**
- 所有逐元素的 **MIDI 参数**（通道、音符范围、CC 分配、MPE 设置等）
- 每个键盘元素的**音阶**和基准音
- 每个元素的 **LED 颜色**和动画
- 两个踏板输入的**踏板**配置
- 它自己的 Looper 循环和步进音序器演奏型

当您切换 Layout 时，上一个 Layout 中的任何元素数据都不会延续过来。每个 Layout 都是完全自包含的。

> **提示：**可以把 Layout 想象成 “场景”：您可以将一个 Layout 专门用于旋律演奏，另一个用于节奏打击垫，第三个用于宏控制，然后在它们之间切换而无需中断您的音乐。

4.2 每个项目的 Layout 数量

每个项目恰好包含 **8 个 Layout**，编号从 1 到 8。所有 8 个槽位始终存在；任何尚未配置的槽位就是空的（表面为空白，不产生任何输出）。

您无需填满全部 8 个槽位。空的 Layout 槽位是一种有效、可演奏的状态，可用作瞬时静音或演奏暂停。

> **提示：**把空的 Layout 槽位用作有意为之的“静默”场景。切换到它会干净利落地切断所有正在进行的触摸事件。

4.3 在 Erae 2 上切换 Layout

使用专用的 **N1** 至 **N8** 面板按键来切换 Layout。Layout 在您松开按键时切换。空槽位是有效的可演奏状态，可用作静默或静音场景。

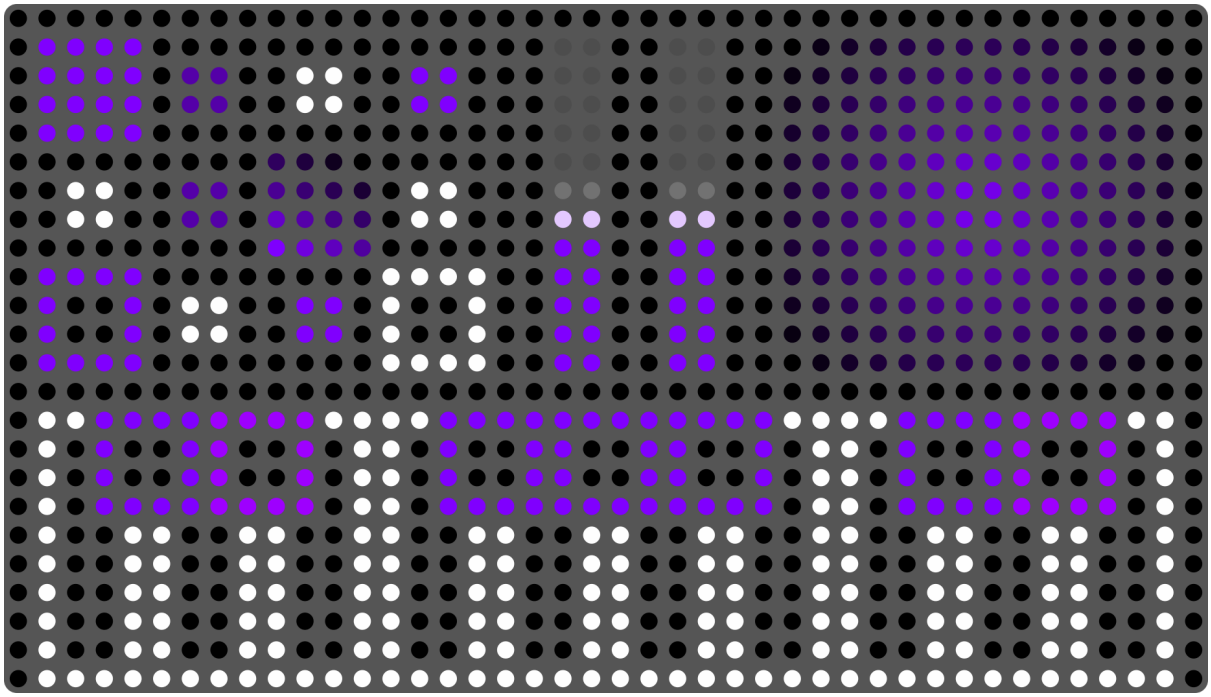
Home 主屏幕会在 Looper 圆环内显示当前 Layout 的编号。Erae 2 上没有单独的 LCD Layout 选择器，也没有下一个/上一个 Layout 的操作流程。

在手指按住时切换 Layout，会释放旧 Layout 上的触摸，并把按住的手指重新引入新 Layout。这可以防止在演奏中切换 Layout 时出现卡音和幻影触摸。

4.4 LED 指示灯

N1 至 **N8** Layout 按键显示每个 Layout 槽位的状态：

- **当前 Layout** 的按键以全亮度点亮。
- 其他 Layout 按键微弱发光，当该 Layout 上的某个元素触发音符时（例如在 Looper 或音序器回放期间）会短暂变亮。
- 循环被静音的 Layout 以红色显示（参见琶音器与 Looper）。



混合 Layout LED

这让您即使在昏暗的演奏环境中，也能一眼看清自己的当前位置以及哪些 Layout 正在播放。

4.5 在 Erae Lab 中编辑 Layout

Layout 在您电脑上的 **Erae Lab** 中创建和编辑。在 Erae Lab 内，您可以：

- 将元素拖到表面画布上并自由摆放它们
- 配置每个元素的 MIDI 输出、音阶和视觉颜色
- 为每个 Layout 命名，便于识别
- 在项目内复制、粘贴和重新排序 Layout
- 在发送到设备之前预览 Layout 的 LED 外观

当 Erae Lab 连接时，Layout 编辑会自动保存，并通过后台同步链接流向设备。通常您无需在每次编辑后手动执行推送或拉取。在断开 USB 之前，请等待传输活动完成。

在设备本身上，您可以调整键盘的音阶和基准音（参见音阶与调音）、Mapping 画面中的元素设置，以及琶音器、Looper 和音序器。这些编辑会随项目一起保存。

5 元素

元素是 Layout 的基本构建模块。每个元素占据触控面板上的一个矩形区域，定义它生成何种 MIDI（或 CV）数据，并决定其下方的 LED 如何渲染。一个 Layout 可以包含任意组合的并排放置的元素，让您能够完全自由地组织自己的演奏面板。

每个元素都有三个共有属性：

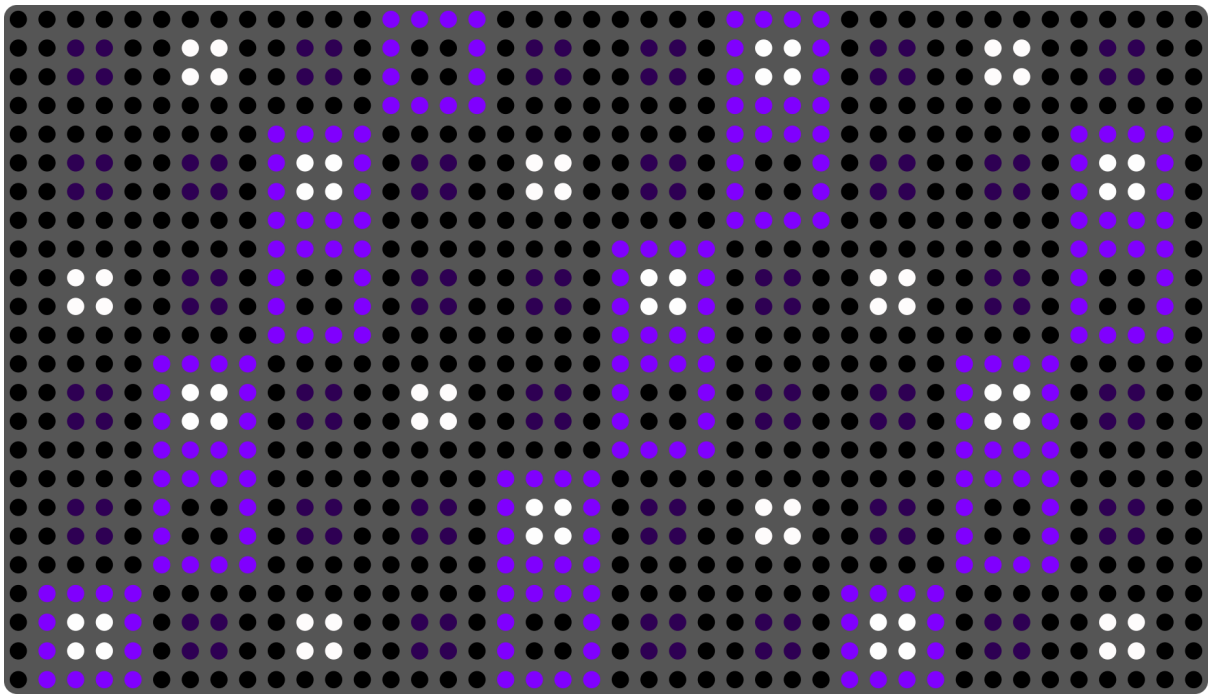
- **几何（Geometry）：**在面板网格上的位置与尺寸（x、y、宽度、高度）
- **样式（Style）：** LED 颜色、渐变，或由行为驱动的可视化
- **动画（Animations）：**在您触摸、滑动或松开时触发的可选 Finger Glow、Ripple、Halo 或 Color Morph 效果

元素专属参数（MIDI 通道、CC 编号、音符分配、表现力设置）都是按元素逐个配置的，不会影响同一 Layout 中的其他元素。每个生成 MIDI 的元素还会选择自己的输出端口（参见逐元素输出目标）。

元素在 Erae Lab 中创建和编辑，它们在元素面板中显示为 **Key**、**Keygrid**、**Keyboard**、**Drumpad**、**Button**、**Fader 1D**、**Fader 2D**、**LivePad** 和 **API zone**。踏板输入则为每个 Layout 单独配置。关于 Layout 的工作方式，请参阅第 4 章；关于与 Erae Lab 的连接，请参阅第 14 章；关于在 Erae Lab 中配置每种元素类型的完整指南，请参阅 Erae Lab 用户手册第 7 章。

本章中的音符名称遵循 Erae 约定：MIDI 音符 60 为 C3。

5.1 5.1 Iso Keyboard



Iso Keyboard

Iso Keyboard（在 Erae Lab 中称为 **Keygrid**）是一种同构网格乐器：无论您处在面板的哪个位置，任意两个相邻琴垫之间都存在相同的音程关系。任何指法或和弦形状在每个调性中都以完全相同的方式工作，使得移调变得轻而易举，音阶模式也可以即时迁移。

音符排列在一个由琴垫组成的矩形网格中，每个琴垫都具备完整的逐音符表现力：压力、水平滑动（Pitch Bend 和滑音）、垂直滑动，以及 CC 和 CV 输出。

Layout 与调音。向右移动一个琴垫会演奏下一个音符。向上移动一行会按行偏移量跳跃。开启 **Show Offscale** 时，每个半音都有一个琴垫，行跳跃为 **Semitones**（默认 5，即纯四度）。关闭 **Show Offscale** 时，只有音阶音拥有琴垫，行跳跃为 **Degrees**（默认 3 个音阶音级）。在 Erae Lab 中更改这两个偏移量之一，会相应地切换 **Show Offscale**。左下角琴垫的基准音由根音和八度设置（默认 C2）。

Key Width / Key Height 控制每个琴垫占据多少个面板单元。更大的琴垫会让每个音符拥有更大的压感区域。

MPE。启用 **MPE** 后，每一次同时触摸都会被分配到各自的 MIDI 通道，使逐音符的 Pitch Bend、压力和滑动能够无串扰地到达支持 MPE 的合成器。**MPE Master** 通道可以是 `Channel 1` 或 `Channel 16`。开启 MPE 时，**Channel** 设置变为 **Nb channel**，即成员通道的数量（1 到 15）。

自定义音阶。启用了 MPE 的 Keygrid 可以使用自定义音阶。参见音阶与调音。

琶音器（Arpeggiator）。 Iso Keyboard 内置琶音器。完整细节请参阅第 8 章。

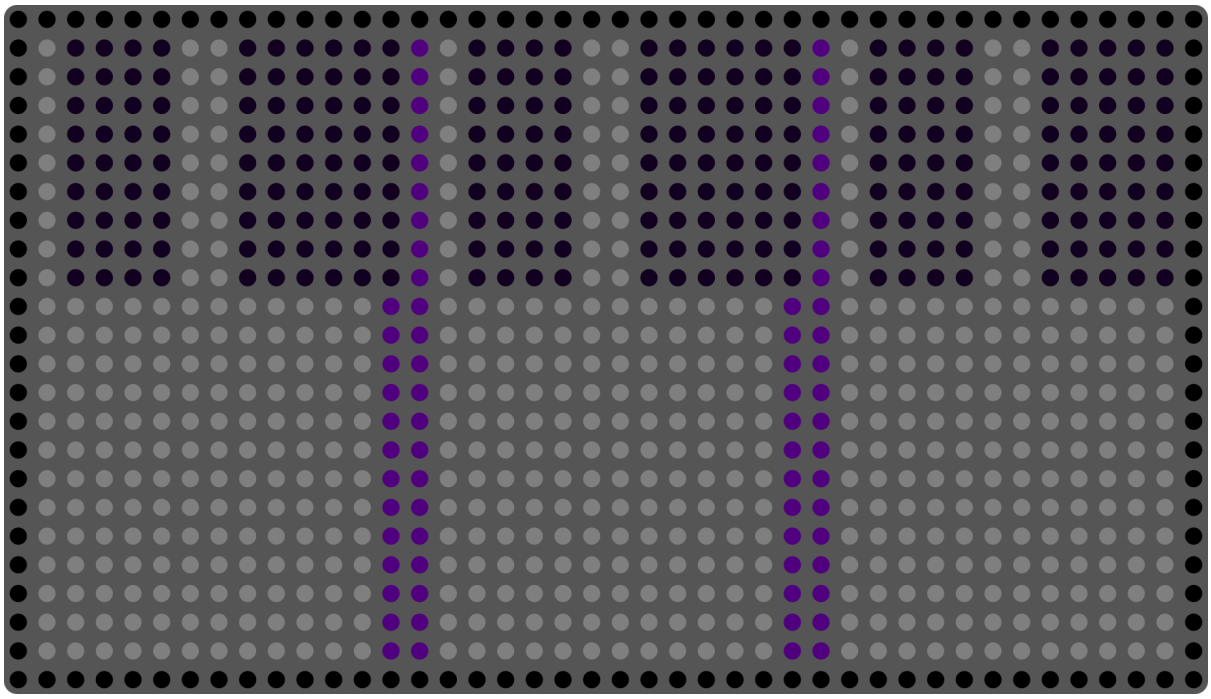
LED 可视化。音阶音级琴垫从音阶音级样式中获取颜色（每个音级一种样式）。音阶外的琴垫使用音阶外样式。当某个琴垫被触摸时，会根据该元素的 Animation 设置播放动画。

参数	说明	范围	默
Root note / Octave	左下角琴垫的音高	`C-2` 到 `G8`	C
Scale	当前激活的音乐音阶	14 种工厂音阶，或开启 MPE 时的自定义音阶	M
Show Offscale	显示每个半音，或仅显示音阶音	On / Off	C
Semitones	开启 Show Offscale 时的行跳跃	1 到 48	5
Degrees	关闭 Show Offscale 时的行跳跃	1 到 48	3
Key Width	每个琴垫的宽度（面板单元数）	单元数	4
Key Height	每个琴垫的高度（面板单元数）	单元数	4
Use global scale	跟随项目全局音阶	On / Off	C
PB Range	以半音为单位的 Pitch Bend 范围	1 到 96	1
MPE	启用 MPE 逐音符通道	On / Off	C
MPE Master	MPE 区域的主通道	`Channel 1` / `Channel 16`	C
Channel	MIDI 通道（开启 MPE 时为 Nb channel）	1 到 16	1

> **提示：**在同构布局中，同一个和弦形状（比如大三和弦）无论您的手指放在哪里，都以相同的方式工作。在一个位置练习和弦配置，然后就能自由地平移它们。

> **提示：**保持 **Show Offscale** 关闭，并将 **Scale** 设置为 `Blues` 或 `Japanese`。每个单元都演奏音阶内的音符，非常适合即兴演奏而不会弹错音。

5.2 5.2 Chroma Keyboard



Chroma Keyboard

Chroma Keyboard（在 Erae Lab 中称为 **Keyboard**）在触控面板上呈现传统的钢琴式布局，包含白键和黑键。这种布局对钢琴演奏者来说一目了然，当您希望以标准键盘的方式思考时非常有用。

与物理钢琴不同，每个琴键都具有完整的压感能力，沿琴键滑动会在该音符上产生 Pitch Bend 和滑动手势。

Key Width 设置每个琴键占据多少个面板单元。可见琴键的数量会随元素的宽度而变化。**Line offset** 设置上下叠放的琴键行之间的音高跳跃。

Scale 设置仅用于为琴键着色：根音、音阶内和音阶外的琴键各自获得自己的样式，但每个半音都保持可演奏。Chroma Keyboard 始终使用十二平均律，因此不能使用自定义音阶或全局音阶。

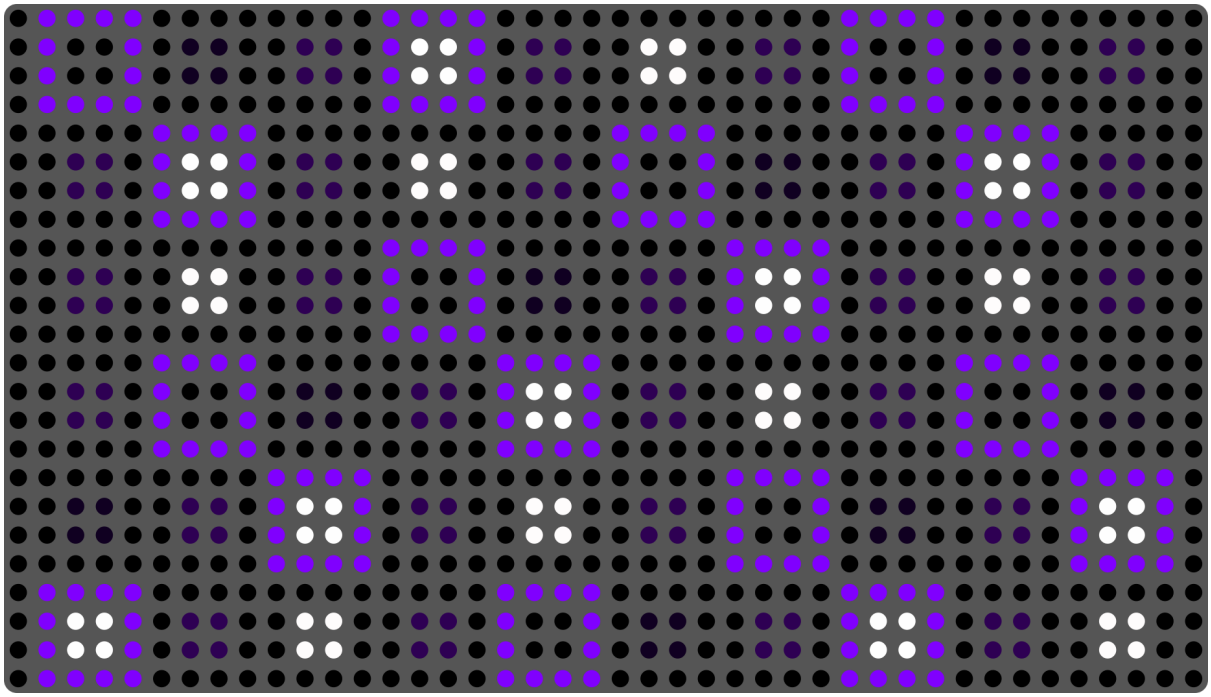
Glissando 控制手指在琴键间滑动时音高如何跟随。**Sound Brightness (CC 74)** 部分将一种手势（压力、X 或 Y 位置、相对 X 或 Y，或运动速度）映射到 CC 74。参见MIDI 配置。

参数	说明	范围	默认值
Octave	最低琴键的八度	-2 到 8	2（最低琴键为 C2）
Scale	用于着色的音阶	15 种工厂音阶	Chromatic
Key Width	每个琴键的面板单元数	单元数	4
Line offset	行之间的音高跳跃	1 到 48	
PB Range	以半音为单位的 Pitch Bend 范围	1 到 96	12（MPE 时为 48）
MPE	启用 MPE 逐音符通道	On / Off	Off

参数	说明	范围	默认值
Channel	MIDI 通道（开启 MPE 时为 Nb channel）	1 到 16	1
Y rel CC	以 CC 形式输出的相对垂直滑动	CC 或禁用	CC 74

> **提示：**在 Layout 底部放置一个横跨整个宽度的 Chroma Keyboard 作为演奏键盘，然后在上半部分添加推子或按钮作为调制控制，全部都在同一个 Layout 之内。

5.3 5.3 Drumpad



Drumpad

Drumpad 是一种针对打击乐优化的键盘变体。它使用与 Iso Keyboard 相同的网格结构，但始终为半音排列：每个琴垫演奏下一个 MIDI 音符，不进行音阶筛选。这使得将琴垫映射到鼓机音色变得非常直接，每个音符对应一种乐器。

排列方式。琴垫从左到右、从下到上排列。第一个琴垫（左下角）演奏基准音，其右侧的每个琴垫演奏下一个半音。每向上一行按行偏移量跳跃（默认 4，因此 4 个琴垫宽的网格演奏连续的音符）。General MIDI 鼓映射从音符 36（C1）开始，这也是默认的基准音。

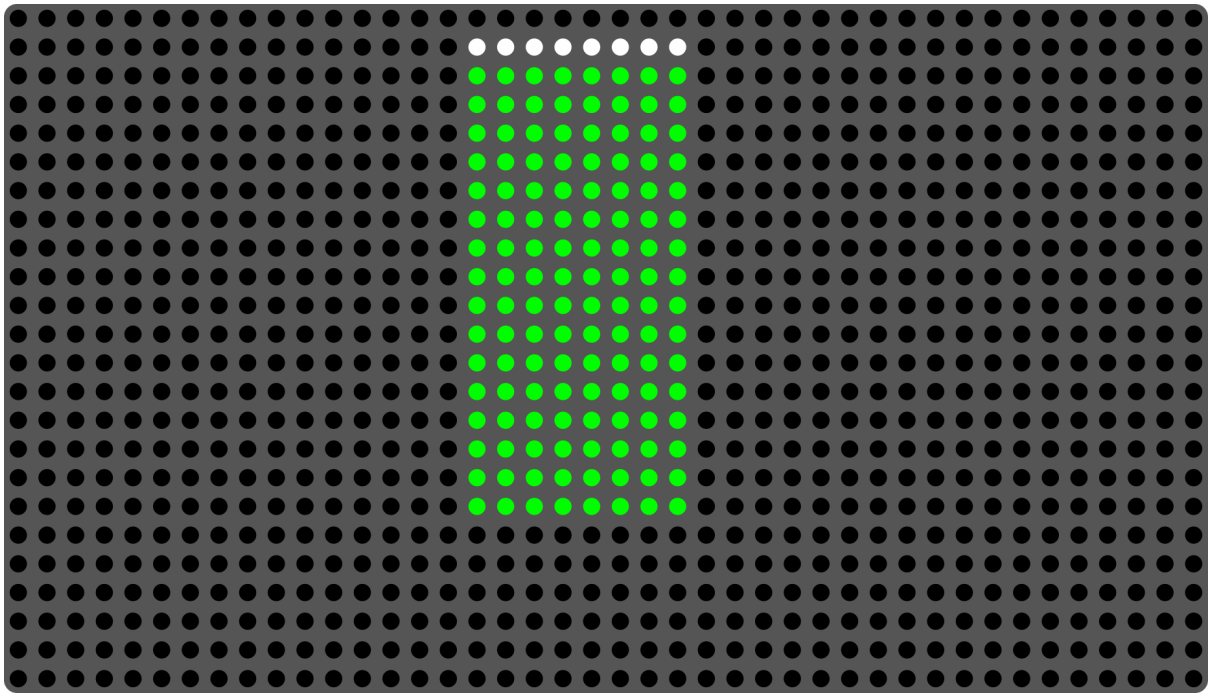
LED 可视化。每个琴垫从对应其半音位置（C 到 B）的样式中获取颜色，让您可以为相关音色进行颜色编码。

Drumpad 可以跟随项目全局音阶设置（**Use global scale**），但它始终保持半音排列；自定义音阶不适用。

参数	说明	范围	默认值
Root note / Octave	第一个（左下角）琴垫的音符	`C-2` 到 `G8`	`C1`
Key Width	每个琴垫的宽度（面板单元数）	单元数	4
Key Height	每个琴垫的高度（面板单元数）	单元数	4
Channel	所有琴垫的 MIDI 通道	1 到 16	1

> **提示：**要得到经典的 4×4 鼓垫网格，将元素尺寸设置为 8×8 个单元，并将 **Key Width** 和 **Key Height** 设置为 2。您将获得 16 个琴垫，每个都占据宽裕的二乘二单元区域。

5.4 5.4 Fader 1D



Fader 1D

Fader 1D 是一种单轴连续控制器，它追踪手指在元素边界内的垂直位置。当您将手指从底部滑向顶部时，它会发送一个从 `0` 到 127 的 CC 值。LED 填充始终显示当前值，即使没有被触摸时也是如此。

绝对位置。推子输出一个绝对的 Y 位置：该值直接对应于您的手指在面板上的位置，而不是它移动了多远。将手指放在新位置时，值会跳到该位置。

压力与运动。可选的次要 CC 可以跟随手指压力（**Pressure**）和手指移动速度（**Motion**），从而让同一个推子同时输出值和表现力。

CV 输出。 CV 输出可以跟随 Y 轴（**Y abs**）和压力（**Pressure**），使 Fader 1D 无需 MIDI 转 CV 转换器即可用于模块化合成器场景。参见第 10 章。

默认值。 **Default Y** 设置加载 Layout 时推子的起始值。默认值为 63（中心）。

中心值。 **Center Y** 设置 LED 填充所使用的视觉零点。填充绘制在中心值和当前值之间，因此中心值为 `0` 时看起来像一个从底部开始填充的传统推子，而中心值为 63 时则会创建一个从中间向上或向下填充的双极性推子。

当您在 Erae Lab 中创建推子时，其 CC 会自动设置为该通道上尚未使用的控制器编号。

参数	说明	范围	默认值
Y abs CC	垂直位置的 CC	CC、14 位 CC、RPN 或 NRPN	第一个未使用的 CC
Default Y	加载 Layout 时的起始值	`0` 到 127	63
Center Y	LED 填充的视觉零点	`0` 到 127	`0`
Pressure CC	手指压力的可选 CC	控制器或禁用	禁用
Motion CC	手指移动速度的可选 CC	控制器或禁用	禁用
Y abs CV	垂直位置的 CV 输出	输出或禁用	禁用
Pressure CV	压力的 CV 输出	输出或禁用	禁用
Channel	MIDI 通道	1 到 16	1

> **提示：**将多个 Fader 1D 元素并排放置，各控制不同的 CC，即可搭建一组调音台推子或发送电平。

5.5 5.5 Fader 2D

Fader 2D 将水平（X）和垂直（Y）方向的手指位置追踪为两个独立的绝对 CC 值，使其成为一个 XY 触控板。在元素内任意位置触摸，会将两个轴都设置为该位置。LED 可视化指示当前的 X/Y 位置，并且可以从可配置的中心点开始渲染。

双 CC 输出。 **X abs** 和 **Y abs** 是独立分配的。一种常见的分配方式是在 X 轴上控制滤波器截止频率，在 Y 轴上控制共振。

压力与运动。可选的 **Pressure** 和 **Motion** CC 增加了更多的控制维度。

CV 输出。两个轴以及压力都有对应的 CV 输出（**X abs**、**Y abs**、**Pressure**），可从单个元素实现三轴 CV 控制。

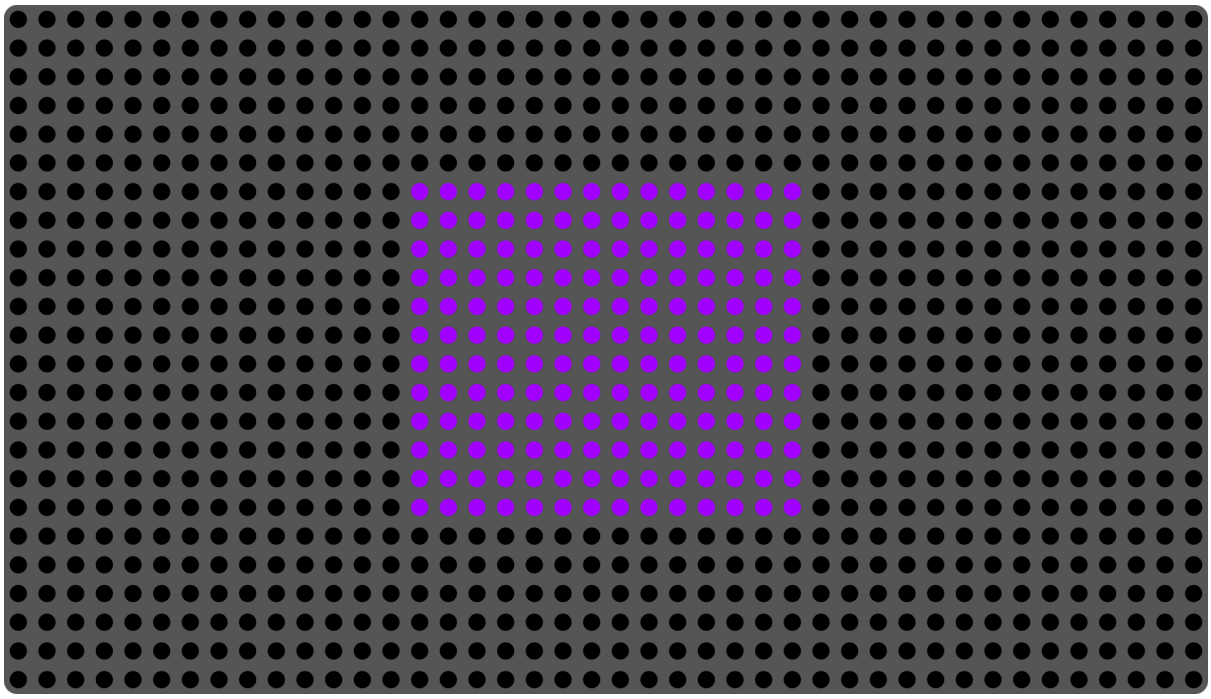
默认位置。 **Default X** 和 **Default Y** 设置加载 Layout 时的起始坐标（默认 63，中心）。

中心位置。 Center X 和 **Center Y** 设置 XY 填充所使用的视觉零点。激活区域绘制在中心点和当前位置之间。使用默认中心 `0`，`0` 时，触控板从左下角开始填充。将中心设置为 63，63 会创建一个中心止位的 XY 触控板：向右和向上移动填充一个象限，向左和向下移动填充相反的象限，而一个暗淡的中心十字准线会保持可见作为参考。

参数	说明	范围	默认值
X abs CC	水平位置的 CC	CC、14 位 CC、RPN 或 NRPN	第一个未使用的 CC
Y abs CC	垂直位置的 CC	CC、14 位 CC、RPN 或 NRPN	第一个未使用的 CC
Default X	加载 Layout 时的起始 X	`0` 到 127	63
Default Y	加载 Layout 时的起始 Y	`0` 到 127	63
Center X	X 渲染的视觉零点	`0` 到 127	`0`
Center Y	Y 渲染的视觉零点	`0` 到 127	`0`
Pressure CC	手指压力的可选 CC	控制器或禁用	禁用
Motion CC	手指移动速度的可选 CC	控制器或禁用	禁用
X abs / Y abs / Pressure CV	CV 输出	输出或禁用	禁用
Channel	MIDI 通道	1 到 16	1

> **提示：**将 Fader 2D 用作合成器滤波器的演奏 XY 控制器（X 控制截止频率，Y 控制共振），同时利用压力维度来调制驱动或饱和度。

5.6 5.6 Button {#56-button}



按钮

Button 元素是一个离散触发器，在被触摸时发送一条预定义的 MIDI（或 CV）消息。它有五种类型：**Note**、**Control Change**、**Program Change**、**Control Voltage** 和 **Tap Tempo**。**Latched** 选项让按钮在两次按下之间切换状态，而不是充当瞬时触发器。

瞬时（Momentary）与锁定（Latched）。在瞬时模式下（默认），按下时发送 “开” 消息，松开时发送 “关” 消息。在锁定模式下，第一次按下发送 “开” 消息，按钮保持其状态；下一次按下发送 “关” 消息。

Note。按下时以固定的中等力度发送 Note On，松开时发送 Note Off。可以在 MIDI 音符之外启用一个 CV 输出（门信号和 1 V/octave 音高）。

Control Change。按下时发送 **On** 控制器及其值，松开时（锁定时为第二次按下时）发送 **Off** 控制器及其值。两者可以分别启用，并且各自可以使用 CC、14 位 CC、RPN 或 NRPN 控制器。

Program Change。为 “开” 状态发送一条程序变更，带可选的库选择（MSB 和 LSB）。在锁定模式下，会为 “关” 状态发送第二条程序变更，带其自己的可选库选择。

Control Voltage。在一个 CV 输出上输出可配置的开电压和关电压，不发送任何 MIDI 消息。可用它直接从触控面板触发模块化包络、时钟或逻辑门。

Tap Tempo。每次按下都会敲击项目速度。Erae 2 会测量连续敲击之间的间隔，并相应地更新速度。无需任何 MIDI 设置。

LED 可视化。按钮显示静止状态和激活状态，样式包括 Dual Intensity、Dual Color 和 Center Fill。

参数	说明	范围	默认值
Type	按钮消息类型	Note / Control Change / Program Change / Control Voltage / Tap Tempo	Note
Latched	切换或瞬时	On / Off	Off
Note（Note）	MIDI 音符	`C-2` 到 `G8`	`C2`
On 控制器及值（CC）	“开” 状态的消息	控制器或禁用，值 `0` 到 `127`	
Off 控制器及值（CC）	“关” 状态的消息	控制器或禁用，值 `0` 到 `127`	
Bank / Program（PC）	开和关状态的库选择 MSB/LSB 与程序	各为 `0` 到 `127`，或禁用	
CV On/Off（Control Voltage）	带开电压和关电压的 CV 输出	输出或禁用	禁用
Channel	MIDI 通道	`1` 到 `16`	`1`

> **提示：**使用一个锁定的 Control Change 按钮来开关一个效果：按一次启用，再按一次禁用，LED 会确认当前状态。

> **提示：**在启用锁定的 Program Change 模式下，单个按钮可以在两个预设之间交替：第一次按下时为 “开” 程序，第二次按下时为 “关” 程序。

5.7 5.7 Key

琴键元素是一个单独的表现力琴垫：从键盘中取出的一个键，作为一个独立元素放置。它在被触摸时发送 Note On，在松开时发送 Note Off，并支持全套逐音符表现力。

表现力。每个触摸维度都被独立映射：

- **Pressure：**通道压力或复音触后，带其自己的曲线
- **Vibrato：**将水平方向的手指移动映射到 Pitch Bend
- **Pressure / X abs / Y abs / Motion** CC：以 CC 形式输出压力、绝对位置和移动速度
- **X rel / Y rel** CC：以 CC 形式输出相对于手指落点的移动
- **CV：**用于模块化的门信号、1 V/octave 音高和表现力输出

当您想要一个具备完整表现力、用于单个音符的大型演奏琴垫时，琴键元素是一个很好的选择：例如一个低音持续音琴垫、一个带压力的踩镲触发器，或一个宏调制面板。

参数	说明	范围	默认值
Note	要发送的 MIDI 音符	C-2 到 G8	C3
Pressure	触后类型和曲线	Channel / Poly	Channel，已
Vibrato	由水平移动产生的 Pitch Bend	风格、强度、回归速度	已启用
PB Range	以半音为单位的 Pitch Bend 范围	1 到 96	12
Pressure / X abs / Y abs / Motion CC	可选 CC	控制器或禁用	禁用
X rel / Y rel CC	可选的相对 CC	控制器或禁用	禁用
CV	CV 输出	输出或禁用	禁用
Channel	MIDI 通道	1 到 16	1

5.8 5.8 Ableton Launchpad

Ableton Launchpad 元素（在 Erae Lab 中称为 **LivePad**）将 Erae 2 转变为一个 Ableton Live 会话控制器：启动片段和场景，并控制音轨音量、静音、独奏和发送。LED 颜色和片段状态实时跟随 Ableton Live。它需要在 Ableton Live 中安装 Erae 2 MIDI Remote Script；参见附录 C：DAW 设置指南。

位置。该元素占据面板的 **Full Width**（整个宽度，全部 42 列）、**Left Half** 或 **Right Half**。拆分可以让您在一半上放置会话网格，在另一半上放置其他元素（推子、键盘、按钮）。

缩放。 **Zoom** 设置在 **Small** 单元（一次显示更多片段）和 **Large** 单元（更易点按）之间选择。在设备上，选中该元素时，**Plus** 和 **Minus** 可更改缩放。

模式。选中该元素时按 **Arp**，可在 Launcher、Mixer 和 Split 视图之间循环切换。

参数	说明	范围	默认值
Position	元素在面板上的位置	Full Width / Left Half / Right Half	由您放置的位置决定
Zoom	网格缩放	Small / Large	Large

> **提示：**将 Ableton Launchpad 元素放在面板左半部分，并在右半部分放置一组 Fader 1D 元素，即可在单个 Layout 中同时拥有会话片段控制和额外的推子。

5.9 5.9 API Zone {#59-api-zone}

API 区域元素将触控面板上的一个区域保留给通过 Erae 2 开发者 API 进行的直接编程控制。API 区域不会将触摸转换为 MIDI，而是将手指追踪数据（位置和压力）发送给已连接的主机应用程序。

这适用于高级集成：自定义 Max 补丁、TouchDesigner 设置、定制软件乐器，或任何想要直接访问多点触控数据的应用程序。

区域索引（Zone Index）。一个 Layout 中可以共存多个 API 区域，每个都由其 **Zone Index** 标识。主机应用程序读取区域索引以得知数据来自哪个区域。

Max Num Fingers 和 **Finger Data Rate** 会随 Layout 保存，但目前尚未生效：该区域会以设备自身的扫描速率报告每一次触摸。参见附录 D：开发者 API。

参数	说明	范围	默认值
Zone Index	供主机 API 使用的区域标识符	0 到 127	0
Max Num Fingers	已保存，尚未生效	0 到 10	10
Finger Data Rate	已保存，尚未生效	0 到 255	0

> **开发者 API。**用于手指流传输、LED 绘制（`SetPixel`、`DrawRectangle`、`DrawImage`）、区域边界查询和版本协商的完整 SysEx 线路协议，记录在附录 D：开发者 API 中。在将手指报告与绘制命令关联之前，请阅读 Y 轴一节：手指 Y 以底部为原点，绘制 Y 以顶部为原点。

5.10 5.10 Pedal

踏板元素为一个 Layout 配置两个踏板输入（**Pedal A** 或 **Pedal B**）之一。它在触控面板上没有区域；它定义如何解读所连接的踏板，以及它生成何种 MIDI 或 CV 消息。您可以在 Erae Lab 中，或在设备上的 **Settings > Pedal A** 和 **Pedal B** 下进行设置（参见设置）。

两个踏板输入与 CV 输出共用第 11 和第 12 号插孔。踏板类型可从 **Switch**、**Kick**、**Sustain (Binary)**、**Sustain (Continuous)** 和 **Expressive** 中选择。在设备上，两种延音类型显示为 **Sustain**，带 **Binary** 或 **Continuous** 模式。

输出路由。与所有生成 MIDI 的元素一样，踏板拥有自己的输出路由（USB Device、USB Host、MIDI A、MIDI B）和 MIDI 通道。

5.10.1 Switch

一个瞬时或锁定式脚踏开关。发送 Note、Control Change、Program Change 或 Tap Tempo 消息。在瞬时模式下，按下时发送消息，松开时发送一条互补消息；在锁定模式下，每次按下都会在开和关

之间切换。

消息类型：- **Note：**按下时发送 Note On，松开时发送 Note Off - **Control Change：**按下时发送 **On** 控制器及其值，松开时发送 **Off** 控制器及其值（各自可单独启用）- **Program Change：**为开和关状态发送一条程序变更（带可选的库选择）- **Tap Tempo：**每次按下都会敲击项目速度

可以在消息之外启用一个 **CV On/Off** 输出，用于同时进行模块化触发。

参数	说明	范围
Latched	切换或瞬时	On / Off
Message Type	Note / Control Change / Program Change / Tap Tempo	
Note（Note）	MIDI 音符	`C-2` 到 `G8`
On / Off 控制器及值（CC）	按下和松开时的 CC 消息	控制器或禁用，值 `0` 到 `127`
CV On/Off	带开电压和关电压的 CV 输出	输出或禁用

5.10.2 Kick

专为底鼓踏板设计。Kick 类型检测底鼓踏板敲击时的突然力度，测量其强度，并发送一条具有相应力度的 Note On。Note Off 会在固定时长后发送，或在踏板松开时发送。

参数	说明	范围	默认值
Note	要触发的鼓音符	`C-2` 到 `G8`	`C0`
Velocity Sensitivity %	冲击力度曲线强度	`0` 到 `100`	`100`
Impact Threshold %	触发所需的最小敲击力	`1` 到 `100`	`10`
Duration Mode	Note Off 的时机	Until Release / Fixed	Until Release
Fixed Duration (ms)	Fixed 模式下的音符长度	`1` 到 `5000`	`100`
CV Gate	门信号 CV 输出	输出或禁用	禁用
CV Velocity	力度 CV 输出	输出或禁用	禁用

5.10.3 Sustain (Binary)

一个标准的开/关延音踏板。按下时以开值发送一个 CC（默认 CC 64，延音），松开时发送关值。如果您需要反转踏板方向，请使用连续类型。

参数	说明	范围	默认值
Controller	CC 编号	0 到 127	64（Sustain）
On Value	按下时的 CC 值	0 到 127	127
Off Value	松开时的 CC 值	0 到 127	0
Latched	切换模式	On / Off	Off
CV On/Off	带开电压和关电压的 CV 输出	输出或禁用	禁用

5.10.4 Expressive

一个连续表情踏板，将踏板位置映射到 CC 值。踏板的完整行程映射到完整的 CC 范围（0 到 127）。**CC 11**（表情）是默认值。**Invert** 可为极性反转的踏板反转映射关系。

Pressure CV 输出提供与踏板位置成正比的电压。

参数	说明	范围	默认值
Controller	CC 编号	0 到 127	11（Expression）
Invert	反转踏板方向	On / Off	Off
CV	连续 CV 输出	输出或禁用	禁用

> **提示：**使用表情踏板并将 CC 11 路由到合成器的音量，即可实现自然的渐强控制，同时让您的双手在触控面板上保持自由。

5.10.5 Sustain Continuous

功能与 Expressive 类型相同，但默认采用 **CC 64**（延音）。它适用于能够报告连续位置而非开/关的半踏板（half-damper）延音踏板，在配合兼容的钢琴音色时可实现渐进式的延音深度。

参数	说明	范围	默认值
Controller	CC 编号	0 到 127	64（Sustain）
Invert	反转踏板方向	On / Off	Off
CV	连续 CV 输出	输出或禁用	禁用

> **提示：**对于电子鼓，将 Pedal A 设为 Kick，并在触控面板上使用 Drumpad 演奏军鼓、嗵鼓和镲。

> **提示：**踏板的路由独立于触控面板元素。您可以通过 MIDI A 向鼓机发送底鼓音符，同时通过 USB 将其他所有内容发送到您的 DAW。

CV 时钟输出（发送到 *CV* 输出的项目级时钟信号）将在第 *10* 章中介绍，因为它是一个项目范围的设置，而非逐元素的参数。

6 MIDI 配置

Erae 2 是一款功能完备的 MIDI 控制器，可生成标准 MIDI 1.0 消息，并在内部支持 MIDI 2.0。控制面上的每个元素（琴键、按钮、推子）都拥有各自独立的 MIDI 通道和输出目标，让您能够逐个元素地完全掌控消息发送到哪里以及如何被表达。

本章涵盖 USB MIDI 端口布局、通道分配、表现力参数映射、CC 路由、MPE、高分辨率 CC 和 NRPN，以及 MIDI 路由矩阵。

6.1 USB MIDI 端口

当 Erae 2 在常规 MIDI 1.0 模式下通过 USB 连接时，主机电脑会在单个 USB MIDI 设备中看到两条面向用户的 MIDI 线缆：

端口	名称	用途
Main	`Erae 2 MIDI`	标准 MIDI 输出：大多数 DAW 和乐器路由都使用此端口
MPE	`Erae 2 MIDI (MPE)`	MPE 输出：在 DAW 中选择此线缆以获得完整的表现力演奏；MPE 消息始终发往此

> **提示：**在 DAW 中设置 MPE 乐器轨道时，请将其指向 `Erae 2 MIDI (MPE)` 线缆。标准的非 MPE 轨道应从 `Erae 2 MIDI` 接收。

MIDI 2.0 是一种备用 USB 模式，通过 **Settings** 中的 **MIDI 2.0** 一行控制。更改它时会要求确认并重启设备。除非某个特定工作流程明确要求，否则请勿在 DAW 中手动路由 MIDI 2.0 线缆。

> **从固件 1.x 更新。** 2.x 固件将端口数量从三个减少至两个。专用的 Lab/API SysEx 线缆（原名 “MIDI (Lab)”，端口 2）已被移除：Erae Lab 现在使用其自己的 USB 连接，该连接不会出现在您的 DAW 中。MPE 数据原来位于端口 3（“MIDI (MPE)”），现在位于端口 2（“Erae 2 MIDI (MPE)”）。针对旧端口 2 的 API SysEx 宿主软件应重新映射到 `Erae 2 MIDI`（端口 1）。针对端口 3 用于 MPE 的 DAW 设置应重新映射到 `Erae 2 MIDI (MPE)`（端口 2）。完整的端口映射表请参阅 “快速入门” 章节。

6.1.1 2.x 更新后重新映射三端口设置

如果您在 2.x 固件之前使用过 Erae，DAW 项目、MIDI 工具或自定义 API 宿主可能仍然指向一个不再存在的 MIDI 端口。专用的第三 MIDI 端口（Erae Lab 和 SysEx 工具原来用于与设备通信的端口）在 2.x 中已被移除。Erae Lab 现在通过其自身的 USB 连接与设备通信，该连接不是 MIDI 端口，也不会出现在任何 DAW 或 MIDI 工具的端口列表中。

操作方法：

- **原来从旧主端口接收音符/CC 的 DAW 或乐器：**将其切换到 **`Erae 2 MIDI`**。这条线缆承载旧主端口所有的内容。
- **原来按编号或名称指向旧 Lab 端口的 SysEx 或 API 宿主：**将其切换到 **`Erae 2 MIDI`**（主端口）。API SysEx 现在通过主线缆传输：请参阅附录 D：开发者 API。
- **任何原来将第三端口用于 MPE 数据的工具：**将其切换到 **`Erae 2 MIDI (MPE)`**。

如果某个工具显示原第三端口位置为空或已变灰，该条目已过时。请移除或重新选择它，并将工具指向上述两个端口之一。在 2.x 中，您无需将旧的第三端口用于任何用途。

6.2 MIDI 通道与组

每个生成 MIDI 输出的元素都有一个 MIDI 通道：

参数	范围	默认值	说明
Channel	1 到 16	1	该元素输出所用的 MIDI 通道

Channel 决定该元素的 Note On/Off、CC 和 Program Change 消息由 16 个标准 MIDI 通道中的哪一个承载。控制面上的每个元素都可以使用不同的通道，因此单个 Layout 可以同时驱动 DAW 中的多个乐器或声部。在启用了 MPE 的键盘上，此设置变为 **Nb channel**（参见MPE 配置）。

元素还带有一个 MIDI 2.0 组号。Erae Lab 不显示该设置：元素使用第一个组，并且在 MIDI 1.0 模式下不会传输组号。

> **提示：**如果您希望多个元素演奏同一个乐器，请将它们放在同一通道上；如果要从一个 Layout 驱动多个声部，则使用不同的通道。

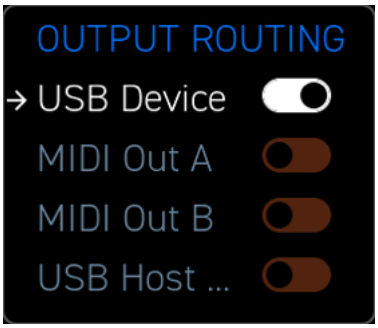
6.3 逐元素输出目标

每个元素还会选择由哪些物理端口传输它的消息。您可以启用四个端口的任意组合：

输出	端口
USB Device	USB Device（**Computer** 端口，连接到您的电脑）
USB Host	USB Host（**Synth** 端口，连接到外部 USB 设备）
MIDI A	MIDI 输出 A
MIDI B	MIDI 输出 B（在低功耗模式下被禁用）

新元素仅发送到 **USB Device**。启用额外的目标不会改变通道：相同的消息会被发送到每个选中的端口。

您可以在 Erae Lab 中设置输出，也可以在设备上设置：在 **Mapping** 画面中打开该元素，选择 **Routing** 进入 **OUTPUT ROUTING** 画面，其中有 **USB Device Out**、**MIDI Out A**、**MIDI Out B** 和 **USB Host Out** 开关。



元素路由

> **提示：**将贝斯键盘发送到 MIDI A 以驱动硬件合成器，并将其他所有内容保留在 USB Device 上供您的 DAW 使用。

6.4 琴键元素：表现力参数

琴键和键盘是主要的表现力触摸区域。琴键在触碰时发送 Note On，在按住期间跟踪压力和位置，并在松开时发送 Note Off。音符力度来自您敲击表面的方式，并由设置中全设备通用的 **Velocity Curve** 塑形（参见触控表面与触控）。在 Erae Lab 中，表现力参数分组在 **Pressure**、**Glissando**（仅限键盘）和 **Vibrato** 之下。

6.4.1 压力

压力是 note-on 之后施加的持续作用力。Erae 2 可以将压力作为复音触后（逐音符）或通道压力（逐通道）发送：

参数	范围	默认值	说明
Enable	on / off	on	发送压力消息
Type	`Channel` / `Poly`	`Channel`	压力输出所用的消息类型
Tracking	`None` / `LastPlayed` / `Highest` / `Lowest`	`None`	用于通道压力：按住多个手指时由哪个手
Filter	`Exponential` / `Linear`	`Exponential`	响应曲线
Min/Max	`0` 到 `127`	`0` / `127`	输出范围
Intensity	`0` 到 `255`	`127`	灵敏度
Smoothing	`0` 到 `255`	`0`	应用于压力读数的低通平滑

> **提示：**对于 MIDI 1.0 MPE 乐器，通道压力通常是最稳妥的选择，因为每个手指本就拥有各自的成员通道。对于只有单一触后输入的传统非 MPE 合成器，请使用 `Channel` 并将 **Tracking** 设为 `Highest`。

> **警告：**在非 MPE 模式下，同时按住的琴键共享一个 MIDI 通道。使用 `Poly` 时，每个音符的压力消息都带有其音符编号，但许多乐器无论如何都会把所有复音压力映射到单一数值。使用 `Channel` 时，每个通道只发送一个压力值，因此多个手指会争夺这一条数据流。请启用 MPE 以获得真正的逐音符压力独立性。

6.4.2 揉弦（Pitch Bend）

Vibrato 模块控制琴键上水平方向的手指移动如何映射为 pitch bend。为了准确跟踪半音，pitch bend 范围（**PB Range**）必须与您的合成器中的设置相匹配。

参数	范围	默认值	说明
Enable	on / off	on	根据手指移动发送 pitch bend
Style	`Linear` / `Quadratic` / `FarQuadratic`	`Linear`	从位置到 pitch bend 的映射曲线
Strength	滑块		给定移动量对应的 pitch bend 量
Return Speed	`None` 到 `Max`		bend 回归中心的速度
PB Range	1 到 96 个半音	12（MPE 时为 48）	pitch bend 消息的半音范围

开启 MPE 会将 **PB Range** 设为 48；关闭 MPE 会将其重置为 12。

6.4.3 滑音（Glissando）

Glissando 模块（仅限键盘）控制手指从一个琴键滑到另一个琴键时的音高：

参数	范围	默认值	说明
Enable	on / off	on	启用滑音
Tune Loc	`Pad` / `Finger`	`Pad`	零 pitch bend 的参考点：琴键中心，或手指落下的位置
In Tune Width	`0` 到 `100` %	`50` %	准音区域的宽度，以琴键宽度的百分比表示
Retrigger	on / off	off	滑动时越过到新琴键时发送新的 Note On
Vertical Gliss.	on / off	on	让行与行之间的垂直移动参与滑音
Smoothing	`Low` 到 `High`		滑音的平滑度

> **提示：**将 **In Tune Width** 设为 `100 %`，即可获得干净的半音，琴键内不会出现微分音漂移。

6.4.4 音色明亮度（CC 74）

键盘有一个 **Sound Brightness (CC 74)** 部分，可根据所选的 **Gesture** 发送 CC 74：`None`、`Pressure`、`X Absolute`、`Y Absolute`、`X Relative`、`Y Relative` 或 `Motion Speed`。它还具有 **Initial value**、**Min/Max**、**Intensity**、**Smoothing**、**Tracking** 和 **Filter**。在新键盘上它是关闭的（`None`）。

6.5 CC 映射

琴键、键盘和推子可以根据触摸位置和压力发送连续控制器（Continuous Controller）消息。可用的 CC 槽位包括：

CC 槽位	轴	元素类型
Pressure	接触力（Z）	琴键、键盘、Fader 1D、Fader 2D
X abs	在琴键或元素内的水平位置	琴键、键盘、Fader 2D
Y abs	在琴键或元素内的垂直位置	琴键、键盘、Fader 1D、Fader 2D
X rel	相对于手指落点的水平移动	琴键、键盘
Y rel	相对于手指落点的垂直移动	琴键、键盘
Motion	手指移动速度	琴键、键盘、Fader 1D、Fader 2D

每个槽位都有一个 **Enable** 开关、一个控制器 **Type**（见下文）、一个控制器编号、**Min/Max** 输出范围（`0` 到 `127`）、**Intensity**、**Smoothing**、**Tracking** 和 **Filter**。被禁用的槽位不发送任何数据。绝对槽位跟随手指在琴键或元素上的位置。相对槽位在手指落下时从 **Initial value**（默认 `64`）开始，并随其位移而变化。Motion 跟随手指的移动速度，经过平滑并从 `0` 归一化到 `100 cm/s`。

Control Change 类型的 **Button** 元素发送一条 **On** 消息和一条 **Off** 消息，各自带有自己的控制器和值，并且都可以被禁用。

> **提示：**在一个较高的琴键上将 **Y abs** 映射到滤波器截止频率，即可在单个琴垫内创建一条丝带（ribbon）风格的表现力条带。

6.6 高分辨率 CC 和 NRPN

每个控制器槽位都有一个 **Type**：

- **CC：**在控制器 `0` 到 127 上发送的标准 7 位 Control Change。
- **CC 14-bit：**在控制器 `0` 到 31 上发送的高分辨率 Control Change。Erae 2 按照 MIDI 规范，在所选控制器上发送 MSB，在控制器编号加 32 上发送 LSB。
- **RPN** 和 **NRPN：**Registered 与 Non-Registered Parameter Number，通过 **Bank** 和 **Index**（各为 `0` 到 `127`）寻址。

接收软件必须支持 14 位 CC、RPN 或 NRPN，才能利用额外的分辨率。

> **提示：**要发送 14 位数值，请将槽位的 **Type** 设为 `CC 14-bit`，并选择 `0` 到 `31` 之间的控制器。

6.7 MIDI 1.0 与 MIDI 2.0：您实际读取到的数值

Erae 2 以高内部分辨率捕获并处理触控，但最终到达您软件中的内容取决于连接实际使用的协议，而这是由您的电脑与设备之间决定的，并非设备单方面决定。

- **MIDI 1.0（默认）。**两条 USB MIDI 线缆（`Erae 2 MIDI` 和 `Erae 2 MIDI (MPE)`）承载标准 MIDI 1.0。MIDI 1.0 接收方看到的 Control Change 值为 7 位（`0` 到 `127`）。要在 MIDI 1.0 上获得超过 7 位的精度，请使用上文所述的 `CC 14-bit`、RPN 或 NRPN 控制器类型：接收方必须能理解它们，才能重建高分辨率值。
- **MIDI 2.0（备用 USB 模式）。**在 **Settings** 中开启 **MIDI 2.0**（设备会重启）后，设备进入 USB MIDI 2.0 模式，暴露通用 MIDI 数据包（Universal MIDI Packet，UMP）流，可以原生承载完整的高分辨率值。

> **读取高分辨率值需要双方达成一致。**在设备上启用 MIDI 2.0 仅是其中一半。您的宿主或操作系统必须与设备协商 MIDI 2.0 协议，且您的软件必须从 MIDI 2.0 端点/流中读取数据。如果宿主仍处于 MIDI 1.0，或者您的软件通过 MIDI 1.0 端口读取设备，即便已开启 MIDI 2.0，您仍然只会看到 7 位 CC 值。如果您预期获得高分辨率数据但只看到 `0` 到 `127`，请确认：（1）MIDI 2.0 已开启且设备已重启，（2）您的操作系统/宿主将该设备报告为 MIDI 2.0 端点，（3）轨道或工具正在读取 MIDI 2.0 流，而不是同一设备的旧版 MIDI 1.0 视图。

6.8 MPE 配置

MPE（MIDI Polyphonic Expression，MIDI 复音表现力）允许每个手指在各自的 MIDI 通道上拥有独立的 pitch bend、压力和滑移，从而在复音音色中实现逐音符表现力。

Erae 2 实现了由 MIDI MPE 规范定义的两个 MPE 区域。区域通过 **MPE Master** 设置**逐键盘元素**地设定：

MPE Master	MPE 区域	成员通道
Channel 1（默认）	Lower Zone	通道 2 及以上（动态分配）
Channel 16	Upper Zone	通道 15 及以下（动态分配）

启用 MPE 后，键盘的 **Channel** 设置变为 **Nb channel**：即该键盘使用的成员通道数量（1 到 15）。开启 MPE 还会将 **PB Range** 设为 48。

MPE 消息始终在 `Erae 2 MIDI (MPE)` 端口上发送。请参阅本章开头的 USB MIDI 端口部分。

每个键盘元素与 MPE 相关的设置：

设置	推荐的 MPE 值
MPE	`on`
MPE Master	`Channel 1`（Lower Zone）或 `Channel 16`（Upper Zone）
PB Range	48 个半音
压力 **Type**	MIDI 1.0 MPE 使用 `Channel`；使用 MIDI 2.0 逐音符压力时使用 `Poly`
滑移	CC 74，来自 **Y rel** CC 槽位或 **Sound Brightness (CC 74)** 部分

> **提示：**大多数 MPE 合成器要求控制器和乐器上的 pitch bend 范围设置完全一致。请在元素编辑器中将 **PB Range** 设为 48，并使用乐器自身的 MPE 设置页面进行匹配。

6.8.1 让滑移和压力进入您的 DAW

滑移和压力是逐音符表现力，因此只有当整条链路均处于 MPE 模式时，它们才能完整地传递给您的乐器。如果音符能正常演奏，但滑移和压力没有任何效果，原因几乎必然是链路中存在非 MPE 的环节，而不是设备本身的设置问题。请逐项检查：

1. **从 MPE 线缆读取数据。**将 DAW 轨道的 MIDI 输入指向 **`Erae 2 MIDI (MPE)`**，而非主端口。逐音符表现力只在 MPE 线缆上发送。
2. **传递全部 16 个通道。**将轨道的输入通道设为 **All Channels**（或 **Any**）。MPE 将每个手指分散到各自的成员通道上；如果 DAW 将传入的 MIDI 折叠到单个通道，每个音符的表现力都会叠加到一起。

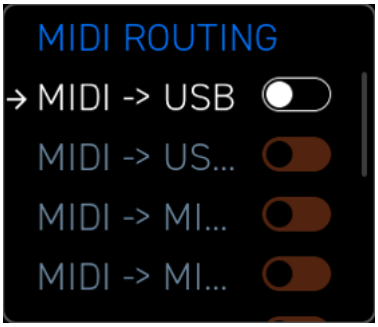
3. **在轨道和乐器中启用 MPE。**将轨道切换到 MPE 模式（适用于 DAW 有明确开关的情况），并在乐器本身中启用 MPE / Per-Note / Poly Expression。
4. **将滑移轴映射到乐器的滑移控制。**滑移通常为 CC 74（Timbre）。确认键盘为垂直滑移发送 CC 74（见上方 MPE 设置表），且乐器以 CC 74 作为其滑移/音色输入。
5. **两侧 pitch bend 范围匹配**（默认 48 个半音），以确保滑动时音高跟踪正确。

有关 Ableton Live、Logic、Cubase、Bitwig、Reaper 和 FL Studio 的逐 DAW 分步操作说明，请参阅附录 C：DAW 设置指南。

> **注意事项：**大多数 DAW 在将 MIDI 传递给非 MPE 轨道之前，会将传入的 MIDI 折叠到单个通道，从而悄悄丢弃滑移和压力数据。如果您从主端口而非 `Erae 2 MIDI (MPE)` 接收，或者将轨道保持在标准（非 MPE）模式，您会听到音符，但不会有任何表现力。

6.9 MIDI 路由矩阵

路由矩阵控制哪些物理端口将 MIDI **输入**消息转发到其他端口，这与元素的输出目标相互独立。这使得 Erae 2 能够充当 MIDI 合并与直通（thru）盒。



MIDI 路由

从 **Settings > Routing** 打开它；该画面标题为 **MIDI ROUTING**。每一行都是一个开关：

开关	源 -> 目标	作用
MIDI -> USB Host	MIDI In -> USB Host Out	将硬件 MIDI In 转发到已连接的 USB 设备
MIDI -> USB Device	MIDI In -> USB Device Out	将硬件 MIDI In 转发到主机电脑
MIDI -> MIDI A	MIDI In -> MIDI A Out	MIDI Thru 到 MIDI A
MIDI -> MIDI B	MIDI In -> MIDI B Out	MIDI Thru 到 MIDI B

开关	源 -> 目标	作用
USB Device -> USB Host	USB Device In -> USB Host Out	将主机电脑的 MIDI 路由到已连接的 USB 设备
USB Device -> MIDI A	USB Device In -> MIDI A Out	将主机电脑的 MIDI 路由到 MIDI A 上的硬件设备
USB Device -> MIDI B	USB Device In -> MIDI B Out	将主机电脑的 MIDI 路由到 MIDI B 上的硬件设备
USB Host -> USB Device	USB Host In -> USB Device Out	将已连接 USB 设备的 MIDI 路由到主机电脑
USB Host -> MIDI A	USB Host In -> MIDI A Out	将已连接的 USB 设备路由到 MIDI A 上的硬件设备
USB Host -> MIDI B	USB Host In -> MIDI B Out	将已连接的 USB 设备路由到 MIDI B 上的硬件设备

将某个开关打开（**on**）即可启用该路由路径。所有路由开关都是相互独立的：多个源可以馈入同一个目标。

> **提示：**要将 Erae 2 用作简单的 2 端口 MIDI 接口，请启用 **MIDI -> USB Device** 和 **USB Device -> MIDI A**。这样，您的 DAW 就能通过 Erae 2 向硬件合成器发送数据并从中接收数据，而无需任何额外的接口。

路由设置随**项目**保存，因此每个项目都可以携带各自的合并配置。

6.10 相关主题

- 元素：逐元素类型的配置细节，包括按钮 CC 和 Program Change 的设置
- LCD 界面：在 LCD 显示屏上导航菜单
- 设置：项目级设置，包括 MIDI 时钟和同步
- MIDI 实现表：完整的消息参考表

7 音阶与调音

音阶设置是**针对每个键盘元素**的，而非全局设置（除非您选择加入全局音阶，见下文）。Layout 中的每个键盘元素都会独立存储自己的音阶和基准音（根音和八度）。音阶界面编辑的是**当前所选元素**的设置。

音阶界面控制所选键盘元素如何将触摸坐标解释为音高。

音阶选择器

按下前面板上的 **Scale** 按键可打开所选键盘的音阶界面；再次按下即可关闭。若要选择另一个键盘，请按住 **Scale**：除所选元素外的所有内容都会变暗，触摸某个键盘即可选中它。松开 **Scale** 后，音阶界面会为该键盘打开。

标题行显示 **SCALE**。其下是音阶列表、**ROOT** 和 **OCTAVE** 数值，在 Keygrid 上还有 **GLOBAL** 和 **OFFSCALE** 开关。转动编码器，高亮会按此顺序移动（音阶、根音、八度、**GLOBAL**、**OFFSCALE**）。按下数值即可编辑，转动来更改，再按一次确认；按下开关即可切换。

7.1 内置音阶库

音阶列表列出了出厂音阶。出厂音阶是固定的，无法编辑或删除。

音阶库包含 15 个出厂音阶：

1. `Chromatic`：全部 12 个半音
2. `Major`：标准的自然大调音阶
3. `Minor`：自然小调（爱奥利亚调式）
4. `Melodic Minor`：升高的六度和七度
5. `Harmonic Minor`：升高的七级音

6. `Arabic`
7. `Dorian`
8. `Phrygian`
9. `Gypsy`
10. `Mixolydian`
11. `Romanian`
12. `Gypsy Minor`
13. `Japanese`
14. `Spanish`
15. `Blues`

Keygrid（Iso Keyboard）列出除 `Chromatic` 之外的 14 个音阶：若要在 Keygrid 上演奏每个半音，请改为开启 **Show Offscale**。Chroma Keyboard 列出全部 15 个音阶。

> **提示：**在 Chroma Keyboard 上，`Chromatic` 表示不进行音阶着色：每个半音看起来都像普通的钢琴键。

7.2 根音

ROOT 数值设置所选音阶的调性中心。共有十二个可选值：`C`、`C#`、`D`、`D#`、`E`、`F`、`F#`、`G`、`G#`、`A`、`A#`、`B`。

根音决定了哪个音级被视为第一级（主音），它也是左下角琴垫的音符。改变根音会整体移调整个音阶，而不会改变其音程结构。Drumpad 和 Chroma Keyboard 都显示 ROOT 和 OCTAVE。

> **提示：**在 Keygrid 上，落在根音上的琴垫使用第一音级的样式，使调性中心在触摸表面上一目了然。

7.3 八度

OCTAVE 数值设置键盘基准音的八度，范围为 -2 到 8。根音为 C 时，八度 3 会将中央 C（MIDI 音符 60）放在左下角琴垫上；八度 -2 从 MIDI 音符 0 开始。

在音阶界面之外，**Plus** 和 **Minus** 按键也可以将所选键盘向上或向下移动一个八度。

> **提示：**演奏时，使用八度按键可将键盘调整到合成器所期望的音域。

7.4 显示非音阶音（Show Offscale）

OFFSCALE 开关（音阶界面底行，仅限 Keygrid）即 Show Offscale 设置。它控制非音阶半音是否出现在 Keygrid 上。

- **开：**全部 12 个半音均可演奏。音阶内的音符采用音级颜色样式；音阶外的音符采用非音阶样式。
- **关：**仅显示音阶音级。触摸表面会重新映射，使每个物理单元格都演奏一个属于所选音阶的音符；无法访问任何非音阶半音。

> **提示：**关闭 Show Offscale 后便无法弹出“错音”，这非常适合现场演出、即兴演奏以及初学者。

7.5 音阶如何影响各类键盘

键盘类型	音阶支持	是否有 Show Offscale 开关?	全局音阶?
Iso Keyboard（Keygrid）	完整支持：过滤并高亮音级，设置行间音程	有	是
Chroma Keyboard	仅着色：全部 12 个半音保持可演奏	无	否
Drumpad	无：始终为半音；隐藏音阶列表	无	是（保持半

Iso Keyboard 拥有最丰富的音阶交互：音阶还控制行间音程。当 Show Offscale 关闭时，每一行的步进以音级而非半音计算，因此同构几何结构会在音阶范围内得以保留。

Chroma Keyboard 仅将音阶用于着色。钢琴布局是固定的，并始终使用十二平均律。

Drumpad 忽略音阶；其音阶界面只显示 ROOT 和 OCTAVE 列表以及 **GLOBAL** 开关。每个琴垫从基准音开始，依次映射到连续的 MIDI 音符。

7.6 全局音阶

通常每个键盘元素都带有自己的音阶。当您希望由一个音阶统领整个项目时，请让键盘加入**全局音阶**：在音阶界面上开启 **GLOBAL** 开关，或在 Erae Lab 中开启 **Use global scale**。更改任何已加入的键盘（位于任何 Layout）的音阶，都会同时更改所有已加入键盘的音阶。

- 只共享音阶。根音和八度仍由每个键盘各自保留。
- 关闭该开关会使键盘脱离：它保留当前音阶，不再跟随之后的更改。
- Chroma Keyboard 无法加入。Drumpad 可以加入，但它始终保持半音排列。

7.7 自定义音阶

除出厂音阶外，Keygrid 还可以使用**自定义音阶**：每个八度可包含任意数量的音符，可按比率自由调律，或按八度的等分调律，包括微分音调律。自定义音阶在 Erae Lab 中或设备上的 **Settings > Custom Scales** 下创建和编辑，并由所有项目共享。

- 自定义音阶仅在启用了 **MPE** 的 Keygrid 上提供。每个微分音音符都以一个基准音加上其自己的 pitch bend 在 MPE 成员通道上演奏，因此在和弦中调律也能保持正确。
- 在这样的键盘上，音阶列表依次列出出厂音阶、您的自定义音阶，以及 **NEW SCALE**，后者会创建一个音阶并打开设备上的编辑器。
- 在使用自定义音阶的键盘上关闭 MPE，会使其恢复为出厂音阶。

有关创建、导入和编辑自定义音阶（包括设备上的编辑器）的完整指南，请参阅 Erae Lab 用户手册第 18 章。

7.8 各元素独立的音阶设置

每个键盘元素都会独立存储自己的音阶分配。音阶界面始终编辑**当前所选元素**的音阶。若要为不同元素配置不同的音阶，请按住 **Scale**，触摸下一个键盘，然后松开。

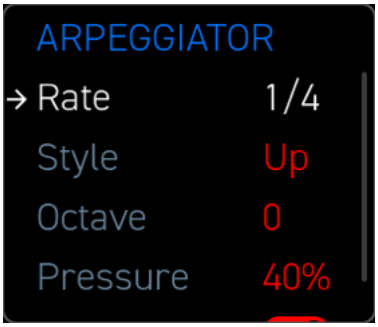
> **提示：**您可以创建这样一个 Layout：一个元素演奏 Japanese 音阶，而另一个元素演奏 Blues 或 Major，这对分层乐器或分区触摸表面设置非常有用。

8 琶音器与 Looper

Erae 2 提供两种实时演奏工具：**琶音器**，可根据按住的音符生成旋律性的演奏型；以及 **Looper**，可将您的演奏录制下来，并作为与项目时钟同步的循环演奏型反复播放。

8.1 琶音器

琶音器会将按住的音符转化为有节奏的序列。每个键盘元素（Keyboard、Keygrid 或 Drumpad）都有自己的琶音器。



琶音器

选中一个键盘，然后按下前面板上的 **Arp** 按键。这会开启（或关闭）该键盘的琶音器，并打开 **ARPEGGIATOR** 画面。若要打开该画面而不切换琶音器的开关状态，请按住 **Settings** 并按下 **Arp**。

8.1.1 启用琶音器

每按一次 **Arp**，都会开启或关闭所选键盘的琶音器；开启时 **Arp** 按键亮绿色。琶音器关闭时，音符正常发声。开启后，按住的音符不会直接发声，而是由琶音器逐个演奏。开关状态会随 Layout 保存。

琶音器也作用于步进音序器在该键盘上演奏的音符：琶音器开启时，在音序器中绘制的和弦会以琶音方式发声。

8.1.2 速率（Rate）

Rate 设定琶音音符相互跟随的速度。可选值如下：

1/32、1/16、1/8、1/4、1/2、1/1、Pressure

速率是以项目速度下一拍（一个四分音符）为单位的分数：1/1 每拍演奏一个音符，1/4 每拍演奏四个音符（十六分音符），1/32 每拍演奏三十二个音符。

在 Pressure 模式下，速率跟随您按压的力度：轻压时大约每拍演奏一个音符，按得越用力琶音越快，最快可达 1/32。

> **提示：** 1/4 产生十六分音符琶音，这是最常用的起点。如果想要由手指控制速度、富于表现力的琶音，请使用 Pressure 模式。

8.1.3 演奏方式（Style）

Style 决定按住的音符按何种顺序演奏：

- Up：音符从最低音到最高音依次演奏，然后重复
- Down：音符从最高音到最低音依次演奏，然后重复
- UpDown：音符先上行再下行，然后重复；位于顶端和底端的转折音**不会**演奏两次
- UpAndDown：音符先上行再下行，然后重复；位于顶端和底端的转折音**会**演奏两次（上行时一次，下行时一次）
- Random：每个步进从按住的音符集合中随机选取一个音符

> **提示：** 对一个四音和弦使用 UpDown 会产生一个六步的演奏型，当演奏型长度无法整除小节时，会形成有趣的相位错位。如果您希望转折音通过其重复出现而得到强调，请使用 UpAndDown。

8.1.4 八度范围（Octave Range）

Octave 设定琶音器在演奏音符之上额外跨越多少个八度。范围：0 到 8。

设为 0 时，琶音停留在按住音符所在的音区内。设为 1 时，琶音器先演奏按住的音符，然后将它们升高一个八度重复，再循环回来。

> **提示：** 将 Down 方式与 Octave 3 搭配，可产生跨越数个八度、层层下落的琶音，在 Pad 音色和拨弦音色上效果尤为戏剧化。当您想要紧凑、单一音区的琶音时，请将 Octave 设为 0。

8.1.5 压力转力度（Pressure to Velocity）

Pressure 设定当前手指压力在多大程度上塑造每个琶音音符的力度。范围：0% 到 100%，步长为 10%。默认值：40%。

设为 0% 时，琶音音符使用您初次按下时的力度。设为 100% 时，每个音符的力度跟随您在那一刻按压的力度。介于两者之间的取值会将二者混合。

> **提示：** 将 Pressure 设为 60% 到 80%，可在表现力演奏中让手指力度的细微变化形成自然的重音，同时保持整体力度的稳定。

8.1.6 量化（Quantize）

Quantize 开关决定琶音何时开始。它默认开启。

- **On**：琶音从项目时钟网格的下一个步进开始，因此它与速度及其他声部保持对齐。在下一个网格步进之前松开的音符不会发声。
- **Off**：触摸时立即演奏第一个音符，琶音从您的触摸开始运行，而不是从网格开始。

在两种情况下，速率都跟随项目速度。

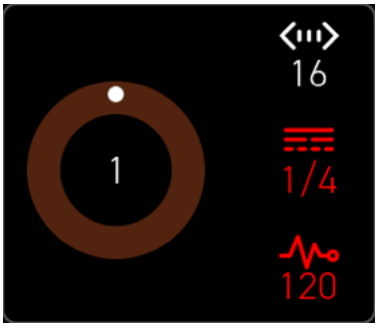
> **提示**：配合 DAW 或 Looper 演奏时，请保持 Quantize 开启。当您希望非常短的触摸或较短的音序器步也能发声时，请将其关闭。

8.2 Looper

Looper 会录制您在 Layout 上的触控手势（包括压力和滑动），并与项目时钟同步循环回放。由于它录制的是手势而非 MIDI，回放的循环会再次经过 Layout 中的元素：更改某个元素的设置，循环也会随之变化。

每个 Layout 都有自己的循环。Home 主屏幕显示您当前所在 Layout 的循环。

8.2.1 主屏幕 Looper 显示



LCD 主屏幕

Home 主屏幕显示：

- **循环圆环**： Layout 编号，以及一段显示循环内播放位置的圆弧。当该 Layout 的循环被静音时，圆弧会变为另一种颜色。
- **Length**：以步为单位的循环长度（一步为一个十六分音符）。
- **Quantize**：应用于录制手势的量化网格。
- **Tempo**：以 BPM 表示的项目速度。

8.2.2 录制与回放

Play/Rec 和 **Stop** 按键控制走带：

- 内部时钟停止时，**Play/Rec** 会启动时钟。时钟运行期间，**Play/Rec** 按键亮绿色。
- 时钟运行期间（或项目跟随外部时钟时），**Play/Rec** 会预备或取消预备录制。预备录制时，该按键亮红色，您在当前 Layout 上演奏的所有内容都会被录入其循环。
- **Stop** 停止时钟。使用内部时钟时，停止也会取消录制预备。已录制的循环会被保留。

录制始终为叠录：预备录制时，新的手势会叠加到循环已有的内容之上。每经过一遍循环就形成一层。

撤销一层。在 Home 主屏幕上，按 **Return** 可移除最近录制的一层。

清除循环。按住一个或多个 Layout 按键（**N1** 至 **N8**）并按 **Return**，即可清除这些 Layout 的循环。如需同时清除音序，请按住 **Return** 约一秒：屏幕会列出这些 Layout 并请求确认，选择 **Yes** 即可清除它们的循环以及每个 Layout 的全部八个 sequencer 演奏型。

静音循环。按住 **Stop** 并按某个 Layout 按键，即可静音该 Layout 的循环；按住 **Play/Rec** 并按某个 Layout 按键，即可取消静音。被静音 Layout 的按键以红色显示。

> **提示：**由于录制始终为叠录，较短的循环会很快被填满。开始录制之前，请先设置 Length。

将循环并轨到步进音序器。已录制的循环可以通过 Sequencer Settings 中的 **Bounce Loop** 一行转换为可编辑的音序器演奏型。参见第 9 章。

8.2.3 Looper 控制

Looper 控制位于 **Home** 主屏幕上。转动编码器可高亮某个控件，按下编码器进行编辑，转动以更改数值，再次按下即可完成。

Length：以步为单位的循环长度，范围为 1 到 256（默认 16）。编辑 Length 时，**Plus** 会将其加倍，**Minus** 会将其减半。

Quantize：录制手势吸附到的网格。参见下文的量化网格。

Tempo：以 BPM 表示的项目速度。编辑 Tempo 时，**Plus** 会将其加倍，**Minus** 会将其减半。仅当时钟源为内部时，才能在此更改 Tempo。

> **提示：**录制时将 Quantize 设为 `Off` 可保留自然、富于表现力的时值；设为 `1/4` 则可在录制时将节奏声部收紧到十六分音符。

8.2.4 时钟同步

Looper 始终锁定到项目时钟。当项目时钟源为 `INT`（内部）时，Looper 以项目速度运行。当时钟源为 `USB-dev`、`MIDI` 或 `USB-host` 时，Looper 会跟随传入的 MIDI 时钟，从而与 DAW 或硬件时钟源实现紧密同步。

时钟源配置请参阅第 15 章：设置。

8.2.5 演奏型长度

演奏型长度通过 Home 主屏幕上的 **Length** 控制设定，以十六分音符为步进，最多 256 步（16 小节 4/4 拍）。

8.2.6 量化网格

Quantize 网格会在录制手势时，将每个手势的起点吸附到最近的网格位置。可选值：

`Off`、`1/16`、`1/12`、`1/8`、`1/6`、`1/4`、`1/3`、`3/8`、`1/2`、`3/4`、`1`、`5/4`、`2`、`3`、`4`

数值是以一拍（一个四分音符）为单位的分数：`1/4` 吸附到十六分音符，`1` 吸附到四分音符，`4` 吸附到完整的 4/4 小节。默认值为 `1/4`。

量化在录制时应用。更改网格只影响之后录制的手势，不影响循环中已有的手势。

> **提示：**网格只移动每个手势的起点。手势内部的压力、滑动和其他表现力保持其自然的时值。

8.2.7 击拍设速（Tap Tempo）

Tap Tempo 通过随拍敲击来设定项目速度。它在时钟源为内部时有效。要使用 Tap Tempo：

- 按住 **Sequencer**，每拍按一下编码器。
- 将一个踏板（Switch 类型）配置为 **Tap Tempo** 消息类型。每次按下踏板都会敲击一次速度。
- 使用触摸表面上的一个 **Button** 元素，并将其设为 **Tap Tempo**（参见按钮）。

Erae 2 会测量连续敲击之间的间隔并更新项目速度。Looper 和琶音器会立即跟随新的速度。

9 第 9 章：步进音序器（Step Sequencer）

步进音序器是内置在表面中的基于演奏型的 MIDI 音序器。Looper 捕捉的是一段自由的演奏，而音序器让您以两种方式构建演奏型：直接在 LED 网格上逐步绘制并编辑，或者在时钟运行时将演奏实时录制进演奏型。您可以为每一步独立控制音高、力度、闸门长度以及连续表现力。它的状态会随您的项目按 Layout 保存。

9.1 进入与退出音序器

按下 **Sequencer** 按钮即可进入音序器模式。触控表面会切换到三区音序器视图（见下文），LCD 会显示 Sequencer 画面。

若要在进入时选择要音序化的元素，请按住 **Sequencer**：除所选元素外的所有内容都会变暗，此时触摸某个元素会选中它而不是演奏它。松开 **Sequencer** 后，音序器模式会在该元素上打开。

该按钮会在三种情形之间循环：

- 音序器模式关闭时：按下 **Sequencer** 会进入该模式。
- 音序器模式开启，且 LCD 正显示 Sequencer 或 Sequencer Settings 画面时：按下 **Sequencer** 会退出音序器模式，并将 LCD 返回 Home。
- 音序器模式开启，但 LCD 正显示其他画面（Home、Scale 等）时：按下 **Sequencer** 会重新调出 Sequencer LCD 画面，而不会离开该模式。

如果当前活动 Layout 中没有任何可供音序器录制的内容（例如一个只包含 API 区域的 Layout），该按钮不会有任何反应：音序器需要至少一个可被音序化的元素才能打开。

离开音序器模式会提交您尚在绘制中的音符，因此在退出的瞬间完成的编辑绝不会丢失。

9.2 可以音序化的内容

音序器并不只录制键盘。它可以捕捉四种元素类型，其显示的泳道会随所聚焦元素的不同而变化：

元素	每一步记录的内容	提供的泳道
Keyboard（音阶、半音、鼓垫，以及专用的 Sequencer Keyboard）	一个音符：音高、力度、闸门长度和表现力	Note、Pressure、Slide、Pitch，外加每个已启用按键手势各一条泳道（pressure CC、X abs、Y abs、X rel、Y rel）
Fader 2D（一个 XY 面板）	X、Y 和 Z（压力）数值	X、Y、Z
Fader 1D（单个推子）	单一数值自动化	Value
Button	每一步的开或关状态（带持续时长的按下）	State（显示为 `steps`）

音序器会聚焦您最后触摸过的那个 Layout 元素。如果进入时没有合适的聚焦对象，系统会自动选中 Layout 中第一个可被音序化的元素。

要更改当前正在音序化的元素，请单独按住该 Layout 的按钮（**N1** 到 **N8**），不要同时按住 Sequencer。表面会临时显示该槽位的常规完整 Layout，方便您点按想要的元素。松开按钮后，三区音序器视图会返回，并绑定到您的选择上。请注意 “按住” 和 “点按” 的区别：快速点按并松开当前 Layout 自身的按钮，会完全离开音序器模式。按住 Layout 按钮时按 **Stop**、**Play/Rec** 或 **Return**，会照常静音、取消静音或清除该 Layout 的循环，sequencer 仍停留在你原来所在的 Layout。

在音序器模式中按下另一个 Layout 的按钮，会切换到该 Layout 并保持在音序器模式中，重新打开您上次在该 Layout 上音序化的元素。

9.3 表面：三个区域

音序器处于活动状态时，触控表面会从左到右分成三个固定区域，每个区域都占满整个高度：

- **Live**，最左侧 7 列：所聚焦元素的实时可演奏镜像。详见下文的 Live 区域。
- **Grid**，中间 32 列：用于绘制和编辑的步网格。
- **Launcher**，最右侧 3 列：八个堆叠的垫块，每个演奏型槽位一个，用于切换演奏型。最上方的垫块对应演奏型 1，最下方对应演奏型 8。

启用 **Swap Sides**（见 Sequencer Settings）后，Launcher 会移到左边缘，Live 移到右边，网格仍居中并保持 32 列宽。

每根手指所在的区域在按下的那一刻就已固定。一个从网格开始的手势即使手指滑向 Launcher 也会继续按编辑处理，因此各区域不会在手势中途相互越界触发。

9.3.1 Live 区域

Live 区域是所聚焦元素的可演奏镜像，并且会进行录制。是否录制取决于录制是否已预备（arm），这与 Looper 所使用的 **Play/Rec** 预备机制相同：

- **录制已预备**（Play/Rec 开）：在 Live 区域演奏会直接从当前播放头（playhead）位置录入活动演奏型。键盘会连同其压力曲线、松开力度（release velocity）和真实持续时长一起录制音符；Fader 1D 会录制数值自动化；Fader 2D 会录制 X、Y 和 Z；Button 会在播放头位置录制开状态的步。在一段已有内容上录制会替换掉原来的内容。
- **录制未预备**：Live 区域只播放声音和 MIDI，不进行录制，因此您可以在演奏型运行时试听某个声部或跟随演奏。

预备状态是实时逐刻读取的，因此您可以在一次录制中随时切入切出：它绝不会锁定在手势开始的那一刻。

对于键盘，Live 镜像相对于常规 Layout 旋转了四分之一圈：音高沿着这条竖条竖直排列，因此向上滑动对应更高的音符，向下滑动对应更低的音符。该镜像与网格共享同一个音高滚动位置，因此点亮的可演奏行会与网格中的同一行对齐。Fader 和 Button 的镜像则按正常方向渲染，不会随音高滚动而移动。

9.4 一个单元格意味着什么：音高网格

对于键盘演奏型，网格是一个半音阶钢琴卷帘（piano roll）。纵轴每一行代表一个 MIDI 半音，每行高两个 LED，各行的明暗如同钢琴键：白键对应的半音微微发光，黑键对应的半音保持暗淡。行越高，音高越高。一个音符被绘制成一个琥珀色的小音符头加一条尾巴：

- **水平位置**表示音符的起始时间。
- **宽度**表示闸门长度，即音符持续多久。
- **纵向行**表示音高。
- **音符头亮度**表示力度，即这个音符敲击的力度大小。尾巴的亮度则跟随所录制的压力。

播放头经过某个音符时，该音符会变亮。

9.4.1 相对音阶的录制方式

音符的存储方式取决于所用的键盘：

- 在 **scale（音阶内，in-scale）键盘**上，音符存储的是一个音阶级数，而不是一个固定音高。允许出现低于根音的级数。实际音高会在播放时根据该元素当前的音阶和根音重新计算，因此如果您之后更改音阶或移调根音，整个已录制的演奏型都会随之跟进。一个已录制的级数听起来与手动演奏时完全一致。
- 在 **chroma 键盘或 drumpad** 上，音符存储的是一个绝对音高或垫索引，并且保持固定不变。

一个键盘是 in-scale 还是 chromatic，由该键盘元素上的 **Show Offscale** 开关决定。关闭时每个按键对应一个音阶级数；开启时会显示半音阶按键，各步则以半音位置的形式存储。

微分音自定义音阶回放时音准准确，与您实时演奏时完全相同。

9.5 在网格上编辑步

网格手势的行为取决于您触摸的位置以及拖动的方向。

绘制一个音符：触摸空白处并拖动：

- 手指下方所在的行决定音高。
- 落指时的触摸压力决定力度（按得越用力，音符越响）。之后您可以从音符头纵向拖动来调整它。
- 水平拖动的距离决定闸门长度，会吸附到当前的量化细分（quantize division）。不拖动的单纯点按会生成一个最小长度的音符。
- 绘制过程中的纵向漂移会作为该音符上的一条 pitch bend 曲线被记录下来。轻微的抖动不会被计入：只要手指停留在音符起始所在单元格的范围内，音符就会保持完全笔直，不会记录任何 bend。一旦离开该单元格，从那一点起就会开始捕捉 bend，即使之后手指又漂移回来也是如此。
- 在向右拖动之前，先在起始单元格上向上或向下移动，可设置第一个脉冲的 bend，因此音符可以从偏离其所在行音高的位置开始，再滑入该音高。音符头保持在其所在行上；一个尾巴颜色的较小标记显示音符的起始位置，通过一条暗淡的连接线与音符头相连，并在您按住时跟随您的手指。
- 沿行向右拖动，然后在末端向上或向下移动，会得到一条从该行音高到您停止位置的直线 bend。向左拖回会擦除手指之后的 bend，因此缩短再重新拉长一个音符会重新绘制它。
- 绘制过程中短暂失去接触仍只会生成一个音符，而不是两个：在抬起的位置附近立即重新触摸，就会延续同一个音符。

在已有的音符头上：

- **点按**会删除该音符。
- **向上或向下拖动**会编辑其力度（网格底部为 0，顶部为最大值）。
- **向右拖动**会编辑该音符所聚焦的表现力曲线（具体是哪条曲线，取决于 LCD 顶栏中选中的泳道）。在 Note 泳道上，这会重新绘制 pitch bend，并随您的拖动调整音符长度，其单元格抖动容差与绘制时相同。

在 Pitch 泳道上，网格只编辑已有音符的 pitch bend，就像 Pressure 和 Slide 泳道编辑它们的曲线一样：

- 触摸空白处没有任何作用。
- 在音符头上**点按**会重置其 bend 曲线。音符保留。
- 按住音符头并向上或向下移动，可设置第一个脉冲的 bend，每行一个半音，不改变音符长度。
- 沿音符拖动会在手指经过之处绘制 bend，曲线的其余部分保持不变。这里没有抖动容差：此泳道用于设置精确的数值。

每个音符会同时携带若干条已录制曲线：一条 pitch bend 曲线、一条压力曲线、一条滑音曲线，以及每个已启用按键手势各一条曲线（pressure CC、X abs、Y abs、X rel、Y rel）。力度、压力和曲线数值都以高分辨率存储，比 0 到 127 的 MIDI 范围更精细，因此表现力十分平滑。

当您按住一个音符编辑其曲线时，其他音符的曲线会变暗，从而突出您正在处理的那一个。松开手指后，它们会恢复到完全的亮度。

手势泳道在回放时会发送与实时按键相同的 CC 消息，使用相同的通道和该手势自身的范围，并且在您将一段循环并轨到演奏型时也会被保留。

对于 **Fader 1D**，网格记录一条数值曲线；对于 **Fader 2D**，它记录 X、Y 和 Z；对于 **Button**，点按会切换某一步的开或关状态。

9.5.1 选择并移动一组音符

要一次移动多个音符，先用一根手指静止地按在网格空白处，再触摸第二根手指：这两根手指会定义一个方框，选中方框内的所有音符。拖动第二根手指即可移动整个选区，同时在时间和音高两个维度上移动。如果方框内没有捕捉到任何音符，两根手指都会退回到普通编辑方式。

9.6 在长演奏型中移动

32 列的网格只是演奏型上的一个窗口，而不是全部内容。一个演奏型最多可达 256 步，因此对于长于该窗口的部分，您一次只能看到其中一段，需要滚动或缩放才能到达其余部分。

- **抓取滚动（Grab scroll）：** 按住 **Sequencer** 按钮并在表面上拖动手指。内容会沿两个方向跟随您的手指移动。这是浏览长演奏型的主要方式。不按住 Sequencer 时，在网格上拖动会改为编辑音符。
- **八度滚动（Octave scroll）： Plus** 和 **Minus** 按钮会将音高窗口按一次一个八度向上（Plus）或向下（Minus）跳动。在音阶键盘上，窗口会按音阶级数跳动，并落在下一个根音上。当编码器开关被按住时，八度滚动会被忽略；当您正在编辑底栏中的演奏型 Length 时，Plus 和 Minus 会改为更改长度。
- **缩放（Zoom）：** 按住 **Sequencer** 并按下 **Plus** 可放大（显示的小节更少），按下 **Minus** 可缩小（显示的小节更多）。最大程度的缩小始终会把整个演奏型都框入视野。
- **编码器滚动（Encoder scroll）：** 按住 **Sequencer** 并转动主编码器，可将窗口横向微调，每次点击移动四列。不按住 Sequencer 时，转动编码器会改为移动 LCD 焦点并进行编辑（见下文的音序器 LCD）。

当您增大演奏型 **Length** 时，窗口会重新调整，让新增的步立即可见。如果您此前特意缩放或滚动到某个区域，则会保留您的视图。

Follow Playhead（Sequencer Settings 中的一项，默认关闭）会自动滚动网格，使移动中的播放头保持在屏幕内，一旦它离开窗口就会重新居中。将其保持关闭，可以在演奏型播放时继续编辑一个固定的区域。

Fader 和 Button 演奏型没有音高轴，因此对它们来说只有水平（时间）方向的滚动适用。

9.7 Sequencer 按钮快捷操作

按住 **Sequencer** 按钮并按下另一个控件，即可触发以下演奏快捷操作。它们与上文的滚动、缩放手势可以配合使用：

按住 Sequencer 并按下	动作
N1 到 N8	在音序器模式中，将活动演奏型切换到该槽位。在音序器模式之外，则跨 Layout 启动该槽位作为一个场景。此快捷操作绝不会并轨循环：如需并轨，请使用 Sequencer Settings 中的 **Bounce Loop** 一行。
Encoder switch	敲击设速（Tap tempo，仅限内部时钟）。
Play/Rec	静音或取消静音活动演奏型。双击可取消静音所有演奏型。
Stop	静音或取消静音活动演奏型。双击可静音所有演奏型。
Plus 或 **Minus**	放大或缩小（见上文的 “在长演奏型中移动”）。
Return	清除所聚焦元素的整条轨道（该元素的每一个演奏型）。这是破坏性操作，请注意不要在伸手去滚动或撤销时误触它。

在这些组合键之外，走带按钮在音序器模式中保持其常规角色：当时钟在内部时钟源上处于停止状态时，**Play/Rec** 会启动时钟，否则会切换录制预备状态；**Stop** 会停止时钟。

9.8 音序器 LCD

LCD 是一个固定的三段式画面：顶部是顶栏（Header），中间是网格（Grid），底部是底栏（Footer）。转动主编码器（在不按住 Sequencer 按钮的情况下）可在三个可选控件之间移动高亮，顺序为顶栏、然后 Division、然后 Length，并循环往复。点击编码器可开始编辑高亮的控件，再次点击则停止编辑。控件处于编辑状态时，其数值会闪烁，其图标保持点亮。

9.8.1 顶栏：泳道选择器

顶栏一次显示一条表现力泳道，以一个白色圆盘中的图标加一个小写标签表示。要更改泳道，请高亮顶栏，点击编码器进入编辑状态，转动编码器循环切换泳道，再次点击退出编辑。

系统只会提供对所聚焦元素有意义的泳道：

元素	泳道循环顺序	标签
Keyboard	Note、Pressure、Slide、Pitch，然后是每个已启用按键手势各一条泳道（循环）	note、pressure、slide、pitch、pressure CC、X abs、Y abs、X rel、Y rel
Fader 2D	X、Y、Z（循环）	X、Y、Z
Fader 1D	仅 Value	value
Button	仅 State	steps

Note 泳道会将音符显示为位于各自音高行上的横条。**Pressure**、**Slide** 以及各个手势泳道会在每个音符之上绘制已录制的曲线，并使音符头变暗，从而让曲线作为前景显示。**Pitch** 以音符尾巴的形状显示每个音符的 pitch bend（没有 bend 的音符显示为一条平直的横条），并让网格编辑该 bend，而不会创建、删除音符或调整其长度。手势泳道只会针对聚焦元素上已启用的手势出现（在其 MIDI 设置中）；禁用某个手势会移除对应的泳道，如果该泳道当时正被选中，顶栏会回退到 Note。切换泳道会重新指定 LED 网格所能编辑的内容并重新标注顶栏，但 LCD 网格始终会继续显示该元素的全部数据。

顶栏还有两种特殊状态。如果一个演奏型的内存用尽，顶栏会以红色显示 **Pattern full**（演奏型已满）约几秒钟。如果您正在音序化的元素被删除，顶栏会以灰色显示 **Element deleted**（元素已删除），而不是一条空白的泳道。

9.8.2 网格：全局地图

LCD 上的网格始终从头到尾绘制整个演奏型。一个半透明的轮廓框标出当前映射到垫表面上的那一段：水平方向上它跟踪您的时间位置，对于键盘还会在垂直方向上收缩到表面所对应的音高带，从而显示您的手指目前处于哪个八度。推子和按钮没有音高轴，因此它们的框会占满整个高度，只在水平方向移动。

音符显示为白色横条，其纵向位置表示相对音高；音高轴会自动缩放以适配当前存在的音符。推子泳道显示为已录制数值的阶梯折线加圆点曲线（在 2D 推子上，X 轴为白色，Y 轴为红色）。按钮演奏型会将其开状态的步显示为居中的横条。在较长的演奏型上，步和小节的网格线会自动变细，以保持可读性。

右边缘的一条竖条会显示全部 8 个演奏型槽位：活动槽位是一个带有其编号的红色方块，其余为灰色圆点。演奏型播放时，一条移动的白线标出播放头，停止时该线消失。窗口框和槽位竖条都只是只读反馈，切换演奏型请在表面上的 Launcher 进行。

9.8.3 底栏：细分与长度

底栏包含三项内容：左侧是量化**细分**（division），中间是 Layout 的实时缩略图，右侧是演奏型**长度**（length）。细分和长度都是可选中的：高亮其中一项，点击进入编辑，然后转动编码器改变数值。

控件	取值	默认值
量化细分	Off、1/16、1/12、1/8、1/6、1/4、1/3、3/8、1/2、3/4、1、5/4、2、3、4	1/4
演奏型长度	1 到 256 步，每次编码器点击调整一步	16 步

在您编辑 Length 时，**Plus** 会将长度加倍，**Minus** 会将其减半（在 1 到 256 步范围内）；离开编辑状态后，它们会恢复为八度滚动功能。

中间的缩略图是 Layout 的一个小型实时画面：您正在音序化的元素会被高亮显示，并且每个元素在其步触发时都会闪烁，因此您可以看到哪些轨道正在触发。

9.8.4 本画面中的编码器与 Return

- **点击编码器**可切换高亮控件（泳道、细分或长度）的编辑状态。
- **Return** 会在有编辑正在进行时退出该编辑；否则会撤销您最近一次的音序器编辑。

9.9 切换演奏型

每个 Layout 都有 8 个独立的演奏型槽位，各自拥有自己的长度、量化细分和音符。点按 Launcher 区域中的一个垫块（最上方的垫块对应演奏型 1）即可切换。垫块只在触摸按下时响应。切换如何生效取决于 Sequencer Settings 中设置的 **Switch Mode**：

- **Wait：**新的演奏型会被排队，在当前循环结束时接管。待处理的 Launcher 垫块会在表面上闪烁，直到它接管为止。
- **Jump：**新的演奏型立即接管，并从小节中相同的位置继续播放。
- **Restart：**新的演奏型立即接管，并从第一步重新开始。

9.10 Sequencer Settings

在音序器视图处于活动状态时按下 **Settings**，即可打开 Sequencer Settings；再次按下 **Settings** 返回，或按 **Return** 离开并进入默认视图。转动编码器可在各项之间移动，点击可执行操作。

条目	说明
Clear Pattern	仅清除活动演奏型的全部步数据。会以一条消息确认，并返回音序器。

条目	说明
Clear Element	清除所聚焦元素在每一个演奏型中的全部步数据。会以一条消息确认，并返回音序器。
Switch Mode	Wait、Jump 或 Restart，如上文所述。
Copy Pattern	将活动演奏型复制到另一个槽位。画面会提示您在 Launcher 中点按一个演奏型垫块；点按目标垫块即可完成复制。Launcher 上的垫块会显示哪些槽位为空、哪些已经有音符。如果目标已有音符，或者源比目标长，会弹出确认让您选择 Replace、Extend、Trim 或 Cancel。点按源垫块会提示您选择另一个演奏型；Return 会静默取消。
Copy Element	与 Copy Pattern 相同，但只将所聚焦元素的音符复制到目标演奏型中的同一元素上。如果该元素没有内容可复制，会以一条消息拒绝。
Bounce Loop	将 Looper 在此 Layout 上录制的循环转换为音序器音符，放入您当前所在的演奏型，替换该演奏型原有的内容，并清除该循环。如果该演奏型已有音符，会弹出确认，说明该演奏型并列出仍为空的演奏型，提供 **Bounce** 和 **Cancel**；取消不会做任何更改。会以一条命名了该演奏型的消息确认。当没有可并轨的循环（`No loop to bounce`，没有可并轨的循环）或循环太大无法放入一个演奏型（`Loop too big to bounce`，循环过大，无法并轨）时，会以一条消息拒绝。无论时钟是否在运行都可执行。这一行是并轨循环的唯一方式。
Reset Pattern Settings	将活动演奏型的长度、量化网格和播放方式恢复为默认值。音符会被保留。如果该演奏型中有音符超出默认长度，会弹出确认说明这些音符仍会被保留存储，但不会再播放。
Follow Playhead	自动滚动网格，使播放头保持在屏幕内。默认关闭。
Swap Sides	将 Live 与 Launcher 区域左右互换。网格保持居中。这是一项设备级设置，不属于项目的一部分。

在一次复制操作等待目标位置时切换 Layout，会以一条消息取消该复制操作。

并轨会通过真实的 Layout 元素重新演绎您所录制的手势，因此该元素的配置和 MIDI 效果都会被烘焙

进最终生成的音符中，并且已启用的手势泳道也会被保留。

9.11 自动保存

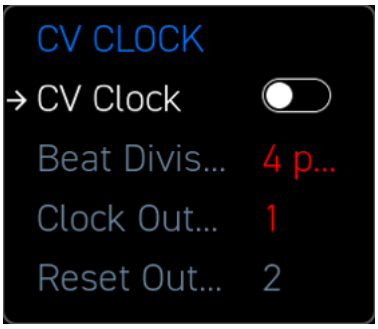
任何会改变演奏型内容的编辑（绘制或录制音符、清除某个演奏型或元素、复制进某个演奏型、并轨循环，或重置演奏型设置）都会随您项目的其余部分自动保存。保存完成后会出现一个简短的保存指示，因此您永远不需要手动保存音序器的内容。

9.12 时钟

音序器运行在项目时钟上，与 Looper 使用的是同一个时钟。项目时钟源可以是 INT（内部）、USB-dev、MIDI 或 USB-host 之一。在内部时钟下，音序器会以项目速度播放。当项目从属于外部时钟时，它会跟随该时钟，并遵循 Start、Stop 和 Continue 指令，且在收到 Start 或 Continue 之前不会运行。

10 CV 输出

Erae 2 在其后面板上提供 **24 个 CV 输出通道**。这些输出可以传递音高电压、门信号或连续控制电压，使 Erae 2 在具备 MIDI 功能的同时，也成为一个模块化接口。



CV Clock

10.1 硬件概览

24 个输出分布在 12 个 TRS 迷您插孔上，每个插孔承载两个输出：插孔 n 的尖端（tip）承载输出 n，环（ring）承载输出 n + 12。直接使用插孔可获得一个输出，使用 TRS 分线缆则可同时获得两个输出。

- **输出范围：** -5 V 到 +8 V
- **音高标准：** 1 V/octave
- **门信号电平：**关为 0 V，开为 5 V

插孔 11 和 12 同时用作 **Pedal A** 和 **Pedal B** 的输入，因此输出 11、12、23 和 24 与踏板共用。

> **警告：** CV 输出在低功耗模式下被禁用。要使用它们，请用随附的电源适配器为 Erae 2 供电（参见快速入门）。

10.2 输出模式

每个 CV 输出传递一种信号，在元素配置中设置：

Pitch（1 V/oct）：输出遵循 1 V/octave 标准，MIDI 音符 0 对应 `-3 V`。中央 C（MIDI 音符 60，`C3`）输出 `2 V`，每个半音对应 `1/12 V`（约 `83.3 mV`）。琴键的 pitch bend 会计入电压。完整的 MIDI 音符范围都在输出范围之内。

Gate：没有音符按住时输出为 `0 V`，有音符按住时输出为 `5 V`。**Gate Mode** 可以是 `Gate`（音符按住期间保持高电平）、`Trig`（每个音符开始时输出一个短脉冲，25 到 `250 ms`）或 `Retrig`（与 Gate 类似，但在每个新音符时短暂回落，以便包络重新开始）。可选的 **Gate Delay**（`0` 到 `5 ms`）会延迟门信号，让音高先稳定下来。

连续控制：输出传递与某个数值（如推子位置、手指压力或滑动位置）成正比的电压。默认情况下，数值映射到 `0 V` 到 `5 V`；您可以在输出范围内设置最小和最大电压。

> **提示：**一个 Gate 输出后接一个 Pitch 输出，就构成了经典的 Gate 加 V/oct 组合，可用于驱动单音模拟合成器。

10.3 按元素分配 CV

CV 输出在 Erae Lab 的元素配置中的 **Control Voltage** 部分分配：

- **Enable：**为该元素开启 CV。
- **Output：**该元素使用的第一个 CV 输出。
- **Num voice：**在键盘上，同时发声的 CV 声部数量（复音数），1 到 15。

每个声部为每个已启用的信号使用一个输出，顺序如下：**Gate**、**1V / Oct**、**CV Velocity**、**CV Pressure**、**CV X abs**、**CV Y abs**、**CV X rel**、**CV Y rel**。各声部依次排列：第二个声部从第一个声部最后一个输出之后的输出开始。

例如，启用 Gate 和 1V / Oct，将 **Output** 设为 1、**Num voice** 设为 4 时，输出 1 和 2 承载第一个声部的门信号和音高，输出 3 和 4 承载第二个声部，依此类推直到输出 8。演奏时，音符会被分配给空闲的声部。

> **提示：**对于四声部音色，启用 Gate 和 1V / Oct，将 **Num voice** 设为 4，把四个音高输出连接到四个 VCO，把四个门信号输出连接到四个包络发生器。

10.4 每个 Voice 的额外 CV 输出

除了核心的 Gate 和 Pitch 组合外，每个键盘声部还可以将表现力作为 CV 输出。每条已启用的数据流会让每个声部额外占用一个输出：

输出	说明
CV Velocity	音符的击打力度（默认 0 V 到 5 V）
CV Pressure	连续的手指压力（默认 0 V 到 5 V）
CV X abs	手指在琴键内的水平位置（默认 0 V 到 5 V）
CV Y abs	手指在琴键内的垂直位置（默认 0 V 到 5 V）
CV X rel	相对于手指落点的水平移动，从一个初始值开始（默认 2.5 V）
CV Y rel	相对于手指落点的垂直移动，从一个初始值开始（默认 2.5 V）

启用全部八种信号时，一个声部占用 8 个输出，因此 24 个输出最多容纳三个这样的声部。请在 Layout 的所有元素之间规划分配。输出重叠时 Erae Lab 会发出警告，但不会阻止。

其他元素提供以下 CV 输出：

- **Fader 1D： Pressure** 和 **Y abs**。
- **Fader 2D： Pressure**、**X abs** 和 **Y abs**。
- **按钮（Note）： Gate** 和 **1V / Oct**。
- **按钮（Control Voltage）：** 一个带开电压和关电压的输出。
- **踏板：** 根据踏板类型提供门信号、力度或连续输出。

> **提示：** 使用 **CV Pressure** 驱动一个 VCA，从而完全在模块化领域内实现音符级动态；使用 **CV X rel** 控制颤音深度。

10.5 Eurorack 快速上手

10.5.1 第一个 Voice 设置（Pitch + Gate）

1. 在 Erae Lab 中选择一个 Keygrid 元素并启用 **Control Voltage**，开启 **Gate** 和 **1V / Oct**，并将 **Num voice** 设为 1。
2. 将 **Output** 设为 1。输出 1 承载门信号，输出 2 承载音高。
3. 将输出 1 连接到包络发生器的 Gate 输入。
4. 将输出 2 连接到 VCO 的 V/oct 输入。
5. 弹奏一个音符：VCO 按 1 V/octave 跟踪音高，包络会在每个音符时触发。

输出 1 和 2 位于插孔 1 和 2（尖端）。使用分线缆时，插孔 1 的环还承载输出 13。

10.5.2 USB Host MIDI 转 Eurorack

将一个 USB MIDI 转 CV 模块（例如 Expert Sleepers FH-2 或 Intellijel uMIDI）连接到 Erae 2 的 **Synth**（USB Host）端口，并将元素路由到 **USB Host**。配置该模块以接收来自 Erae 2 的 MIDI 并转

换为 CV。在需要更高复音数时，这条路径可补充 Erae 2 的原生 CV 输出。

10.5.3 24 ppqn 的 CV Clock

1. 打开 **Settings > CV Clock**。
2. 开启 **CV Clock**。
3. 将 **Beat Division** 设为 24 ppqn。
4. 将 **Clock Output** 设为一个未使用的输出（例如输出 3）。
5. 将输出 3 连接到您的受时钟驱动模块或时钟分频器的时钟输入。
6. 将下一个输出（此处为 4），即 **Reset Output**，连接到该模块的 reset 输入。

10.5.4 推荐的 Eurorack 模块配置

使用场景	推荐模块
MIDI 转 CV（多声部）	Expert Sleepers FH-2
MIDI 转 CV（单声部，简单）	Intellijel uMIDI
高复音 V/oct + Gate	Mutable Instruments Yarns
无 MIDI 桥接的 Erae 2 直接 CV	原生 CV 输出（输出 1 到 24）

10.6 校准

CV 输出在 Embodme 出厂时已完成校准。如果您观察到音高跟踪误差，可以重新运行校准：

1. 拔下所有 CV 插头和踏板。
2. 用随附的电源适配器为 Erae 2 供电。
3. 在 **Settings** 中选择 **Run CV calib**。
4. 校准需要几分钟，完成后 Erae 2 会自动重启。

校准后，您可以在 **Settings** 中通过 **CV Gain (cent/oct)** 微调音高缩放。

> **提示：**为获得最准确的结果，请在设备使用几分钟、处于正常工作温度时进行校准。

10.7 CV 时钟输出

CV 时钟输出在某个 CV 输出上发送与项目速度同步的时钟脉冲。这是一个项目级设置：它作用于所有 Layout，而非针对某个特定元素。

10.7.1 访问 CV Clock 设置

按下前面板上的 **Settings** 按键，然后选择 **CV Clock**。**CV CLOCK** 画面随即打开。

10.7.2 CV Clock 参数

CV Clock：时钟输出的开关。默认关闭。开启时，时钟输出和 reset 输出会被预留给时钟使用。

Beat Division：相对于项目速度的脉冲速率。可用取值：

取值	说明
1 ppqn	每个四分音符一个脉冲（每拍 1 个时钟）
2 ppqn	每个四分音符两个脉冲
4 ppqn	每个四分音符四个脉冲（每个十六分音符一个），默认值
8 ppqn	每个四分音符八个脉冲
24 ppqn	标准 MIDI 时钟速率（每个四分音符 24 个）
48 ppqn	高分辨率时钟，每个四分音符 48 个

24 ppqn 是标准的 MIDI 时钟速率。使用 4 ppqn 可获得十六分音符触发，使用 1 ppqn 可获得每拍一次的触发。

> **提示：**许多 Eurorack 模块期望 24 ppqn 或 48 ppqn 的时钟。将 Erae 2 的 CV Clock 输出直接连接到该模块的时钟输入即可：无需时钟分频模块。

Clock Output：承载时钟脉冲的 CV 输出。范围：1 到 23。

Reset Output：只读，始终为 Clock Output 之后紧接的输出。它发送 reset 脉冲，使模块与演奏型起点保持对齐。它无法被禁用或单独分配。

> **提示：**将 Clock 连接到模块的时钟输入，将 Reset 连接到其 reset 输入：Erae 2 会让两者始终与其内部演奏型起点保持对齐。

> **提示：**只要项目时钟在运行，CV 时钟就会持续运行，与当前是哪个 Layout 无关，因此您可以在演出过程中切换 Layout 而不打断模块化时钟。

11 第 11 章：LCD 界面

Erae 2 配备了一块 280 × 240 像素的彩色 LCD。您可以使用旋转编码器及其按压开关，配合第 12 章所述的面板开关来浏览每一个界面。开机时，LCD 会先短暂显示固件版本，然后出现主界面。

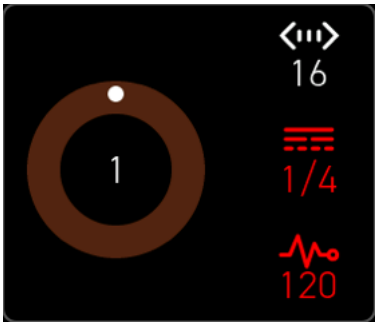
11.1 11.1 编码器导航模型

操作	动作
旋转	在屏幕上的各项之间移动焦点高亮
短按	确认选择、打开子界面或切换开关

在数值行（数字，以及音阶、速率或风格之类的列表）上，短按会进入编辑模式：此时旋转会更改数值。再次短按可确认并退出编辑模式。

> **提示：**按 **Return** 可离开当前界面。在设置的子界面（Pedal、Routing、CV Clock、Velocity Curve、Save、Load）中，它会返回设置；在设置、Mapping、Scale、Arpeggiator 和 Custom Scales 界面中，它会返回主界面。

11.2 11.2 主界面



主界面

主界面是 LCD 的默认状态。它显示当前 Layout 的 Looper：

- **Looper 圆环**：一段圆弧，显示循环内的播放位置。中间的数字是 Looper 编号，即当前 Layout 的编号（每个 Layout 都有自己的 Looper）。当该 Looper 被静音时，圆弧会改变颜色。
- **Length**：以步（十六分音符）为单位的循环长度，范围为 1 到 256。
- **Quantize Grid**：Looper 的量化网格（`Off`、`1/16`、`1/12`、`1/8` 等）。
- **Tempo**：以 BPM 表示的项目速度，经过平均处理以平滑外部时钟的抖动。仅当 **Clock source** 为 `INT` 时，才能在此编辑 Tempo。

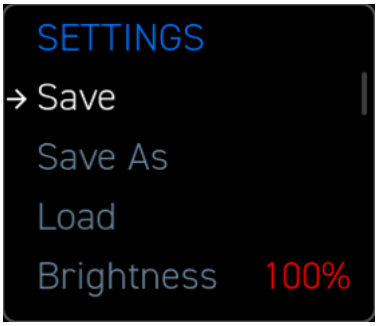
旋转编码器可高亮 Length、Quantize Grid 或 Tempo，按下编码器即可编辑高亮的数值。编辑 Length 时，**Plus** 会将其加倍，**Minus** 会将其减半。编辑 Tempo 时，**Plus** 会将速度加倍，**Minus** 会将其减半。

在主界面上按 **Return** 会移除最近录制的一层 Looper 内容。

Layout 通过 **N1** 至 **N8** 面板开关进行切换。LCD 上没有 Layout 选择器界面。

每当设备自动保存您的项目时，左上角会淡入淡出一个带 SD card 图标的橙色小徽标。

11.3 11.3 设置菜单 {#113-settings-menu}



设置菜单

按下 **Settings** 开关即可打开设置菜单，这是一个垂直滚动的列表。再次按下 **Settings** 或按 **Return** 可返回主界面。旋转以高亮某一行，短按以激活。各行依次为：

项目： **Save**（显示当前项目名称）、**Save As**、**Load**。

全局： **Brightness**（5% 到 `100%`）、**Sensitivity**（列表）。

项目设置： **Tempo**（1 到 999 BPM）、**Clock source**（`INT`、`USB-dev`、`MIDI`、`USB-host`）、**Metronome**（开关）、**CC on layout change**（开关）。

子界面： **Velocity Curve**、**Pedal A**、**Pedal B**、**Custom Scales**、**Routing**、**CV Clock**。

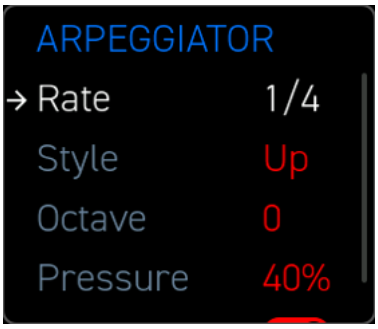
备份与出厂： **Save Backup Project**、**Load Backup Project**、**Load Factory Project**。

校准与系统：编码器校准状态（只读）、CV 校准状态（只读）、**CV Gain (cent/oct)**（-50 到 +50）、**Run CV calib**、**MIDI 2.0: ON/OFF**、**Format SD Card & Reboot**。

未挂载 SD card 时，**Save**、**Save As** 和 **Load** 显示为灰色。在演示模式下，标题显示为 DEMO MODE，列表精简为 **Reset Demo**、**Brightness**、**Sensitivity**、**Tempo**、**Clock source** 和 **Metronome**。

每项设置都在第 15 章中介绍。

11.4 11.4 琶音器设置界面



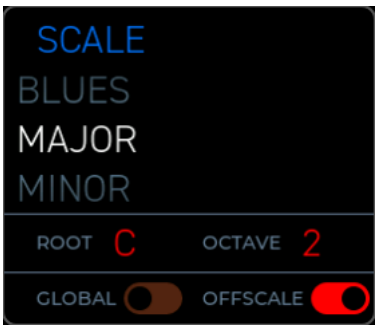
琶音器设置

选中一个键盘元素，然后按 **Arp**：该元素的琶音器会被开启或关闭，同时打开此界面。若要打开此界面而不切换琶音器的开关状态，请按住 **Settings** 并按 **Arp**。

参数	类型	取值
Rate	列表	1/32、1/16、1/8、1/4、1/2、1/1、Pressure
Style	列表	Up、Down、UpDown、UpAndDown、Random
Octave	数值	0 到 8
Pressure	数值	0% 到 100%，步长为 10（手指压力转力度）
Quantize	开关	将琶音器锁定到项目时钟

更改即时生效。各参数的行为请参阅第 8 章。

11.5 11.5 音阶 / 调式选择器 {#115-scale–key-selector}



音阶选择器

音阶界面用于编辑一个键盘元素的音阶。按住 **Scale**：除焦点元素外的所有内容都会变暗，此时触摸某个元素会选中它而不是演奏它。松开 **Scale** 即可为所选元素打开该界面。再次按 **Scale** 可将其关闭。

标题行只显示 **SCALE**。其下界面包含：

- **音阶列表**（全宽）：出厂音阶，后面依次是您的自定义音阶，以及当键盘可以使用自定义音阶时出现的 **NEW SCALE** 条目。先选中它，再转动编码器进行浏览。
- **ROOT** 和 **OCTAVE** 数值：根音为 C 到 B，八度为 -2 到 8。
- **GLOBAL** 和 **OFFSCALE** 开关（keygrid）：**OFFSCALE** 显示或隐藏音阶之外的半音。**GLOBAL** 开启时，键盘跟随项目的全局音阶：在任何开启了 **GLOBAL** 的键盘上更改音阶，都会同时更改所有这些键盘的音阶。根音和八度始终按元素单独保存。

转动编码器，高亮会按此顺序移动：音阶、根音、八度、**GLOBAL**、**OFFSCALE**。按下数值即可编辑，转动来更改，再按一次确认。按下开关即可切换。

元素特定行为： - **Iso Keyboard**：所有控件均可用。仅当键盘启用了 MPE 时，自定义音阶才会出现在列表中。- **Chroma Keyboard**：仅有音阶列表、**ROOT** 和 **OCTAVE**。音阶改变琴键着色；半音布局保持不变。没有 **OFFSCALE** 或 **GLOBAL** 控件。- **Drumpad**：**ROOT** 和 **OCTAVE** 两个列表并排显示，另有一个 **GLOBAL** 开关；没有音阶列表。

选择 **NEW SCALE** 并按下编码器，会打开自定义音阶编辑器（参见11.16）。

> **提示：**使用 **ROOT** 和 **OCTAVE** 数值，可在不通过 Erae Lab 编辑 Layout 的情况下快速对 Iso Keyboard 进行移调。

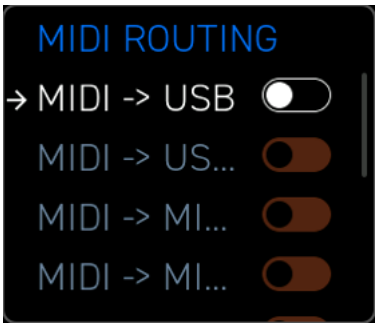
11.6 11.6 信息界面（状态浮层）

一个自动出现、用于报告某种状态的消息界面。**Project Saved** 会在两秒后消失；其他消息会一直显示，直到您按下任意面板开关或编码器。消息包括：

LCD 消息	含义
`Please insert microSD card and reboot`	启动时未找到可用的 SD card
`Project Saved`	已完成保存到备份槽位
`Low Power (Low brightness, CV, MIDI B, Metronome disabled)`	电源功率不足以支持完整运行
`SD Read/Write error`	SD card 本身出现故障
`Project corrupted (Loaded factory default)`	某个项目无法读取，已改为加载工厂项目
`Project read failed (File may be corrupted)`	某个项目文件无法读取
`Project save failed (Please try again)`	保存未能完成
`Project file not found`	SD card 上缺少所请求的项目
`Legacy project Please connect ERAE Lab`	发现了来自旧版固件的项目；Erae Lab 会转换它们。当 Erae Lab 开始发送转换后的项目时，该消息会自行关闭

当 SD card 没有可读的文件系统时，会出现确认界面询问 `No filesystem found Format SD card?`。选择 **Yes** 会格式化该卡并重启 Erae。

11.7 11.7 路由界面



MIDI 路由

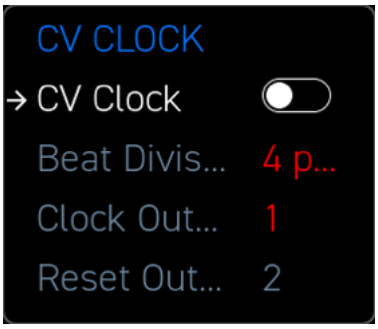
通过 **Settings -> Routing** 进入。这是一组开/关开关，用于将传入的 MIDI 从一个端口转发到另一个端口：

MIDI -> USB Host、USB Device、MIDI A、MIDI B **USB Device ->** USB Host、MIDI A、MIDI B **USB Host ->** USB Device、MIDI A、MIDI B

短按某个开关即可切换。路由随项目一起保存。

> **提示：**启用 **MIDI -> USB Device** 可让 Erae 2 充当硬件合成器与您的 DAW 之间的 MIDI 合并器，而无需额外软件。

11.8 11.9 CV 时钟界面



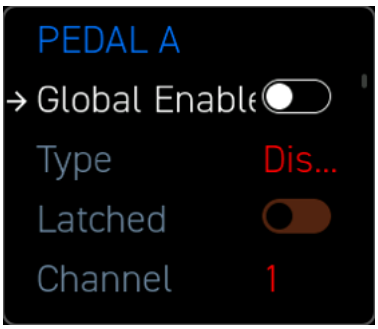
CV 时钟

通过 **Settings -> CV Clock** 进入。

参数	类型	说明
CV Clock	开关	开启或关闭 CV 时钟输出
Beat Division	列表	1 ppqn、2 ppqn、4 ppqn、8 ppqn、24 ppqn、48 ppqn
Clock Output	数值	承载时钟脉冲的 CV 输出（1 到 23）
Reset Output	只读	始终为 Clock Output 之后紧接的输出（2 到 24）

> **提示：**将 **Beat Division** 设为 24 ppqn，即可为期望标准 24 ppqn 时钟的 Eurorack 模块提供时钟。

11.9 11.10 踏板输入配置 {#1110-pedal-input-configuration}



踏板输入配置

通过 **Settings -> Pedal A** 或 **Pedal B** 进入。界面标题为 `PEDAL A` 或 `PEDAL B`。

Global Enable（开关）始终显示。开启后，**Type** 用于选择踏板模式：`Disabled`、`Switch`、`Expressive`、`Sustain`、`Kick`。其他行会根据类型出现：

- **Switch**：**Out Routing**、**Latched**、**Channel**、**Message**（`Note`、`CC`、`PC`、`Tap Tempo`），然后是该消息对应的音符、CC 开关值或程序开关值，以及 **CV Enable** 及其 CV 输出和开关电压。
- **Expressive**：**Out Routing**、**Calibrate**、**Channel**、**CC Type**（`CC`、`RPN`、`NRPN`、`CC14bit`）、**CC Number**、**Invert**、**CV Enable**。
- **Sustain**：**Mode**（`Binary` 或 `Continuous`）。Binary 显示 **Out Routing**、**Latched**、**Channel**、**CC Type**、**CC Number**、**On Value**、**Off Value**、**CV Enable**。Continuous 显示 **Out Routing**、**Calibrate**、**Channel**、**CC Type**、**CC Number**、**Invert**、**CV Enable**。
- **Kick**：**Out Routing**、**Calibrate**、**Channel**、**Note**、**Velocity Sens**、**Note Duration**（`25ms`、`50ms`、`100ms`、`200ms`）、**Impact Thresh**、**CV Gate**（门信号输出及开关电压）和 **CV Velocity**（力度输出及最小和最大电压）。

Out Routing 会打开该踏板的元素路由界面（参见11.12）。**Calibrate** 会打开校准向导。

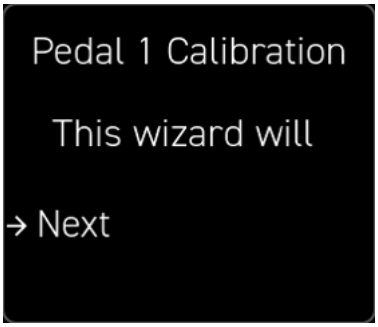
11.9.1 11.10.1 格式化 SD card 确认对话框

当您在设置菜单中选择 **Format SD Card & Reboot** 时，会出现确认界面询问 `Format SD card? (Erae will reboot)`：

- **Yes**：格式化 SD card 并重启设备。
- **No**：取消并返回设置菜单。

MIDI 2.0 一行在重启 Erae 之前也使用同类确认。

11.9.2 11.10.2 踏板输入校准向导



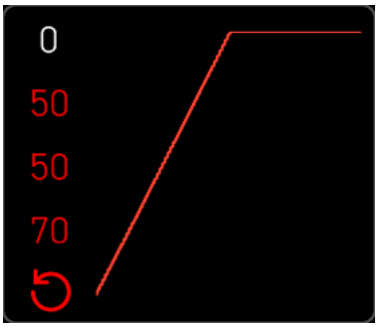
踏板输入校准

一个分步向导，通过踏板界面上的 **Calibrate** 进入：

- **Expressive 和 Sustain Continuous：**先是 Welcome，然后是 **Capture Min**（踏板完全松开，脚跟位置），再是 **Capture Max**（踏板完全踩下，脚尖位置）。如果范围过小，向导会报告失败并提供 **Retry**。
- **Kick：**先是 Welcome，然后在踏板静止时进行 **Capture**，接着至少轻敲 3 次，再至少用力踩踏 3 次。如果用力踩踏没有明显重于轻敲，向导会报告失败并提供 **Retry**。

校准完成后，结果会保存在设备上。

11.10 11.11 力度曲线编辑器



力度曲线

通过 **Settings -> Velocity Curve** 进入。实时曲线图展示手指压力如何映射到 MIDI 力度。左边缘纵向排列着四个数值按键；屏幕上只显示数值本身，没有标签，因此请记住它们从上到下的顺序：

位置	参数	说明
1（最上）	**Threshold**	力度开始上升之前所需的接触量
2	**Drive**	弯折曲线：高于 50 时较轻的触摸也能达到较高力度，低于 50 时需要按得更用力
3	**Compand**	扩展或压缩动态范围的中段（50 为中性值）
4	**Range**	整体灵敏度：数值越高，达到满力度所需的压力越小。它不是力度的上限

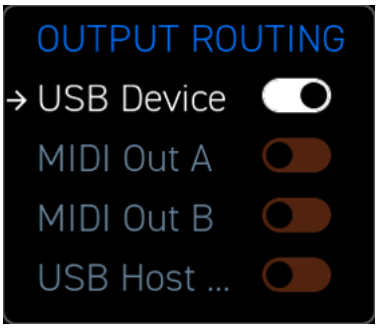
转动编码器可在四个数值和重置箭头之间移动高亮，点击进入编辑，此后每次转动编码器都会增减 5（取值范围为 0 到 95）。**Reset** 箭头可将其恢复为出厂默认值（0、50、50、70）。曲线图会随参数变化实时重绘；其水平轴为便于查看而做了缩放处理，请勿将其读作绝对压力值。**Return** 返回 Settings。

> **提示：**较高的 **Threshold** 配合较低的 **Range**，会让触摸表面更接近传统键盘的手感：需要有意识地施加压力音符才会被记录，且需要更用力的按压才能达到高力度。

11.11 11.11b Sequencer 画面

在步进音序器处于活动状态时，LCD 会显示 Sequencer 画面（泳道顶栏、完整演奏型地图，以及 division/length 底栏），并且 **Settings** 开关会打开 Sequencer Settings 菜单，而不是全局 Settings。这两个画面都在步进音序器一章中有详细说明。

11.12 11.12 元素路由界面 {#1112-element-routing-screen}

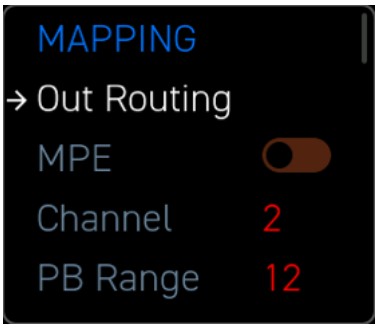


元素路由

`OUTPUT ROUTING` 界面用于选择哪些 MIDI 输出接收来自焦点元素或踏板的数据。可通过 Mapping 界面或踏板界面上的 **Out Routing** 进入。

USB Device Out、**MIDI Out A**、**MIDI Out B**、**USB Host Out**：任意组合均可同时开启。更改即时生效。

11.13 11.13 映射界面



映射

Mapping 界面用于编辑一个元素的 MIDI 和 CV 参数。按住 **Mapping**：除焦点元素外的所有内容都会变暗，此时触摸某个元素会选中它而不是演奏它。松开 **Mapping** 即可为所选元素打开该界面。再次按 **Mapping** 可将其关闭。参数因元素类型而异：

- **Button**：按钮类型（note、control change、program change、tap tempo、CV）、通道、音符、锁定、带可选库选择的 CC 值和程序值、CV 输出设置。
- **Fader 1D / Fader 2D**：每个轴的 CC、默认值、中心值、最小值和最大值（2D 额外增加 X 轴）、可选的压力 CC、CV 输出。
- **Key**：X 和 Y 的绝对与相对 CC、pitch bend 范围、压力设置、CV 输出。
- **Keyboard**：通道、MPE、pitch bend 范围、滑音、滑移（CC74）、压力、X 和 Y CC、行偏移量，以及 CV 部分（音高、门信号和可选的表现力输出）。

每种元素类型还会显示其传输组。底部的 **Out Routing** 按钮会打开该元素的元素路由界面。

11.14 11.15 保存 / 另存为 / 加载界面

保存项目

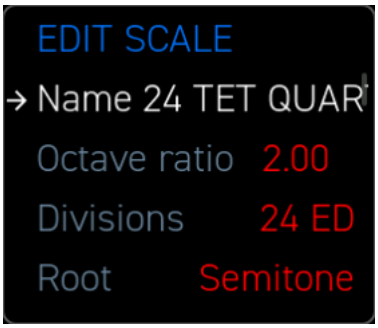
Erae 会自动保存您的工作：在您更改某些内容后不久，当前项目就会被写入 SD card（左上角会出现 SD 徽标）。设置中的项目条目涵盖项目生命周期的其余部分：

- **Save**：将当前项目写入其 SD card 条目。如果该项目还没有 SD 条目（备份项目或工厂项目），则会改为打开 Save As。
- **Save As**：打开 `SAVE AS` 列表：**+ CREATE NEW**，后面是 SD card 上的项目，当前项目以 * 标记。选择一个条目来命名项目：演奏表面上会出现一个字母键盘，并预先填入您所选的名称。在表面上输入，向后转动编码器可删除一个字符，向前转动可添加一个空格，按下编码器确认，或按 **Return** 取消。
- **Load**：打开 `LOAD PROJECT` 列表。短按某个项目即可加载它；它会替换当前项目。
- **Save Backup Project**：将当前项目写入内部备份槽位。`Project Saved` 会确认此操作。
- **Load Backup Project**：加载备份槽位。挂载了 SD card 时，它还会以 `Backup`、`Backup_2` 等名称保存到 SD card。
- **Load Factory Project**：将工厂项目作为一个没有 SD 条目的临时项目加载，直到您保存它为止。

备份项目和工厂项目在您保存之前没有 SD 条目，因此在您使用 **Save** 或 **Save As** 之前，它们不会被自动保存，也不会与 Erae Lab 同步。

> **提示：**在加载工厂项目或备份之前，请确认您刚才演奏的项目已有 SD 条目（其名称显示在 **Save** 旁边），这样它就已经安全地保存在卡上了。

11.15 11.16 自定义音阶界面 {#1116-custom-scales-screens}



自定义音阶编辑器

Settings -> Custom Scales 会列出您的自定义音阶，后面依次是 **+ Create scale**（仍有空间时）、**Save scales** 和 **Load scales**。选择一个音阶即可编辑它，选择 **+ Create scale** 则可新建一个。

编辑器（`NEW SCALE` 或 `EDIT SCALE`）从上到下列出这些行：

- **Name**：打开字母键盘。
- **Octave ratio**：音阶重复的音程（`2.00` 为一个八度）。
- **Divisions**：`Free` 表示任意比率，或八度比率的等分（例如 `12 ED`）。更改在您按下离开编辑时生效。

- **Root**：`Semitone` 或 `Frequency`，它决定下一行。这同样在您按下离开编辑时生效。
- **Root Offset**（相对于元素根音的半音偏移）或 **Root Freq**（以 Hz 表示的绝对频率）。
- **DEGREES, CENTS ABOVE ROOT** 区域，每个音级一行，每行显示其高于根音的音高（以 cent 为单位）。
- **+ Add degree**（在等分音阶上隐藏）、**Load Template**（内置调律，例如 Just Intonation Major、Pythagorean 或 Bohlen-Pierce）和 **Delete**（当某个键盘正在使用该音阶时显示为 `In use` 并被禁用）。

按下某个音级即可打开它的页面：**Ratio**、**Semitone**、**Cents**（在半音范围内为 `0` 到 99）和 **Delete degree**。在等分音阶上，编辑某个音级会把该音阶变为自由音阶。**Return** 会返回编辑器，并高亮该音级所在的行。

音阶会在您停止编辑约 2 秒后自动保存，离开编辑器时则立即保存。

自定义音阶只能分配给启用了 MPE 的 Iso Keyboard。在 Erae Lab 中构建和管理自定义音阶（包括导入 Scala 文件）的内容，请参阅 Erae Lab 手册中的自定义音阶一章。

---

12 第 12 章：开关与编码器

Erae 2 通过 18 个配有 RGB LED 指示灯的面板开关，以及一个带按压开关的旋转编码器，为您提供直接的物理控制，这些控件全部环绕在 LCD 周围。借助它们，您无需触碰演奏触控面，便可切换 Layout、控制 Looper 和音序器，并浏览 LCD 菜单。

12.1 面板开关

面板上有 18 个开关，每个都有自己的 LED：八个编号开关和十个功能开关。

12.1.1 编号开关（N1 到 N8）

八个标记为 **N1** 到 **N8** 的开关用于选择当前项目的八个 Layout。按下其中一个，即可将整个演奏表面切换到该 Layout。每个 Layout 还有自己的 Looper，编号开关与其他开关组合使用时，也可作为 Looper 控制（参见开关组合）。

> **提示：**将您最常切换的 Layout 放在相邻位置，这样只需拇指一动，就能改变整个演奏表面。

12.1.2 功能开关

开关	功能
Play/Rec	使用内部时钟且时钟停止时，启动时钟。否则预备或取消预备 Looper 录制（在音序器模式下，则为录入
Stop	停止时钟和播放
Plus	将所选元素升高一个八度。在主界面上，将您正在编辑的循环长度或速度加倍
Minus	将所选元素降低一个八度。在主界面上，将您正在编辑的循环长度或速度减半
Scale	按住后触摸某个元素即可选中它，松开后打开它的音阶界面
Sequencer	进入或离开步进音序器；也是音序器快捷方式的按住修饰键（参见第 9 章）
Mapping	按住后触摸某个元素即可选中它，松开后打开它的 Mapping 界面
Arp	开启或关闭所选键盘的琶音器，并打开琶音器界面
Settings	打开或关闭设置菜单
Return	离开当前界面。在主界面上，移除最近录制的一层 Looper 内容

Play/Rec 和 Stop 还会让 LCD 返回主界面（或音序器界面）。在 LivePad 元素（Ableton Live 片段控制）上，**Plus** 和 **Minus** 用于缩放，**Arp** 用于循环切换 LivePad 模式。

12.2 编码器

Erae 2 在 LCD 旁配有一个旋转编码器。它是浏览菜单和输入数值的主要控件。

12.2.1 滚动

旋转编码器可在列表项之间移动、更改数值，或在屏幕上的项目之间逐步移动。每一个档位精确移动一步；编码器不会加速。

> **提示：**要大幅更改速度，请在主界面上编辑 **Tempo**：**Plus** 会将其加倍，**Minus** 会将其减半，然后再用编码器进行微调。

12.2.2 按下

按下编码器可确认高亮的选项：它会打开一个子界面、切换一个开关，或在某个数值上进入和退出编辑模式。

12.3 Return 开关

Return 用于离开当前界面。在设置的子界面（Pedal、Routing、CV Clock、Velocity Curve、Save、Load）中，它会返回设置菜单。在设置菜单、Mapping、Scale 和 Arpeggiator 界面中，它会返回主界面，主界面显示当前 Layout 的 Looper 和当前速度。

12.4 开关组合

按住一个开关的同时按下另一个开关，即可使用以下功能：

组合	动作
Return + N1 到 N8	清除这些 Layout 的循环
Return（按住）+ N1 到 N8	确认后，清除这些 Layout 的循环及全部八个 sequencer 演奏型

组合	动作
Play/Rec + N1 到 N8	取消该 Layout Looper 的静音
Stop + N1 到 N8	静音该 Layout 的 Looper
Settings + Arp	打开琶音器界面，而不切换琶音器的开关状态

Sequencer 开关有自己的一组组合（切换演奏型、敲击设速、缩放、清除轨道），参见第 9 章。

13 第 13 章：LED 反馈

Erae 2 触控面板上覆盖着一个 42 × 24 的全彩 RGB LED 网格：共 1,008 个可独立寻址的像素，让您的 Layout 栩栩如生。每种元素类型都有自己的视觉语言，触摸交互会即时产生反馈，因此您随时都能确认面板正在响应。

13.1 RGB 配色系统

每个 LED 在每个通道（红、绿、蓝）上显示 8 位色彩。固件会应用逐通道的颜色校准，使您在 Erae Lab 中选择的颜色在面板上呈现得自然而一致，并对整体 LED 强度设置上限，以防止面板在长时间使用时过热。

当电源无法提供足够的电力时，Erae 2 会自动降低 LED 强度并显示 Low Power 消息（参见故障排除）。

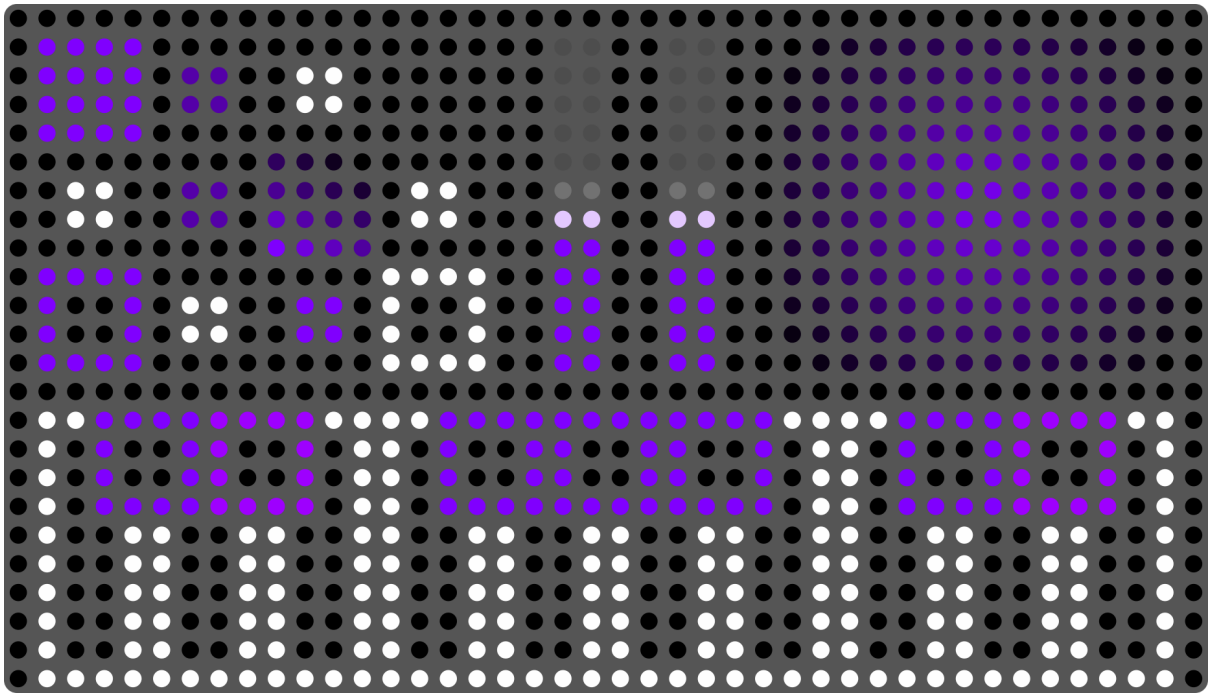
> **提示：**您在 Erae Lab 中设置的颜色，就是您在硬件上看到的颜色。校准已内置于固件中；您无需调整任何参数。

13.1.1 亮度控制

全局亮度通过设置菜单中的 **Brightness** 调节。在昏暗的录音棚中降低亮度会很有帮助，还能降低功耗。

13.2 各元素的可视化

每种元素类型都使用一种独特的视觉样式。样式在 Erae Lab 中分配，并保存在项目中。



LED 触摸反馈

13.2.1 琴键

琴键元素会显示覆盖其区域的纯色或渐变色。识别音阶的键盘会按音级为每个琴键着色，使根音和音阶内的音符在音阶外的音符中脱颖而出。

琴键的样式选项：- **Solid Color：**整个琴键区域采用均匀的填充 - **Linear Gradient：**颜色在琴键上从一种色调过渡到另一种色调 - **Circular Gradient：**颜色从琴键中心向外辐射 - **Rectangular Gradient：**颜色从中心向边缘过渡 - **Edge Color：**独特的边框色搭配对比强烈的填充

13.2.2 按钮

按钮使用 **Solid Color**、**Dual Intensity**、**Dual Color** 或 **Center Fill** 样式来显示其开和关状态。使用 Dual Intensity 时，关闭状态是该颜色的较暗版本；使用 Dual Color 时，关闭状态使用其自己的颜色。锁定式按钮会在两次按压之间保持其点亮状态。

13.2.3 推子（1D 与 2D）

推子元素会显示一条光带或一片活动光区，跟随您的手指位置移动。填充从推子的 **Center Value** 开始，并延伸至当前值。新建的推子使用 `0` 作为中心值，这与传统的从最小值开始渲染的方式相匹配；将中心值设为 63 则可创建一种带中心定位点的视觉响应。

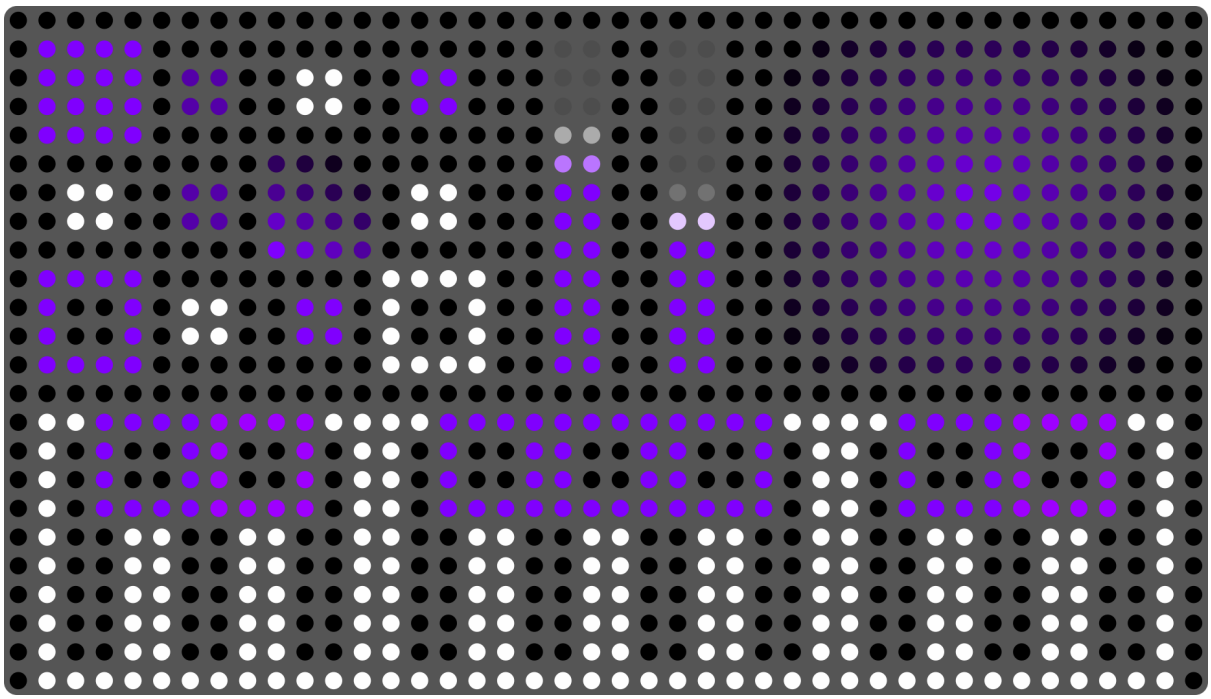
- **Fader 1D：**一条垂直或水平的光带从中心值朝您的触摸点填充。当中心值为 `0` 时，它从底部或最小值端开始填充。当中心值为 63 时，它从中点向任一方向填充。
- **Fader 2D：**一个十字准线或圆点标记出您在元素区域内的 X/Y 位置。活动光区绘制在中心点与当前位置之间，并且对于带中心定位点的 XY 布局，可保留一个暗淡的中心十字准线作为参考。

13.2.4 API 区域

API 区域没有内置的视觉样式：每个像素的颜色完全由您的软件通过开发者 API 驱动，从而实现完全自定义的可视化效果。

13.3 动画

动画在元素的基础样式之上叠加动态视觉效果。它们在 Erae Lab 中逐元素配置，并响应触摸事件进行播放。



LED 动画

可用的动画触发点有三个：

触发点	触发时机
Click	手指首次接触面板的瞬间立即触发
Slide	手指在面板上移动期间持续触发
Release	手指抬离时触发

每个动画都有一个**着色模式**（元素自身的颜色，或您设置的颜色）、一个**速度**和一个**颜色**。ColorMorph 还有一个**强度**。

13.3.1 FingerGlow

一团光晕跟随移动的手指在元素上扩散，留下一道短暂的、会自然消退的光迹。

> **提示：**一个使用对比色的快速点击 FingerGlow，能让弹奏琴键带来类似钢琴琴槌的效果，细微但在现场演奏中清晰可辨。

13.3.2 Ripple

一圈光环从触摸点向外扩散，并在远离接触点的过程中逐渐消退。Ripple 能为打击乐和节奏类元素带来一种真实的鼓垫触感。

13.3.3 Halo

一团光晕出现在接触点处，并在整个触摸期间停留在那里；它不会跟随手指拖曳。Halo 适用于持续延音的琴键，当您希望每个按住的音符下方都有一团持久的光晕，而非移动的效果时尤为合适。

13.3.4 ColorMorph

元素会向所配置的颜色过渡。**Intensity** 控制颜色变化的幅度；**Speed** 控制过渡所需的时间。可将 ColorMorph 分配给 Click 触发点、Release 触发点，或两者。

> **提示：**在推子元素上使用 ColorMorph，为 Click 设置一种暖色，为 Release 设置一种冷色，让活动的推子一目了然。

13.4 按压反馈

当您触摸键盘琴键时，LED 会立即以两种方式作出响应：

1. **强度变化：**琴键会根据其在 Erae Lab 中的强度变化设置变亮或变暗。正值会让琴键在手指下方变亮；负值则会让它变暗。
2. **动画触发：**如果该元素设有点击动画，它会在同一时刻触发。

> **提示：**按压反馈来自触摸检测，而非 MIDI 输出，因此即使该元素的所有 MIDI 输出都已关闭，您也能看到它。

13.5 状态指示

LCD 与开关 LED 承载着不会出现在主面板网格上的额外状态信息。

13.5.1 自动保存指示

每当 Erae 自动将您的项目保存到 SD card 时，LCD 左上角会淡入淡出一个带 SD card 图标的橙色小徽标。

13.5.2 项目已保存确认

当您使用 **Save Backup Project** 保存到备份槽位时，LCD 会显示 **Project Saved** 两秒钟。

13.5.3 时钟同步脉冲

当选择了外部 MIDI 时钟源（`USB-dev`、`MIDI` 或 `USB-host`）时，主界面的速度会显示传入的速度，并经过平均处理以平滑抖动。

> **提示：**如果速度显示快速波动，传入的 MIDI 时钟可能存在抖动。在排查问题时，可将 **Clock source** 切换为 `INT` 以稳定速度。

13.5.4 开关 LED

全部 18 个面板开关都有一个 RGB LED：

- **N1 到 N8：**当前 Layout 以全亮度白色点亮。其他 Layout 柔和发光，并在演奏音符时脉动变亮。Looper 被静音的 Layout 以红色点亮。
- **Play/Rec：**时钟运行时为绿色，录制时为红色，时钟停止但已预备录制时为脉动的红色。
- **Plus / Minus：**显示所选元素的八度。处于中间八度时两者都点亮；高于中间八度时，**Plus** 会呈现一种随八度变化的颜色，低于中间八度时则由 **Minus** 呈现。
- **Scale、Mapping、Sequencer、Settings：**显示其界面时会更亮。
- **Arp：**所选键盘的琶音器开启时以彩色点亮。

> **提示：**开关 LED 跟随全局亮度：在设置中降低 **Brightness** 会同时调暗面板和开关面板。

14 第 14 章：连接到 Erae Lab

Erae Lab 是配套的桌面应用程序，用于设计、编辑和管理您的 Erae 2 项目。将硬件连接到 Erae Lab 后，即可使用 Layout 编辑、固件更新以及后台项目同步等功能。

14.1 USB 连接

使用 USB 线缆将 Erae 2 连接到电脑。Erae Lab 不使用 MIDI 端口，因此使用它无需配置任何 MIDI 端口，您的 DAW 也可以同时继续使用这些端口。

在 macOS 和 Windows 上无需安装驱动程序。在 Linux 上，Erae Lab 首次需要 USB 权限时会显示 **Fix Now** 通知；点击它并输入您的管理员密码即可。

14.1.1 自动检测

Erae Lab 每隔几秒会查找一次已连接的设备，因此您可以在启动 Erae Lab 之前或之后插入 Erae 2。找到设备后，Erae Lab 会读取其固件版本和状态，然后将设备上存储的项目与 Erae Lab 资料库进行比对。在连接正常的情况下，这只需片刻。

> **提示：**如果 Erae Lab 无法连接到设备，它会显示一条带 **Rescan** 按钮的消息。请检查线缆；如果消息提示设备无响应，请将 Erae 2 断电后重新开机，然后按 **Rescan**。

14.2 连接指示

- **Erae Lab：**设备已连接时，顶部栏右侧的 Erae 标志会以彩色点亮；未连接时则显示为灰色。
- **Erae Lab 通知：**同步、固件更新和文件传输期间会出现进度消息。
- **Erae 2 LCD：**每当项目更改被保存到卡上时，左上角的橙色 SD 徽标会淡入显示。

> **提示：**请等待 Erae Lab 的进度消息结束后再断开连接，尤其是在固件更新期间。

14.3 项目同步

Erae Lab 与 Erae 2 在后台保持同步。在 Erae Lab 中进行的 Layout 编辑会自动保存，并在您停止编辑后不久发送到设备。您在设备上所做的更改（例如在 Scale 或 Mapping 界面上）会保存到 SD card，并回报给 Erae Lab。您无需在每次编辑后手动执行推送或拉取操作。

尚无 SD card 条目的项目（工厂项目，或在没有 SD card 时加载的备份）在您于设备上使用 **Save** 或 **Save As** 保存之前不会被同步。

> **注意：** Brightness、Sensitivity、Velocity Curve 和 CV Gain 是存储在 Erae 2 本身上的设备级设置。它们不属于任何项目，也不会被同步。

14.3.1 冲突解决

如果自上次同步以来，某个项目在 Erae Lab 和设备上都被更改过，Erae Lab 会询问保留哪个版本：

- **Keep Device：**用 Erae 上当前的版本替换 Erae Lab 中的项目。
- **Keep Both：**重命名 Erae 上的版本，将其作为单独的项目导入 Erae Lab，然后将两个项目都同步回 Erae。
- **Keep Lab：**将 Erae Lab 的版本发送到 Erae，并丢弃设备上有冲突的更改。关闭该消息也等同于选择 Keep Lab。

> **提示：**为避免冲突，请在开始使用 Erae Lab 编辑之前先连接好您的 Erae 2。

14.4 通过 Erae Lab 进行固件更新

固件更新通过 **Erae Lab** 提供，Erae Lab 内置了匹配的固件。它既可以更新正在运行的设备，也可以恢复已处于引导加载（bootloader）模式的设备。

更新方法：

1. 将 Erae 2 连接到外部电源，然后连接到您的电脑，并打开 Erae Lab。当设备报告电力不足时，Erae Lab 会拒绝开始更新。
2. 有更新的固件可用时，Erae Lab 会显示 **Firmware update ready**。点击 **Start Update**。
3. 设备会重启进入其引导加载程序，Erae Lab 随即传输固件。请勿断开 USB 线缆。
4. 设备会重启进入新固件，然后用大约一分钟完成安装。在此期间请勿关闭电源。

如果设备固件对于您的 Erae Lab 版本来说过旧，Erae Lab 会告知您，并要求您先更新才能连接。

如果 Erae Lab 发现某台设备正在其引导加载程序中等待，它会显示 **Bootloader detected**，并提供三个选项：**Install latest firmware**、**Select file**（您电脑上的固件 `.syx` 文件）或 **Dismiss**。

> **提示：**固件更新不会清除您的项目。项目存储在 SD card 上，与固件分开。

14.5 哪些功能需要 Erae Lab，哪些可独立使用

即使未连接 Erae Lab，Erae 2 也能完整使用。下表汇总了哪些功能需要 Erae Lab：

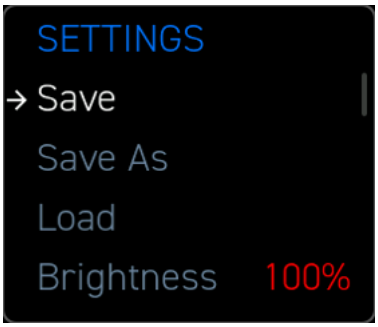
功能	独立使用	需要 Erae Lab
演奏 Layout	是	否
切换 Layout（N1 到 N8）	是	否
调整速度和时钟	是	否
Looper、琶音器和步进音序器	是	否
踏板输入和 CV 时钟配置	是（LCD 设置）	否
编辑元素的音阶、MIDI 和 CV 映射	是（LCD 的 Scale 和 Mapping 界面）	高级编辑需要
创建和编辑自定义音阶	是（LCD 的 Custom Scales）	导入 Scala 文件需要
移动、调整大小、设置样式或更改元素类型	否	是
创建新 Layout	否	是
固件更新	否	是
导出和导入项目文件	否	是

> **提示：**即使未连接 Erae Lab，您在设备上所做的更改也会自动保存到 SD card。

详情请参阅 Erae Lab 手册中的设备同步和固件更新章节。

15 第 15 章：设置

设置菜单是您的设备端控制中心，用于管理全局偏好、项目设置、保存与加载、MIDI 路由、踏板、CV 时钟以及校准。按 **Settings** 开关即可打开它；再次按下该开关或按 **Return**，即可返回主界面。完整的行列表见第 11 章。



设置

15.1 全局设置

全局设置存储在 Erae 2 本身上。它们适用于所有项目，不会与 Erae Lab 同步。

15.1.1 亮度

Brightness 同时控制触控面和开关面板的 LED 强度，范围为 5% 到 100%（默认 100%）。较低的数值能降低功耗，并且在昏暗环境下对眼睛更友好。

> **提示：**对于大多数现场演出环境而言，亮度设置在 60% 到 70% 左右会比较舒适。请将最高亮度留给装置展示或灯光充足的舞台。

15.1.2 灵敏度

Sensitivity 设定触控面对触摸的响应程度。共有四个预设选项可供选择：

选项	说明
`XSensitive`	响应最灵敏：适合录音室使用或触感非常轻柔的演奏者
`Sensitive`	响应灵敏，同时对误触有一定防护
`Safe`	默认值：在大多数演奏风格和环境下保持平衡
`XSafe`	响应最不灵敏：在高振动环境下减少误触发

> **提示：**如果触控面感觉反应迟钝或需要过大的压力，请切换到 `XSensitive`。如果在未触摸时出现幽灵音符，请切换到 `Safe` 或 `XSafe`。

15.1.3 力度曲线

按 **Velocity Curve** 打开力度曲线编辑器。它塑造整台设备上手指压力转化为音符力度的方式。共有四个参数依次塑造响应：

- **Threshold：**力度开始上升之前所需的接触量。低于该阈值不会产生任何力度；高于该阈值时，剩余的压力范围会被重新缩放。提高该值可减少轻触时的误触发。
- **Compand：**扩展或压缩范围中段，使力度动态更易或更难控制。50 为中性值（线性）。
- **Drive：**弯折曲线。高于 50 的数值会让曲线更早上升（较轻的触摸也能达到较高力度），低于 50 的数值会让曲线更晚上升（需要按得更用力才能得到同样的力度）。
- **Range：**曲线的整体灵敏度，而非力度的上限。数值越高，达到满力度所需的压力越小。它不会限制最大力度。

每个参数的取值范围为 0 到 95，以 5 为步进变化。出厂默认值为 Threshold 0、Drive 50、Compand 50、Range 70。

在您调节每个参数时，LCD 上会实时渲染曲线预览。更改会立即应用到您的演奏中，并在稍后自动保存。按重置按钮可将这四个数值恢复为出厂默认值。

力度曲线是存储在设备本身上的设备级设置，与元素一章中描述的逐元素力度参数（例如某个琴键的 Velocity Tune）是分开的：全局曲线塑造的是压力读数本身，而元素参数则进一步塑造最终发送的音符。

> **提示：**如果您的演奏感觉动态过强、音符总是处于最大力度，请降低 **Range** 并略微提高 **Threshold**。如果所有音符都显得太轻，请提高 **Range** 或 **Drive**。

15.2 项目设置

这些设置随当前项目一起保存。

15.2.1 速度

Tempo 设定 Looper、琶音器、音序器、节拍器和 CV 时钟使用的内部速度。范围：1 到 999 BPM。改动会立即生效。

15.2.2 时钟源

Clock source 用于选择时间基准：

取值	说明
`INT`	内部时钟：Erae 2 自行生成速度，并向 USB Device、USB Host、MIDI A 和 MIDI B 发送 MIDI 时钟、Start 和 Stop
`USB-dev`	跟随在 USB Device 端口（您的电脑）收到的 MIDI 时钟
`MIDI`	跟随在 TRS MIDI 输入收到的 MIDI 时钟
`USB-host`	跟随来自连接到 USB Host 端口的设备的 MIDI 时钟

使用外部时钟源时，Erae 2 还会跟随传入的 Start、Stop 和 Continue 消息。

> **提示：**当使用 DAW 作为时钟主控时，请将 **Clock source** 设为 `USB-dev`。这样 Erae 2 会将其 Looper、琶音器和音序器锁定到 DAW 的走带。

15.2.3 节拍器

开启 **Metronome** 后，在时钟运行期间，Erae 2 会用短暂的振动脉冲标记每一拍，并在每小节的第一拍使用较长的脉冲。节拍器不发送任何 MIDI。

15.2.4 切换 Layout 时发送 CC

开启 **CC on layout change**（默认开启）后，切换 Layout 时会在 MIDI 通道 16 上向 USB Device 端口发送 Control Change 消息：Layout N1 到 N8 对应 CC `102` 到 `109`，您进入的 Layout 值为 `127`，您离开的 Layout 值为 `0`。推子在其所在 Layout 被进入时也会重新发送其当前值。这使外部软件能够跟随 Layout 的切换。将其关闭可让 Layout 切换保持静默。

15.3 自定义音阶

按 **Custom Scales** 可列出存储在设备上的自定义音阶、新建音阶、编辑已有音阶，或通过 **Save Scales to Project** 和 **Load Scales from Project** 保存和加载音阶集。编辑器的说明见第 11 章。

项目的全局音阶不在此处设置：请在每个需要跟随全局音阶的键盘的音阶界面上开启 **GLOBAL**（参见第 11 章）。

15.4 MIDI 路由

按 **Routing** 打开路由界面。每个开关将传入的 MIDI 从一个端口转发到另一个端口：

路由	说明
MIDI -> USB Host	将 TRS MIDI 输入转发到 USB Host 端口
MIDI -> USB Device	将 TRS MIDI 输入转发到您的电脑
MIDI -> MIDI A	直通：TRS MIDI 输入到 MIDI Out A
MIDI -> MIDI B	直通：TRS MIDI 输入到 MIDI Out B
USB Device -> USB Host	将来自您电脑的 MIDI 转发到 USB Host 端口
USB Device -> MIDI A	将来自您电脑的 MIDI 转发到 MIDI Out A
USB Device -> MIDI B	将来自您电脑的 MIDI 转发到 MIDI Out B
USB Host -> USB Device	将来自 USB Host 设备的 MIDI 转发到您的电脑
USB Host -> MIDI A	将来自 USB Host 设备的 MIDI 转发到 MIDI Out A
USB Host -> MIDI B	将来自 USB Host 设备的 MIDI 转发到 MIDI Out B

> **提示：**要将 Erae 2 用作 MIDI 直通盒，请启用 **MIDI -> MIDI A**，并将上游控制器连接到 MIDI 输入、合成器连接到 MIDI Out A。

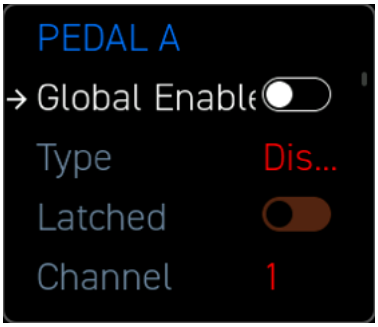
15.5 CV 时钟输出

CV Clock 在一个 CV 输出上发送锁定到项目速度的时钟脉冲。开启 **CV Clock**，选择 **Clock Output**（1 到 23），并设定 **Beat Division**（默认 4 ppqn）：

分频	说明
1 ppqn	每四分音符一个脉冲
2 ppqn	每四分音符两个脉冲
4 ppqn	每四分音符四个脉冲（十六分音符）
8 ppqn	每四分音符八个脉冲
24 ppqn	标准 MIDI 时钟速率
48 ppqn	双倍 MIDI 时钟速率

Reset Output 为只读：它始终是时钟输出之后紧接的输出，无法被禁用或单独分配。详情请参阅CV输出。

15.6 踏板输入配置



踏板输入

Erae 2 有两个踏板输入：**Pedal A** 和 **Pedal B**。在设置中按下对应的行即可打开其配置界面。每种类型显示的行列在第 11 章中。

15.6.1 类型

Type 用于选择踏板的行为：

类型	最适合
Disabled	忽略该输入
Switch	开/关式开关踏板：在按下和松开时发送 Note、CC、Program Change 或 Tap Tempo
Expressive	连续表情踏板，带 **Invert** 选项

类型	最适合
`Sustain`	延音踏板。**Mode** 为 `Binary` 时发送开值和关值；为 `Continuous` 时跟随踏板位置
`Kick`	力度感应的底鼓触发

15.6.2 全局启用

切换 **Global Enable** 可在不丢失踏板配置的情况下激活或旁通该踏板输入。

15.6.3 路由

Out Routing 会打开输出路由界面，您可以在其中选择由哪些 MIDI 输出接收踏板消息。

15.6.4 锁定

开启 **Latched** 后（适用于 Switch 和 Sustain Binary 类型），单次按下即可切换输出状态，而无需一直按住踏板。

15.6.5 通道

Channel 设定踏板消息所使用的 MIDI 通道（1 到 16）。

> **提示：**将 Pedal A 设为延音、Pedal B 设为表情，即可获得类似键盘的演奏配置。

15.7 校准流程

15.7.1 CV 校准

状态行显示 `CV calibrated` 或 `CV default calibration`。按 **Run CV calib** 可运行自动 CV 校准：Erae 2 会将其 24 路 CV 输出逐一与内部参考电压进行比对测量，并保存结果。如果在您的振荡器上八度跟踪略宽或略窄，可使用 **CV Gain (cent/oct)**（`-50` 到 `+50`）微调输出的音高缩放。

15.7.2 编码器校准

状态行显示 `Encoder calibrated` 或 `Encoder default calibration`。它仅供参考；无需用户操作。

15.7.3 踏板输入校准

对于 Expressive、Sustain Continuous 和 Kick 踏板，在踏板界面上按 **Calibrate** 即可启动校准向导。

对于 Expressive 和 Sustain Continuous 踏板：

1. **Welcome：**按 **Next**。
2. **Capture Min：**让踏板完全松开（脚跟位置），然后按 **Capture Min**。
3. **Capture Max：**将踏板完全踩下（脚尖位置），然后按 **Capture Max**。

对于 **Kick** 踏板：

1. **Capture：**让踏板保持静止，然后按 **Capture**。
2. **轻击：**轻轻敲击踏板至少三次，然后按 **Continue**。
3. **重踩：**用力踩踏踏板至少三次，然后按 **Continue**。

随后向导会保存校准结果。如果踏板范围过小，或重踩没有明显重于轻击，向导会报告失败并提供 **Retry**。

> **提示：**在首次演出之前，请先校准每个新踏板。各款踏板的行程范围各不相同，未经校准的踏板可能无法达到其完整范围。

15.8 项目管理

- **Save：**将当前项目写入其 SD card 条目。对于备份项目或工厂项目，会打开 Save As。
- **Save As：**以新名称保存，或覆盖所选项目。
- **Load：**从 SD card 加载项目。
- **Save Backup Project：**将当前项目写入内部备份槽位。
- **Load Backup Project：**加载备份槽位。挂载了 SD card 时，它还会以 `Backup`、`Backup_2` 等名称保存到卡上。
- **Load Factory Project：**将工厂项目作为临时项目加载，直到您保存它为止。
- **Format SD Card & Reboot：**擦除并重新格式化 SD card，然后重启。仅在需要从 SD 错误中恢复时使用。
- **Reset Demo：**恢复演示内容（仅限演示模式）。

只要项目有 SD card 条目，Erae 还会在每次更改后不久自动将您的更改保存到 SD card。工厂项目和备份项目在 **Save** 或 **Save As** 为它们创建 SD card 条目之前没有该条目，因此它们既不会被自动保存，也不会与 Erae Lab 同步。项目的重命名和删除在 Erae Lab 中完成。

15.8.1 MIDI 2.0

MIDI 2.0: ON/OFF 一行用于切换 USB MIDI 模式。它默认关闭（MIDI 1.0）。选择它会要求确认，然后 Erae 2 会以新模式重启。

15.9 存储状态与恢复

Erae 2 保存的项目通常存放在 SD card 上。内部闪存用于存放备份槽位、校准数据和设备设置。

- **已挂载 SD card：**可使用常规的项目保存、加载和 Erae Lab 同步。
- **无 SD card：**设备可以使用工厂项目或备份项目，但无法保存到 SD 项目库或从中加载。
- **无文件系统：**设备会在可以使用该卡之前询问是否进行格式化。
- **读写故障：**保存或加载可能失败；请格式化或更换 SD card。

项目写入具有断电保护：如果在写入过程中断电，设备会在下次启动时恢复上一份有效的项目数据。格式化 SD card 会擦除其上的项目并重启设备。

16 第 16 章：故障排除

本章介绍使用 Erae 2 时可能遇到的最常见问题及其解决方法。

16.1 连接问题

16.1.1 Erae Lab 无法识别 USB

现象： Erae Lab 显示没有连接的设备；顶部栏中的 Erae 标志保持灰色。

步骤：

1. 确认 USB 数据线支持数据传输（而不是仅供充电的线缆）。换一根线缆试试。
2. 确认线缆插在 Erae 2 上用于连接电脑的 USB 端口，而不是电源端口或用于 USB MIDI 设备的端口，并直接连接到电脑，而不要经过集线器。
3. 如果 Erae Lab 显示 **Rescan** 按钮，请按下它。如果其消息提示设备无响应，请先将 Erae 2 断电后重新开机。
4. 在 Linux 上，Erae Lab 需要 USB 权限。如果出现 **Fix Now** 通知，请点击它并输入您的管理员密码，然后重新连接设备。
5. 退出 Erae Lab，拔下设备，等待五秒后重新插上，然后重新启动 Erae Lab。

16.1.2 DAW 中不出现 MIDI 端口

现象： Erae 2 已连接，但其 MIDI 端口未出现在 DAW 的 MIDI 设备列表中。

步骤：

1. 确认设备已连接：其 LCD 已点亮，并且 Erae Lab（如已打开）显示它已连接。设备一连接，MIDI 端口就会出现；Erae Lab 无需运行。
2. 在 macOS 上，检查 **音频 MIDI 设置**（应用程序 -> 实用工具）：Erae 2 应出现在 MIDI 工作室窗口中。
3. 在 Windows 上，检查 **设备管理器** 中的 MIDI 设备列表，确认没有黄色警告图标。
4. 在 Erae 2 已连接的情况下重启您的 DAW。某些 DAW 仅在启动时扫描 MIDI 端口。
5. 检查设置中的 **MIDI 2.0: ON/OFF** 一行。它默认关闭；如果您开启了它，而您的 DAW 不支持 MIDI 2.0 设备，请将其关闭（Erae 2 会重启）。

> **提示：**在默认的 MIDI 1.0 模式下，Erae 2 会提供 `Erae 2 MIDI`（标准）和 `Erae 2 MIDI (MPE)`（富表现力演奏）两个端口。

16.1.3 macOS 上不出现 MPE 端口（设备条目过期）

现象：在固件更新或重新连接后，`Erae 2 MIDI (MPE)` 端口未出现在 macOS 上的 DAW 中，或出现时端口编号错误。

步骤：

1. 打开 **音频 MIDI 设置**（应用程序 -> 实用工具 -> 音频 MIDI 设置）。
2. 如果 MIDI 工作室窗口不可见，点击 **窗口 -> 显示 MIDI 工作室**。
3. 找到任何过期的 Erae 2 设备条目（它可能显示为灰色或显示旧的名称）。
4. 右键点击过期条目并选择 **移除设备**。
5. 断开并重新插上 USB 线缆。设备会重新枚举，预期的 MIDI 端口便会出现。
6. 重启您的 DAW 以获取刷新后的端口列表。

16.2 MIDI 路由问题

16.2.1 音符未在预期的输出上发声

步骤：

1. 在 Erae Lab 中，或通过设备 Mapping 界面上的 **Out Routing**，检查该元素的 MIDI 输出目标（USB Device、MIDI A、MIDI B、USB Host）。
2. 检查设置 -> Routing 中的路由开关是否造成了意外的回路。
3. 确认 MIDI 通道与您的合成器或 DAW 音轨相匹配。
4. 对于 MPE 乐器，确保在键盘元素和接收乐器上都启用了 MPE，并将您的 DAW 音轨指向 `Erae 2 MIDI (MPE)` 端口。

16.2.2 意外的音符或重复触发

现象：音符发声两次，或未触碰表面却出现音符。

步骤：

1. 检查 **Routing** 中是否启用了直通路由（例如 **MIDI -> USB Device**）。直通路由与 DAW 回声设置叠加可能导致音符重复。
2. 如果出现幽灵触发，请检查设置中的 **Sensitivity**。切换到 `Safe` 或 `XSafe` 以减少误触发。

16.3 触摸反应迟钝

现象：需要重压才能触发音符，或力度值始终偏低。

步骤：

1. 在设置中，将 **Sensitivity** 设为 `XSensitive` 以获得最高的表面灵敏度。
2. 打开 **Velocity Curve** 并提高 **Range**（数值越高，达到满力度所需的压力越小）；如果音符完全需要太大压力才能触发，请降低 **Threshold**。
3. 检查表面是否有碎屑或冷凝水。用干燥的无绒布清洁。
4. 演奏前让设备运行几分钟：传感器读数会随设备升温而略有稳定。

16.4 电源问题

16.4.1 Low Power 消息

现象： LCD 显示 `Low Power (Low brightness, CV, MIDI B, Metronome disabled)`。

电源无法为完整运行提供足够的电力，因此 Erae 2 会降低 LED 亮度，并关闭 CV 输出、MIDI Out B 和节拍器。请将 Erae 2 连接到能够提供更大功率的外部电源，然后重新启动它。当设备报告电力不足时，Erae Lab 也会拒绝更新固件。

16.5 LED 无响应

16.5.1 表面 LED 不亮或部分点亮

步骤：

1. 检查设置中的 **Brightness**；它可能被设为了较低的数值（最小值为 5%）。
2. 检查是否有 Low Power 消息：电力有限时 LED 会变暗。
3. 确认当前 Layout 中含有指定了非黑色颜色的元素。空 Layout 会让表面显示为暗色。
4. 如果网格的某个区域保持黑暗，而周围区域亮着，请给设备断电重启（断开电源，等待五秒，再重新连接）。如果问题仍然存在，请联系支持。

16.6 出厂、备份和重置流程

Load Factory Project 会将工厂项目作为临时项目加载。SD card 上的项目不会被清除，且工厂项目在您保存它之前不会成为 SD 库的一部分。

1. 确认您刚才演奏的项目是安全的：如果它的名称显示在设置中 **Save** 的旁边，说明它已经保存在 SD card 上。
2. 在设置中按下 **Load Factory Project**。它会立即加载，不会要求确认。

Save Backup Project 会将当前项目写入内部备份槽位。**Load Backup Project** 会恢复该备份；挂载了 SD card 时，它还会以 `Backup`、`Backup_2` 等名称保存。

Reset Demo 会在演示模式激活时恢复演示内容。仅在您有意要擦除 SD card 并从存储问题中恢复时，才使用 **Format SD Card & Reboot**。

16.7 引导加载程序恢复

如果设备在更新中断后无法正常启动，请使用引导加载程序恢复来重新安装固件：

1. 将设备连接到外部电源和电脑，然后打开 Erae Lab。
2. 当 Erae Lab 显示 **Bootloader detected** 时，选择 **Install latest firmware**，或选择 **Select file** 以使用保存在您电脑上的固件 `.syx` 文件。
3. 传输过程中请勿断开 USB。
4. 设备会重启进入新固件，并用大约一分钟完成安装。在此期间请勿关闭电源。

如果更新反复失败，Erae Lab 会要求您拔下设备、重新插上后再试一次。如果问题仍然存在，请联系支持。

> **提示：**在电脑上保存一份最新固件文件的副本，以便在需要时使用 **Select file**。

16.8 孤立引导加载程序检测

如果在更新过程中断电，设备可能会重启进入其引导加载程序并停留在其中。Erae Lab 会自动检测到这一情况，并显示 **Bootloader detected**，提供 **Install latest firmware**、**Select file** 和 **Dismiss** 三个选项。每台设备都有恢复用的引导加载程序，因此上述流程可以将其恢复。

> **提示：**请使用优质的 USB 线缆，让设备保持外部供电，并且不要在固件更新期间或更新后的一分钟内关闭设备电源。

16.9 SD card 错误

FAT32 是 SD card 推荐的文件系统。如果重新插入后错误仍然存在，请使用设置中的 **Format SD Card & Reboot** 重新格式化。如果问题仍未解决，请尝试更换一张 microSD card。

已保存的项目通常存放在 SD card 上。在没有 SD card 的情况下，设备可以使用工厂项目或备份项目，但无法保存到项目库或从中加载，并且这些项目在被保存到卡上之前不会与 Erae Lab 同步。

LCD 可能会显示以下某条消息：

消息	含义	处理方法
`Please insert microSD card and reboot`	启动时没有可用的 SD card	插入一张 microSD card（FAT32，最大 32 GB）并重新启动
`No filesystem found Format SD card?`	该卡没有可读的文件系统	选择 **Yes** 进行格式化（Erae 会重启），或在电脑上格式化
`SD Read/Write error`	SD card 出现故障	尝试 **Format SD Card & Reboot**；如果仍然存在，请更换该卡
`Project corrupted (Loaded factory default)`	某个项目无法读取	加载另一个项目，或执行 **Load Backup Project**
`Project read failed (File may be corrupted)`	某个项目文件无法读取	加载另一个项目，或执行 **Load Backup Project**
`Project save failed (Please try again)`	保存未能完成	再次保存；如果仍然存在，请检查或更换该卡
`Project file not found`	卡上缺少该项目	加载另一个项目
`Legacy project Please connect ERAE Lab`	发现了来自旧版固件的项目	连接 Erae Lab；它会转换这些项目

> **提示：**如果重新插入后 SD card 仍反复无法被检测到，请给设备断电重启。

16.10 联系支持

如果您已按上述步骤操作但问题仍然存在，请联系 Embodme 支持：

邮箱： support@embodme.com

请附上您的固件版本（开机时显示在 LCD 上）、问题描述以及您已经尝试过的步骤。

17 第 17 章：Erae 兼容性

本章是为初代 Erae（即 Erae Touch）用户准备的参考资料。如果您使用的是 Erae 2，本章的大部分内容并不适用于您：请参阅本章末尾的表格，了解两者的并排对比。Erae Touch 还有自己专属的说明书（Erae Beta User Manual），其中包含关于其按键、音序器和力度曲线的完整章节。

17.1 什么是 Erae Touch?

Erae Touch 是第一代硬件。它与 Erae 2 运行相同的核心触控表面和 MIDI 引擎，但在若干重要方面存在差异：它没有 LCD 屏幕、没有编码器、没有 CV 输出、没有踏板输入，也没有 TRS MIDI 输入，并且通过圆形插孔（barrel jack）上的专用电源适配器供电，而不是通过 USB 供电。导航完全通过五个专用的物理按键和 LED 表面来完成。在其硬件条件的限制内，Erae Touch 固件获得了与 Erae 2 相同的 Layout、音阶、音序器和 MIDI 功能。

17.2 项目与 Layout

Erae Touch 存储**四个项目**，命名为 A 到 D，每个项目各持有**八个 Layout**。按下 **Home** 会在 LED 表面上打开 Layout 选择器：一行 A 到 D 的项目单元格，下方是八个编号的 Layout 单元格。所有编辑都会在您最后一次更改约两秒后自动保存到当前项目，保存期间 **Alt** LED 会以橙色脉动；设备上没有手动保存步骤。

Erae Lab 会让这四个项目与其自身的副本保持同步，并将它们固定显示在项目列表的顶部。

> **提示：**在进行大幅 Layout 更改之前，请使用 Erae Lab 备份您的项目。

17.2.1 Main/Alt Layout 配对

Alt 按键将八个 Layout 两两配对：Layout 1 与 5、2 与 6、3 与 7、4 与 8。点按 **Alt** 会让表面在当前配对的两个 Layout 之间切换，因此您可以，例如，把一个半音键盘放在 Layout 1、一个鼓垫放在 Layout 5，然后只需按一次即可在两者之间切换。

17.3 五按键导航系统

Erae Touch 有五个专用的物理按键：**Home**、**Alt**、**Scale**、**Plus** 和 **Minus**。没有触摸屏、没有编码器，也没有软按键。

- **Home**：点按可打开或关闭 Layout 选择器。长按进入睡眠模式。
- **Alt**：点按可在当前配对的 Main 和 Alt Layout 之间切换，或关闭任何已打开的叠加视图。长按，选择一个键盘，然后松开，即可在该键盘上打开**步进音序器**（见下文）。
- **Scale**：点按可为焦点键盘打开音阶选择器。长按可为焦点元素打开 CC 映射。
- **Plus / Minus**：对焦点键盘进行八度升降。
- **Alt + Scale**：打开设备端的 **Velocity Curve** 编辑器（见下文）。

17.4 Erae Touch 上的步进音序器

Erae Touch 固件包含与 Erae 2 相同的音序器引擎，完全呈现在 LED 表面上：在带有键盘元素的 Layout 上按住 **Alt**，然后松开即可进入；点按 **Alt** 即可离开。表面会分为一条可演奏的 Live 条、一个 32 列的钢琴卷帘（piano roll），以及一条 Launcher 条，其上的垫块用于选择演奏型并控制走带、撤销/清除，以及一个表面端的设置页面（演奏方式、时钟源、量化、速度和长度）。

与 Erae 2 相比，Erae Touch 的音序器：

- 仅音序化**键盘和独奏琴键元素**（没有推子或按钮）
- 每个 Layout 提供**四个演奏型槽位**，而不是八个
- 只录制**音符和力度**：逐音符的表现力曲线无法在设备上录制或编辑，不过包含这些曲线的演奏型仍能正确回放
- 演奏型长度最多可设为 **64 步**（Erae 2：256 步）
- **没有 Looper 并轨**

Erae Beta User Manual 对此有完整介绍。

17.5 Erae Touch 上的力度曲线

Erae Touch 拥有与 Erae 2 相同的全局力度曲线（Threshold、Drive、Compand、Range，计算方式完全一致），并在**设备上**进行编辑：在 Layout 视图中按下 **Alt + Scale** 打开 LED 编辑器，点按某个参数单元格的左半或右半以更改其数值，并在表面右侧观看实时曲线图。该设置存储在设备的全局设置中；Erae Lab 不会修改它。两款设备的传感器调校方式不同，因此相同的数值在 Erae Touch 和 Erae 2 上手感可能不同。

17.6 仅 Z 轴压力感应

Erae Touch 传感器仅读取 **Z 轴**：即每个单元格的压力，而没有像 Erae 2 那样的逐单元格 X 或 Y 读数。

手指位置（表面上的 X 和 Y）是根据压力在相邻单元格上的分布计算得出的。这种方式可为手指追踪提供精确的 X 和 Y 坐标，但 **XY 分辨率取决于压力在多个单元格上的分布模式**。

实际影响：

- **力度（Velocity）**来自手指在接触最初瞬间落下的力度，提供与 Erae 2 相当的自然力度响应。
- **触后（Aftertouch）**（通道压力或复音压力）按预期工作，功能完整。
- **Pitch Bend 和滑动** 通过追踪根据压力分布计算出的 X 和 Y 位置来实现。响应是精确的，但在非常轻的触碰下可能感觉与 Erae 2 略有不同，此时压力分布在更少的单元格上，会降低位置精度。

> **提示：**为获得最准确的滑动和 Pitch Bend 响应，请使用自信、全接触的手指按压。非常轻或仅用指尖的触碰可能会降低 X 和 Y 追踪精度。

17.7 无 LCD

Erae Touch 没有 LCD 屏幕。状态信息通过 LED 传达：保存更改期间 **Alt** LED 会以橙色脉动，**Plus** 和 **Minus** LED 显示焦点键盘的八度，出现存储问题时表面会显示带边框的消息。

设备端的编辑画面（Layout 选择器、音序器及其设置页面、力度曲线编辑器、音阶选择器、CC 映射）都直接绘制在 LED 表面上。

> **提示：**如果您需要确认固件版本或检查设备状态，请连接到 Erae Lab。

17.8 无 CV 输出

Erae Touch 没有 CV 输出。Erae 2 的 24 个 CV 通道、音高和门信号路由以及 CV 校准系统在此完全不存在。如果您的设置需要 CV 和门信号输出，则必须使用 Erae 2。

Erae Touch 的 MIDI 输出可在以下接口上使用：

- **USB Device (Main)：**标准 MIDI
- **USB Device (MPE)：** MPE MIDI
- **TRS MIDI out (port A)：**硬件 DIN 风格输出

MIDI 输入仅在 USB Device (Main) 上可用。Erae Touch 没有 TRS MIDI 输入；来自外部设备的时钟和控制信号必须通过 USB 传入。

17.9 无 Looper，也无琶音器画面

Erae 2 的 Looper 在 Erae Touch 上并未作为面向用户的功能开放，其琶音器设置画面在 Erae Touch 仅有 LED 的界面上也没有等价物。上文所述的音序器就是 Erae Touch 上的演奏型工具。

17.10 无编码器

Erae Touch 没有旋转编码器。在 Erae 2 上使用编码器的功能，例如导航 LCD 菜单和微调数值，在 Erae Touch 上由 LED 表面上的触控操作以及 Plus 和 Minus 按键来替代。

17.11 两阶段固件更新

Erae Touch 采用两阶段固件更新流程。这是与 Erae 2 的一个关键区别，后者使用单阶段更新。

更新流程：

1. Erae Lab 将设备重启进入其更新模式。
2. Lab 将一个小型更新程序加载到设备上。该程序等待 Lab 期间，表面会显示 USB。
3. Lab 将新固件发送给更新程序，更新程序在接收和写入期间会显示一个带百分比的进度条。
4. 设备重启进入新固件，并重新连接到 Lab。

整个更新可能需要几分钟。如果出现问题，表面会在一个闪烁的边框中显示错误代码（E 后接两位数字）。

> **提示：**固件更新过程中请勿断开 USB 线缆或关闭设备电源。如果更新被中断，请将设备重新连接到 Lab：当设备以更新模式重新出现时，Lab 会继续传输；当设备以其恢复用引导加载程序重新出现时，Lab 会提供 **Install latest firmware**。

17.12 功能对比：Erae Touch 与 Erae 2

功能	Erae Touch	Erae 2
触控感应	仅 Z（X 和 Y 为计算值）	XYZ
项目 / Layout	4 个项目，各 8 个 Layout（Alt 配对）	每个项目 8 个 Layout
LCD	无	有

功能	Erae Touch	Erae 2
步进音序器	有：仅键盘、4 个演奏型、音符与力度、最多 64 步	有：键盘、推子、按钮；8 个演奏型；表现力
力度曲线	有，在 LED 表面上编辑（Alt + Scale）	有，在 LCD 上编辑（Settings）
Looper	无	有
CV 输出	无	24 通道
编码器	无	有
按键	5 个专用	18 个面板开关外加 Calibrate
踏板输入	无	2 个
TRS MIDI 输入	无	有
供电	专用适配器（圆形插孔）	USB-C Power 端口
固件更新	两阶段	单阶段
项目存储	内部存储器	SD card

18 附录 A：参数参考

本附录按元素类别列出每种元素类型的可配置参数，并使用 Erae Lab 和设备上所用的名称。数值按您在 Erae Lab 中看到的形式显示：MIDI 通道为 1 到 16，CC 值为 `0` 到 127，音符名称以 MIDI 音符 60 为 `C3`。

18.1 通用参数

这些参数出现在每个生成 MIDI 的元素上。

参数	说明	范围	默认值
Channel	发出消息所用的 MIDI 通道	1 到 16	1
Midi Output	由哪些物理输出承载该元素的消息	USB Device、USB Host、MIDI A、MIDI B（任意组合）	仅 USB Device
Trig on entry	切换到该元素所在 Layout 时，该元素发送哪些值	`Off` / `Default` / `All`	`Default`

18.2 控制器槽位

每个 CC 槽位（Pressure、X abs、Y abs、X rel、Y rel、Motion、Sound Brightness）共享以下设置。

参数	说明	范围
Enable	发送该控制器	on / off
Type	控制器类型	`CC` / `CC 14-bit` / `RPN` / `NRPN`

参数	说明	范围
Controller	控制器编号（CC 14-bit：0 到 31，LSB 位于控制器编号加 32）	0 到 127
Bank / Index	参数地址（RPN 和 NRPN）	各为 0 到 127
Min/Max	输出范围	0 到 127
Intensity	灵敏度缩放	0 到 255
Smoothing	低通平滑	0 到 255
Tracking	多指跟踪	None / LastPlayed / Highest /
Filter	响应曲线	Exponential / Linear
Initial value	起始值（相对槽位和 Sound Brightness）	0 到 127

18.3 Key（琴键）

琴键是最主要的表现力构建模块。一个 Key 元素将一个触摸区域映射到单个音符，并提供完整的逐指表现力。键盘的所有琴键使用相同的表现力设置。

18.3.1 基本参数

参数	说明	范围	默认值
Note	MIDI 音符	C-2 到 G8	C3
PB Range	以半音为单位的 pitch bend 范围	1 到 96	12（MPE 时为 48）

音符力度遵循设置中全设备通用的 **Velocity Curve**。

18.3.2 Pressure Tune（压力调校）

控制手指压力如何生成触后（通道或复音）。

参数	说明	范围	默认值
Enable	发送压力	on / off	on
Type	Channel（通道压力）或 Poly（复音触后）	枚举	Channel
Tracking	多指跟踪	None / LastPlayed / Highest / Lowest	None

参数	说明	范围	默认值
Filter	响应曲线	`Exponential` / `Linear`	`Exponenti`
Min/Max	输出范围	`0` 到 `127`	`0` / `127`
Intensity	压力灵敏度	`0` 到 `255`	`127`
Smoothing	低通平滑	`0` 到 `255`	`0`

18.3.3 Vibrato Tune（揉弦调校，Pitch Bend）

控制映射到 pitch bend 的手指水平移动。

参数	说明	范围	默认值
Enable	发送 pitch bend	on / off	on
Style	响应曲线	`Linear` / `Quadratic` / `FarQuadratic`	`Linear`
Strength	pitch bend 灵敏度	滑块	
Return Speed	bend 回归中心的速度	`None` 到 `Max`	

18.3.4 CC 分配（可选）

每个 CC 槽位都可以独立启用（参见控制器槽位）。

槽位	说明	默认值
Pressure	以 CC 形式输出的压力（在触后之外额外提供）	禁用
X abs	绝对水平位置	禁用
Y abs	绝对垂直位置	禁用
X rel	相对于手指落点的水平移动	禁用
Y rel	相对于手指落点的垂直移动	禁用
Motion	手指移动速度	禁用

运动速度在映射到控制器范围之前，会经过平滑并从 `0` 归一化到 `100 cm/s`。

> **提示：** MPE 激活时，pitch bend 和压力会自动通过逐音符通道传输。CC 槽位仍可用于额外的表现力维度。

18.3.5 Control Voltage（控制电压）

参数	说明	范围	默认值
Enable	发送 CV	on / off	off
Output	使用的第一个 CV 输出	1 到 24	
Gate / 1V / Oct	门信号和音高输出	on / off	on
Gate Mode	`Gate`、`Trig` 或 `Retrig`	枚举	`Gate`
Trig Duration (ms)	Trig 模式下的脉冲长度	25 到 250	25
Gate Delay (ms)	门信号打开前的延迟	0 到 5	0
CV Velocity / CV Pressure / CV X abs / CV Y abs	带电压范围的连续输出	-5 V 到 +8 V	off，0 V 到 5 V
CV X rel / CV Y rel	带范围和初始值的相对输出	-5 V 到 +8 V	off，初始值 2.5 V

输出顺序请参阅CV 输出。

18.4 Button（按钮）

按钮在按下时发送一条离散消息，并根据类型和 Latched 选项，在松开时或下次按下时发送第二条消息。

18.4.1 通用按钮参数

参数	说明	范围	默认值
Channel	MIDI 通道	1 到 16	1
Latched	切换模式：按下发送“开”消息，下次按下发送“关”消息	on / off	off
Type	`Note`、`Control Change`、`Program Change`、`Control Voltage`、`Tap Tempo`	枚举	`Note`

18.4.2 Button：Note 类型

参数	说明	范围	默认值
Note	MIDI 音符（以固定的中等力度发送）	C-2 到 G8	C2
CV	可选的门信号和 1 V/oct 输出	输出或禁用	禁用

18.4.3 Button：Control Change 类型

参数	说明	范围	默认值
Enable On / 控制器	按下时发送的控制器	控制器槽位	CC 63，禁用
Value（开）	按下时发送的值	0 到 127	127
Enable Off / 控制器	松开时或下次按下时发送的控制器	控制器槽位	CC 63，禁用
Value（关）	松开时发送的值	0 到 127	0

18.4.4 Button：Program Change 类型

参数	说明	范围	默认值
Bank（开）	随“开”消息发送库选择	on / off	
Bank MSB / **Bank LSB**（开）	Bank Select MSB（CC 0）和 LSB（CC 32）	0 到 127	
Program（开）	随“开”消息发送的 Program Change	on / off，0 到 127	
Bank / **Bank MSB** / **Bank LSB**（关）	“关”消息的库选择	on / off，0 到 127	
Program（关）	随“关”消息发送的 Program Change	on / off，0 到 127	

18.4.5 Button：Control Voltage 类型

参数	说明	范围	默认值
Output	CV 输出	1 到 24	
On/Off	开电压和关电压	-5 V 到 +8 V	5 V / 0 V

18.4.6 Button：Tap Tempo 类型

没有参数：每次按下都会敲击项目速度（仅限内部时钟）。

18.5 Fader 1D（Fader 1D）

一维推子，跟踪手指在其区域内的 Y 位置。

参数	说明	范围	默认值
Channel	MIDI 通道	1 到 16	1
Y abs CC	垂直位置的控制器	控制器槽位	该通道上第一个未使用的 CC
Default Y	Layout 加载后、任何触摸之前发送的值	0 到 127	63
Center Y	推子填充的视觉零点	0 到 127	0
Pressure CC	压力的可选控制器	控制器槽位	禁用
Motion CC	手指移动速度的可选控制器	控制器槽位	禁用
Pressure / **Y abs** CV	可选的 CV 输出	输出、电压范围	禁用

> **提示：**将 **Center Y** 设为 63，即可获得一个从中点向两侧填充的双极性推子。

18.6 Fader 2D（Fader 2D）

二维 XY 触控板，同时跟踪两个轴。

参数	说明	范围	默认值
Channel	MIDI 通道	1 到 16	1
X abs CC	水平位置的控制器	控制器槽位	该通道上第一个未使用的 CC
Y abs CC	垂直位置的控制器	控制器槽位	该通道上第一个未使用的 CC
Default X	加载时的水平值	0 到 127	63
Default Y	加载时的垂直值	0 到 127	63
Center X	X 渲染的视觉零点	0 到 127	0
Center Y	Y 渲染的视觉零点	0 到 127	0
Pressure CC	压力的可选控制器	控制器槽位	禁用
Motion CC	手指移动速度的可选控制器	控制器槽位	禁用
Pressure / **X abs** / **Y abs** CV	可选的 CV 输出	输出、电压范围	禁用

18.7 Keyboard（键盘：Keyboard、Keygrid、Drumpad）

键盘元素跨越多个琴键，并共享一组 Layout 级别的属性。每个琴键都使用上文 Key 中所述的表现力设置。

18.7.1 键盘布局参数

参数	说明	范围	默认值
Root note / Octave	基准音（Keyboard：仅 Octave）	`C-2` 到 `G8`	Keygrid 和 Keyboard 为 `C2`，Drumpad 为 `C1`
Scale	应用于键盘的音阶	Keyboard：15 种工厂音阶；Keygrid：14 种工厂音阶，或开启 MPE 时的自定义音阶	Keyboard 为 `Chromatic`，Keygrid 为 `Major`
Use global scale	跟随项目全局音阶（Keygrid、Drumpad）	on / off	off
Show Offscale	显示每个半音（Keygrid）	on / off	off
Semitones	开启 Show Offscale 时以半音计的行跳跃（Keygrid、Drumpad）	1 到 48	Keygrid 为 5，Drumpad 为 4
Degrees	关闭 Show Offscale 时以音级计的行跳跃（Keygrid）	1 到 48	3
Line offset	行跳跃（Keyboard）	1 到 48	
Key Width / Key Height	每个琴键的尺寸（以网格单元计）	单元数	4 / 4
MPE	启用 MIDI Polyphonic Expression	on / off	off
MPE Master	MPE 区域：`Channel 1`（Lower Zone）或 `Channel 16`（Upper Zone）	枚举	`Channel 1`

参数	说明	范围	默认值
Nb channel	MPE 成员通道数量（开启 MPE 时取代 Channel）	1 到 15	
Num voice	CV 声部数量	1 到 15	1

18.7.2 Glissando Tune（滑音调校）

控制手指在琴键间滑动时的音高行为。

参数	说明	范围	默认值
Enable	启用滑音	on / off	on
Vertical Gliss.	垂直移动参与滑音	on / off	on
Retrigger	越过琴键边界时重新触发音符	on / off	off
Tune Loc	参考点：`Pad`（琴键中心）或 `Finger`（初始触摸位置）	枚举	`Pad`
Smoothing	滑音平滑	`0` 到 `255`	`63`
In Tune Width	准音区域的大小，以琴键宽度的百分比表示	`0` 到 `100`	`50`

18.7.3 Sound Brightness（CC 74）

将琴键内的某个手势轴映射到 CC 74（Brightness / Timbre）。

参数	说明	范围	默认值
Gesture	驱动该 CC 的轴：`None`、`Pressure`、`X Absolute`、`Y Absolute`、`X Relative`、`Y Relative`、`Motion Speed`	枚举	`None`
Initial value	任何触摸之前发送的值	`0` 到 `127`	`64`
其他设置	**Min/Max**、**Intensity**、**Smoothing**、**Tracking**、**Filter**	参见控制器槽位	

18.7.4 Arpeggiator（琶音器）

琶音器在设备上设置（**Arp** 按键）；参见琶音器与 Looper。

参数	说明	范围	默认值
On / off	通过 **Arp** 按键切换	on / off	off
Quantize	在时钟网格上开始琶音	on / off	on
Octave	额外八度的数量	0 到 8	0
Rate	步进速率，以拍的分数表示：1/32、1/16、1/8、1/4、1/2、1/1、Pressure	枚举	1/4
Style	Up、Down、UpDown、UpAndDown、Random	枚举	Up
Pressure	手指压力在多大程度上塑造音符力度	0% 到 100%	40%

18.8 Ableton Launchpad（LivePad）

参数	说明	范围	默认值
Position	Full Width、Left Half 或 Right Half	枚举	由放置位置决定
Zoom	网格单元大小	Small / Large	Large

18.9 API Zone（API 区域）

API Zone 通过 SysEx 流式传输向外部应用程序公开手指数据。

参数	说明	范围	默认值
Zone Index	该区域在数据流中的标识符	0 到 127	0
Max Num Fingers	已保存，尚未生效	0 到 10	10
Finger Data Rate	已保存，尚未生效	0 到 255	0

18.10 Non-Visible Elements（不可见元素）

不可见元素不会显示在表面上，但会响应外部输入。

18.10.1 Pedal（踏板）

每个 Layout 都会配置两个踏板输入：**Pedal A** 和 **Pedal B**。

参数	说明	范围	默认值
Type	`Disable`、`Switch`、`Kick`、`Sustain (Binary)`、`Sustain (Continuous)`、`Expressive`	枚举	`Disable`
Channel	MIDI 通道	1 到 16	1
Midi Output	输出端口	USB Device、USB Host、MIDI A、MIDI B	USB Device

18.10.1.1 Pedal：Switch

参数	说明	范围	默认值
Latched	每次按下都切换（而非按下和松开）	on / off	off
Message Type	`Note`、`Control Change`、`Program Change`、`Tap Tempo`	枚举	`Note`
Note	Note 类型所用的音符	`C-2` 到 `G8`	`C4`

Control Change 和 Program Change 的设置与 Button 元素相同。选择 **Tap Tempo** 时，每次按下都会敲击项目速度。

18.10.1.2 Pedal：Sustain (Binary)

参数	说明	范围	默认值
Controller	延音所用的控制器	控制器槽位	CC 64
On Value	踏板按下时的值	`0` 到 `127`	`127`
Off Value	踏板松开时的值	`0` 到 `127`	`0`

参数	说明	范围	默认值
Latched	每次按下都切换	on / off	off

18.10.1.3 Pedal：Continuous（Expressive / Sustain Continuous）

参数	说明	范围	默认值
Controller	连续输出所用的控制器	控制器槽位	Expressive 为 CC 11，Sustain (Continuous) 为 CC 64
Invert	反转踏板方向	on / off	off

18.10.1.4 Pedal：Kick

参数	说明	范围	默认值
Note	受击时触发的 MIDI 音符	`C-2` 到 `G8`	`C0`
Velocity Sensitivity %	力度响应缩放	`0` 到 `100`	`100`
Impact Threshold %	触发所需的最小敲击力	`1` 到 `100`	`10`
Duration Mode	`Until Release` 或 `Fixed`	枚举	`Until Release`
Fixed Duration (ms)	Fixed 模式下的音符时长	`1` 到 `5000`	`100`

18.11 Project Settings（项目设置）

项目级别的设置在所有 Layout 间全局生效。

参数	说明	范围	默认值
Clock source	`INT`、`USB-dev`、`MIDI`、`USB-host`	枚举	`INT`
Tempo	内部速度	1 到 999 BPM	`120` BPM
Metronome	启用节拍器输出	on / off	off
Pedal A / **Pedal B** Global Enable	激活踏板输入	on / off	off
CC on layout change	切换 Layout 时发送元素的值	on / off	on
Switch Mode	音序器演奏型切换模式：`Wait`、`Jump`、`Restart`	枚举	`Wait`

MIDI 2.0（备用 USB 模式，重启后生效）是同一菜单中的一项设备设置，默认关闭。

18.11.1 CV Clock Output（CV 时钟输出）

参数	说明	范围	默
CV Clock	启用 CV 时钟输出	on / off	o
Beat Division	时钟速率：1 ppqn、2 ppqn、4 ppqn、8 ppqn、24 ppqn、48 ppqn	枚举	4
Clock Output	承载时钟的 CV 输出	1 到 23	1
Reset Output	只读，时钟输出之后的输出	Clock Output 加 1	2

18.11.2 MIDI Routing（MIDI 路由）

每个路由开关将传入的 MIDI 从一个端口转发到另一个端口。

参数	说明	默认值
MIDI -> USB Host	将 MIDI In 路由到 USB Host	off
MIDI -> USB Device	将 MIDI In 路由到 USB Device	off
MIDI -> MIDI A	将 MIDI In 路由到 MIDI Output A	off
MIDI -> MIDI B	将 MIDI In 路由到 MIDI Output B	off
USB Device -> USB Host	将 USB Device 输入路由到 USB Host	off
USB Device -> MIDI A	将 USB Device 输入路由到 MIDI Output A	off
USB Device -> MIDI B	将 USB Device 输入路由到 MIDI Output B	off
USB Host -> USB Device	将 USB Host 输入路由到 USB Device	off
USB Host -> MIDI A	将 USB Host 输入路由到 MIDI Output A	off
USB Host -> MIDI B	将 USB Host 输入路由到 MIDI Output B	off

> **提示：**当把 Erae 2 用作 USB 转 MIDI 接口时，MIDI 路由非常有用。启用 **MIDI -> USB Device**，即可将外接键盘与您的 DAW 所看到的表面输出合并到一起。

19 附录 B：MIDI 实现表

Erae 2 通过 USB Device（符合 class compliant 标准）、USB Host、MIDI Out A 和 MIDI Out B 发送 MIDI。每个元素独立选择其输出目标。传入的 MIDI 通过 USB Device、USB Host 和 TRS MIDI 输入接收，并可通过项目的路由开关（**Settings -> Routing**）在各端口之间转发。

在表中，**识别**表示 Erae 2 本身会对该消息作出响应。它不作响应的传入通道消息仍可由路由开关转发。

19.1 主实现表

功能	发送	识别	备注
Basic Channel	每个元素 Ch 1 到 16		每个元素拥有自己的通道
Note On	是	否	Keys、Buttons（Note 类型）、键盘元素、Switch 和 Kick 踏
Note Off	是	否	在抬指时发送；Lift Tune 设置松开力度
Polyphonic Aftertouch	是（可选）	否	Pressure Type 为 `PolyPressure` 时由 Key 和键盘元素发送
Channel Aftertouch	是（可选）	否	Pressure Type 为 `ChannelPressure` 时由 Key 和键盘元素
Control Change	是	否	见下方 CC 表
Program Change	是	否	Buttons（Program Change 类型）和 Switch 踏板，带可选
Pitch Bend	是	否	Key 和键盘元素；范围为 1 到 96 个半音（默认 12，MPE 时
14-bit CC	是	否	控制器类型 `CC14bit`：MSB 位于 CC 编号，LSB 位于 CC 编
RPN	是	否	在提供 CC Type 的元素和踏板上使用控制器类型 `RPN`
NRPN	是	否	在提供 CC Type 的元素和踏板上使用控制器类型 `NRPN`
MPE	是	否	按键盘启用；MPE 输出始终使用 `Erae 2 MIDI (MPE)` 端
System Exclusive	是	是	Universal Identity Request，以及 Embodme 协议（见下文
MIDI Clock (0xF8)	是	是	**Clock source** 为 `INT` 时发送；选择外部时钟源时跟随
MIDI Start (0xFA)	是	是	内部时钟启动时发送；使用外部时钟源时跟随
MIDI Stop (0xFC)	是	是	内部时钟停止时发送；使用外部时钟源时跟随
MIDI Continue (0xFB)	否	是	使用外部时钟源时，若已停止则开始播放

功能	发送	识别	备注
Active Sensing (0xFE)	否	否	未使用

使用内部时钟时，Clock、Start 和 Stop 会发送到 USB Device、USB Host、MIDI A 和 MIDI B。

> **提示：** Erae 2 不发送 MPE 配置消息或 pitch bend 范围消息。请将您的乐器的 MPE pitch bend 范围设置为与键盘的 pitch bend 范围一致（默认 48 个半音）。

19.2 MPE 详情

当键盘元素上启用了 **MPE** 时，Erae 2 作为 MPE Lower Zone（主通道 1）或 Upper Zone（主通道 16）发送，由键盘的 **MPE Master Channel** 设置决定。

MPE 参数	值
Master Channel	Ch 1：Lower Zone（默认）；Ch 16：Upper Zone
Member Channels（Lower Zone）	从 Ch 2 向上，动态分配
Member Channels（Upper Zone）	从 Ch 15 向下，动态分配
Per-Note Pitch Bend	在成员通道上发送
Per-Note Pressure	在成员通道上发送
Per-Note Timbre	成员通道上的 CC74（当键盘的滑移输出已启用时）
Pitch Bend Range（成员）	键盘的 pitch bend 范围，默认 48 个半音
USB 端口	`Erae 2 MIDI (MPE)`

19.3 Universal Identity Request

Erae 2 会以设备 ID `0x01` 或 `0x7F` 应答标准的 Identity Request（`F0 7E <id> 06 01 F7`），并在请求到达的 USB 端口上回复：

```
F0 7E 01 06 02 00 21 50 00 01 00 02 <channel> <major> <minor> <patch> F7
```

`00 21 50` 是 Embodme 厂商 ID，`00 01` 是 Erae 系列，`00 02` 是 Erae 2 型号。最后四个数据字节给出固件版本。

19.4 SysEx 协议

外部程序可在 Erae 2 上使用的 Embodme SysEx 消息是**开发者 API** 命令，它们用于读取来自 API Zone 元素的触摸并绘制其 LED。这些消息以如下头部开始：

```
F0 00 21 50 00 01 00 02 01 01 04 <sub-service> <payload...> F7
```

`00 21 50` 是 Embodme 厂商 ID，`00 01 00 02` 标识 Erae 2，`01` 是设备 ID，`01 04` 选择 API 服务。发往其他设备 ID 的消息（包括广播 ID `0x7F`）会被忽略。

19.4.1 API Zone 命令

子服务	方向	描述
StartFingerDataStreaming (`0x01`)	Host -> Device	开始发送来自 API 区域的手指数据
EndFingerDataStreaming (`0x02`)	Host -> Device	停止手指数据
ZoneBoundaryRequest (`0x10`)	Host -> Device	查询 API 区域的尺寸
Clear (`0x20`)	Host -> Device	清除 API 区域的 LED
SetPixel (`0x21`)	Host -> Device	设置一个 LED
DrawRectangle (`0x22`)	Host -> Device	填充一个矩形
DrawImage (`0x23`)	Host -> Device	绘制一张图像
VersionRequest (`0x7F`)	Host -> Device	查询 API 协议版本

字节布局、坐标、颜色与图像编码以及实用示例，均记录于附录 D：开发者 API。Erae Lab 和 Ableton Live Remote Script 使用它们自己的消息；您无需自行发送这些消息。

19.5 默认 CC 映射

启用相关功能时，默认使用以下 CC 编号。元素和踏板的 CC 分配可在 Erae Lab 中或设备上更改。

CC 编号	名称	使用者	备注
CC0 / CC32	Bank Select MSB / LSB	Buttons（Program Change）和 Switch 踏板	启用库选择时
CC11	Expression	Expressive 踏板	默认 CC 编号
CC64	Sustain	Sustain 踏板	Binary：开值和关值；Continuous：跟随踏板
CC74	Timbre	键盘滑移输出	键内的 Y 位置；标准 MPE 音色轴
CC102 到 CC109	Layout 切换	**CC on layout change**	通道 16，仅 USB Device：进入 Layout 时为 127，离开时为 0
用户自定义	Pressure	Key 和推子的压力 CC	可选
用户自定义	X Position	Key 和推子的 X 绝对 CC	绝对水平位置
用户自定义	Y Position	Key 和推子的 Y 绝对 CC	绝对垂直位置
用户自定义	X Relative	Key 和键盘的 X 相对 CC	自触摸开始以来的移动
用户自定义	Y Relative	Key 和键盘的 Y 相对 CC	自触摸开始以来的移动
用户自定义	Motion Speed	Key 和推子的运动速度 CC	手指移动速度，经过平滑并从 0 归一化到 100 cm/s

20 附录 C：DAW 设置指南

本附录针对最常见的 DAW 与硬件配置，提供分步连接指南。无论您是要路由标准 MIDI、启用逐音符的 MPE 表现力，还是连接到硬件合成器和 Eurorack 模块，都可以在这里找到相应的操作流程。

20.1 通用 MIDI 设置

20.1.1 USB MIDI 端口

当 Erae 2 以普通 MIDI 1.0 模式通过 USB 连接时，您的 DAW 会在单个符合类规范的 USB MIDI 设备中看到两根面向用户的 MIDI 线缆。在 macOS、Windows 10/11 或 Linux 上均无需安装驱动程序。

线缆	端口名称	用途
Cable 0	`Erae 2 MIDI`	双向标准 MIDI：音符、力度、Pitch Bend、CC、时钟。大多数 DAW 乐器轨道、发送给 Erae 2 的时钟以及常规路由都使用此端口。
Cable 1	`Erae 2 MIDI (MPE)`	仅 MPE 输出：面向 MPE 兼容乐器的全部逐音符表现力（Pitch Bend、压力、滑移）。MPE 消息始终发送到此线缆；该路由不可配置。Erae 2 不会在此线缆上接收 MIDI。

快速经验法则：将非 MPE 乐器轨道路由到 `Erae 2 MIDI`，将 MPE 乐器轨道路由到 `Erae 2 MIDI (MPE)`。MIDI 2.0 是另一种 USB 模式，通过 **Settings -> MIDI 2.0: ON/OFF** 控制，并需要重启设备。

> **陷阱：**除非轨道明确配置为 MPE，否则大多数 DAW 都会在将传入的 MIDI 送入 VST 或 AU 乐器之前，把它合并到单个通道（通常是通道 1）上。在设计供插件（例如 Erae Sound）使用的 Layout 时，请让所有发送 CC 的元素保持在**同一个 MIDI 通道**上，通道 1 是最安全的默认值。将宏 CC 分散到多个通道上的 Layout，会在非 MPE 的 DAW 轨道中丢失这些 CC。此限制不适用于硬件路由、MPE 端口上的 MPE 键盘，或明确启用了 MPE 的 DAW 轨道。

20.1.2 在 DAW 中使用 Erae Sound

当 Erae Sound 作为 VST3 或 AU 插件加载到 DAW 乐器轨道时，Erae 控制面通过标准 USB MIDI 端口发送数据，无需专有驱动。

推子手势和键盘音符（非 MPE）通过 **`Erae 2 MIDI`** 到达。请将该端口作为乐器轨道的 MIDI 输入启用，并将通道设置为与您的 Layout 相匹配（默认：通道 1）。

MPE 滑移和逐音符表现力通过 **`Erae 2 MIDI (MPE)`** 到达。要获得完整的 MPE 播放效果，请在乐器轨道上启用 MPE 模式，将其输入指向 `Erae 2 MIDI (MPE)`，并将通道设为 **All Channels**，以便全部 16 个成员通道都能到达乐器。

> **不要同时运行 Erae Sound 独立版与插件版。**两个实例同时监听相同的 Erae MIDI 输入。音符和手势会同时到达两个引擎，导致双重触发，两路输出都会产生声音。如果在通过插件播放时听到来自 Erae Lab 的声音，请点击 Erae Lab 顶部标题栏中 **Sound** 标签页上的电源图标，以禁用 Erae Sound，这将停止其音频引擎，同时保持设备连接，Layout 编辑和同步均不受影响。

20.1.3 macOS：移除陈旧的设备条目

固件更新后，macOS 有时会在新设备条目旁保留旧的 USB 设备条目。这可能导致端口名称重复或带编号（例如 `Erae 2 2`）。

清理方法：

1. 打开 **Audio MIDI Setup**（位于 `/Applications/Utilities/`）。
2. 选择 **Window -> Show MIDI Studio**（或按 **Command-2**）。
3. 找到任何带有警告图标或编号后缀的旧 `Erae 2` 条目。
4. 右键单击它并选择 **Remove Device**。
5. 拔出并重新插入 USB-C 线缆。设备会重新枚举，干净地呈现为单个条目。

> **提示：**移除陈旧条目后，请重启任何已打开的 DAW，使其重新读取更新后的端口列表。

20.1.4 Windows：确认端口可见性

1. 打开**设备管理器**，展开**声音、视频和游戏控制器**或**通用串行总线控制器**部分。
2. 确认 Erae 2 出现且没有黄色警告图标。
3. 在您的 DAW 中触发 MIDI 设备重新扫描或重启应用程序。Windows MIDI 端口在驱动级别注册：并非所有宿主都支持热重载。

> **提示：**如果设备出现但 MIDI 数据未流通，请检查您的 DAW 是否在该 MIDI 端口上设为“独占模式”，这会阻止其与其他应用程序共享。

20.2 Ableton Live

Ableton Live 11 和 12 在标准 MIDI 和 MPE 两方面都与 Erae 2 完全兼容。Embodme 提供了一个 MIDI Remote Script，用于 LivePad 的片段和场景控制元素。

20.2.1 第 1 步：启用 MIDI 端口

1. 打开 **Ableton Live -> Preferences**（macOS：**Live -> Settings**，Windows：**Options -> Preferences**）。
2. 点击 **Link, Tempo & MIDI** 选项卡（Live 11：**Link/Tempo/MIDI**）。
3. 在 **MIDI Ports** 部分，找到 **`Erae 2 MIDI`** 的 Input 行。
   - 将 **Track** 设为 **On**：这允许在乐器轨道上接收来自该端口的 MIDI 数据。
   - 将 **Remote** 设为 **On**：这允许 Control Surface 脚本使用该端口进行片段启动和参数映射。
4. 找到 **`Erae 2 MIDI (MPE)`** 的 Input 行。
   - 将 **Track** 设为 **On**。
   - 此端口的 **Sync** 和 **Remote** 保持关闭（MPE 端口不需要它们）。
5. 如果您希望 Live 将 MIDI 回送给 Erae 2（例如用于时钟同步或 LivePad 的 LED 反馈），可选地启用 `Erae 2 MIDI` 的 Output 行。MPE 端口没有通往 Erae 2 的输出。

> **陷阱：**如果您只启用 `Erae 2 MIDI` 而不启用 `Erae 2 MIDI (MPE)`，MPE 乐器将无法接收到表现力数据。完整的表现力设置需要两个端口都启用 Track。

20.2.2 第 2 步：设置标准 MIDI 乐器轨道

1. 创建一个带有目标乐器的 **Audio** 或 **MIDI** 轨道。
2. 在轨道的 **MIDI From** 选择器（轨道上方的输入选择器）中，选择 **`Erae 2 MIDI`**。
3. 将通道设为 **All Channels**，或设为与您的 Erae 2 元素 MIDI 通道分配相匹配的特定通道。
4. 为轨道启用录音待命，或将监听设为 **In** 以实时听到声音。

20.2.3 第 3 步：设置 MPE 乐器轨道

1. 创建一个带有 MPE 兼容乐器（Ableton 的 **Drift**、**Meld** 或任意 MPE 插件）的 MIDI 轨道。
2. 在 **MIDI From** 中，选择 **`Erae 2 MIDI (MPE)`**。
3. 将通道设为 **All Channels**：MPE 要求同时接收全部 16 个通道。
4. 打开乐器的插件设置，如果乐器有明确的开关，请启用 **MPE mode**。

> **提示：** Ableton 自家的 MPE 乐器（Drift、Meld）在检测到多通道输入时会自动启用 MPE。第三方 MPE 插件可能需要在其设置面板中手动切换。

20.2.4 第 4 步：安装 MIDI Remote Script（LivePad 元素）

Erae 2 MIDI Remote Script 使 LivePad 元素能够在 Ableton Live 中启动片段、触发场景，并在 session 网格中导航。

1. 从 Embodme 支持站点下载 **Erae 2 Remote Script** 安装包。
2. 将脚本文件夹（名为 `Erae2`）复制到 Ableton 用户库的 MIDI Remote Scripts 文件夹中：
   - **macOS：** `~/Library/Preferences/Ableton/Live x.x.x/User Remote Scripts/`
   - **Windows：** `C:\Users\<username>\Documents\Ableton\User Library\Remote Scripts\`
3. 重启 Ableton Live。
4. 打开 **Preferences -> Link, Tempo & MIDI**。
5. 在 **Control Surface** 部分，点击一个空槽位，从下拉菜单中选择 **Erae2**。
6. 将 **Input** 设为 **`Erae 2 MIDI`**，将 **Output** 设为 **`Erae 2 MIDI`**。

控制面上的 LivePad 元素现在将通过 LED 颜色反馈实时反映片段状态（播放中、已停止、已排队）。

> **陷阱：** 如果 Control Surface 下拉菜单显示了 Erae2，但片段颜色未更新，请确认 Remote Script 的 **Output** 端口设为 `Erae 2 MIDI` 而不是 MPE 端口。LED 反馈通过主线缆传输。

20.2.5 MIDI 时钟同步（Ableton -> Erae 2）

要将 Erae 2 的琶音器和 Looper 同步到 Ableton 的速度：

1. 在 **Preferences -> Link, Tempo & MIDI** 中，找到 **`Erae 2 MIDI`** 的 Output 行。
2. 将 **Sync** 设为 **On**。
3. 在 Erae 2 上，打开 **Settings** 并将 **Clock source** 设为 `USB-dev`。
4. 在 Ableton 中按播放：Erae 2 会自动锁定到 Ableton 的速度。

20.3 Logic Pro

Logic Pro X（10.5+）和 Logic Pro for Mac 原生支持 MPE。每个支持 MPE 的轨道都使用多通道录制，逐通道的表现力会在 MIDI 区域中得以保留。

20.3.1 第 1 步：在 Audio MIDI Setup 中启用 MIDI 输入

1. 打开 **Audio MIDI Setup**，确认 Erae 2 出现在 **MIDI Studio** 窗口中，且标准端口和 MPE 端口均可见。
2. 如果没有端口出现，点击 **Rescan MIDI** 按钮（MIDI Studio 窗口顶部的弯曲箭头图标）。

20.3.2 第 2 步：配置 Logic 的 MIDI 偏好设置

1. 打开 **Logic Pro -> Settings -> MIDI**（macOS Ventura+）或 **Logic Pro -> Preferences -> MIDI**（更早的版本）。
2. 在 **General** 选项卡中，确认已启用 **Auto-demix by channel if multi-channel recording**。这可确保不同通道上的 MPE 数据在录制期间不会被合并到单个通道上。
3. 保持默认端口分配不变：Logic 会自动扫描并暴露所有可用的 MIDI 输入。

20.3.3 第 3 步：创建标准 MIDI 轨道

1. 在 **Tracks** 区域，创建一个新的 Software Instrument 轨道。
2. 打开轨道的 **Region Inspector** 或 **Channel Strip**，确认 MIDI 输入设为 **All**，或设为与您的 Erae 2 Layout 相匹配的特定通道。
3. Logic 会自动从所有已连接的 MIDI 设备接收数据。如果您只想将输入限制为 Erae 2，可在乐器槽位中使用 **External Instrument** 插件，并将输入指定为 `Erae 2 MIDI`。

20.3.4 第 4 步：逐轨道启用 MPE

1. 选中您想要进行表现力演奏的 Software Instrument 轨道。
2. 在 **Track Inspector**（左侧面板）中，点击 **i**（Information）按钮以展开轨道设置。
3. 找到 **MIDI Channel** 参数并将其设为 **All**：MPE 多通道录制所必需。
4. 点击 Track Inspector 中的 **MPE** 开关以启用逐音符表现力。启用后，Logic 会独立录制每个音符的 Pitch Bend、压力和滑移。

> **陷阱：**如果未逐轨道启用 MPE，Logic 会将所有通道合并到通道 1 上，逐音符表现力随之失效。每个接收来自 `Erae 2 MIDI (MPE)` 数据的轨道都必须激活 MPE 开关。

20.3.5 第 5 步：将轨道指向 MPE 端口

1. 在轨道的 **Channel Strip** 中，点击 **MIDI In** 部分。
2. 从输入源下拉菜单中，选择 **`Erae 2 MIDI (MPE)`**。
3. 将通道设为 **All**。

20.3.6 Logic Pro：“Reset MIDI Drivers” 变通方法

在某些 macOS 版本上，Logic 会在启动时缓存 MIDI 设备状态。如果 Erae 2 端口在 Audio MIDI Setup 中出现，但在 Logic 的端口选择器中缺失：

1. 在 Logic 中，打开 **Window -> Show MIDI Environment**（或按 **Command-0**）。
2. 从 **Midi Environment** 菜单栏中，选择 **Special -> Reset All MIDI Drivers**。
3. Logic 会重新扫描所有已连接的 MIDI 设备，而无需重启。
4. 关闭 MIDI Environment 窗口并返回 Tracks 区域：现在端口应当出现在输入选择器中。

> **提示：**如果您对 MIDI Environment 不太熟悉，更简单的变通方法是退出 Logic、拔出并重新插入 Erae 2，然后重新启动 Logic。MIDI 驱动缓存会在应用程序启动期间被清除。

20.3.7 MIDI 时钟同步（Logic -> Erae 2）

1. 在 **Logic Pro -> Settings -> MIDI** 中，启用 **Transmit MIDI Clock**。
2. 在时钟目标列表中勾选 `Erae 2 MIDI` 旁的复选框。
3. 在 Erae 2 上，打开 **Settings** 并将 **Clock source** 设为 `USB-dev`。

20.4 FL Studio

FL Studio（版本 20+）并不像上述 DAW 那样以相同方式实现原生 MPE。逐音符表现力需要手动的逐通道路由，或使用 Patcher 进行多实例乐器设置。对于标准的复音用途，FL Studio 与 Erae 2 的 `Erae 2 MIDI` 端口配合良好。

20.4.1 第 1 步：启用 MIDI 端口

1. 打开 **Options -> MIDI Settings**。
2. 在 **Input** 部分，在设备列表中找到 **`Erae 2 MIDI`**。
3. 点击以高亮它，然后勾选 **Enable** 以激活该端口。
4. 将其分配到一个可用的 **Port** 编号（例如 Port 0）。请记下此编号：您将用它把 MIDI 路由到特定乐器。
5. 如果您计划使用逐通道路由，请对 **`Erae 2 MIDI (MPE)`** 重复上述操作，为其分配一个不同的端口编号（例如 Port 1）。
6. 如果您希望 FL Studio 的走带控制器同步 Erae 2 的时钟，请启用 **Send master sync**。

> **陷阱：**默认情况下，FL Studio 的 MIDI 端口启用状态在会话之间不会保持。如果重启后端口显示为未勾选，请在 MIDI Settings 中重新启用它，并立即保存您的 FL Studio 项目。

20.4.2 第 2 步：将 MIDI 路由到乐器

1. 在 **Channel Rack** 中，右键单击您想要接收 Erae 2 数据的乐器。
2. 选择 **Receive notes from**，并选择您为 `Erae 2 MIDI` 分配的 Port 编号。
3. 在 Receive Notes 对话框中，将 **MIDI channel** 设为与您的 Erae 2 Layout 中分配的通道相匹配（默认：通道 1）。
4. 演奏控制面：音符现在应该会触发该乐器。

> **提示：**如果您希望多个乐器在不同通道上同时接收 Erae 2 的数据，请在 Channel Rack 的 MIDI 设置中为每个乐器分配不同的 MIDI 通道，并将其与 Erae 2 中的逐元素通道分配相匹配。

20.4.3 第 3 步：使用 Multi-Link 进行 CC 映射

FL Studio 的 **Multi-Link to Controllers** 功能是将 Erae 2 的 CC 输出映射到乐器参数的最快方式。

1. 打开 **Mixer** 或您想要控制的乐器插件。
2. 从菜单栏选择 **Tools -> Multi-link to controllers**（或右键单击任意旋钮并选择 **Link to controller**）。
3. 在 FL Studio 中移动目标旋钮或推子。
4. 触摸 Erae 2 上对应的控制面元素以发送 CC：FL Studio 会自动捕获 CC 编号并创建映射。
5. 点击 **Accept** 以确认。

> **提示：** Erae 2 的推子元素会在您于 Erae Lab 中配置的轴上发送 CC。一维竖向推子默认发送 CC Y Absolute；二维推子可同时发送 X 和 Y。将每个轴映射到一个单独的参数，即可从单个触摸区域获得二维表现力。

20.4.4 第 4 步：在 FL Studio 中近似实现逐通道 MPE

FL Studio 并不会原生地将 MPE 消息作为统一的乐器概念来处理。实现表现力演奏的实用方法是将每个 MIDI 通道路由到一个单独的乐器实例：

1. 将 Erae 2 的 MPE 键盘元素分配到 Lower Zone（Master Channel 1，成员通道 2-N）。
2. 在 FL Studio 中，向 Channel Rack 添加 N 个目标乐器实例。
3. 将每个实例分别设为 **Receive notes from Port 1**（分配给 `Erae 2 MIDI (MPE)` 的端口），通道分别为 1、2、3……
4. 如果您希望获得统一的音频输出，请将所有实例连接到同一个 Mixer 轨道。

对于支持 Patcher（FL 的模块化环境）的乐器，单个 Patcher 实例可以承载多个子乐器，并自动将每个传入通道路由到各自的子乐器，从而提供更易于管理的设置。

20.4.5 MIDI 时钟同步（FL Studio -> Erae 2）

1. 在 **Options -> MIDI Settings** 中，在输出部分选择 `Erae 2 MIDI`，启用它，并为其分配一个端口。

2. 为该端口启用 **Send master sync**。
3. 在 Erae 2 上，打开 **Settings** 并将 **Clock source** 设为 `USB-dev`。

20.5 Bitwig Studio

Bitwig Studio（版本 3.2+）拥有所有 DAW 中最全面的 MPE 支持之一。Note Expression 模式让您能够在 Piano Roll 中以每个音符独立的自动化通道的形式查看和编辑逐音符的 Pitch Bend、压力和音色。

20.5.1 第 1 步：配置控制器

1. 打开 **Bitwig Studio -> Settings -> Controllers**。
2. 点击 **+** 按钮以添加一个控制器。
3. 从制造商/脚本列表中选择 **Generic -> Generic Flexi**（或者，为获得完整的 MPE 集成，选择 **ROLI -> Seaboard Rise**：此脚本可与任意 MPE 设备正常配合）。
4. 将 **Input** 设为 **`Erae 2 MIDI`**，将 **Output** 设为 **`Erae 2 MIDI`**。

> **提示：** Generic Flexi 脚本允许您将任意 CC 或 Note 消息映射到任意 Bitwig 参数。如果您只需要音符和 CC，而不需要 ROLI 特有的功能，Flexi 配置起来更简单。如果您想在 Bitwig 的 Note Expression 通道中获得完整的 MPE 音符表现力，请使用 ROLI Seaboard Rise 脚本：它对任意 MPE 发送端都能以相同方式运作。

20.5.2 第 2 步：在乐器轨道上启用 MPE 输入

1. 创建一个带有 MPE 兼容插件，或 Bitwig 自家的 **Polysynth** 或 **Phase-4** 的 **Instrument** 轨道。
2. 在轨道头部，点击 **MIDI Input** 选择器。
3. 选择 **`Erae 2 MIDI (MPE)`** 作为输入端口。
4. 将通道设为 **All**：Bitwig 需要看到全部 16 个通道才能重建逐音符表现力数据。

20.5.3 第 3 步：验证 Note Expression

1. 录制一小段表现力演奏。
2. 打开所录制区域的 **Piano Roll**。
3. 点击任意音符：您应该会在该音符下方看到 **Pitch**、**Pressure** 和 **Timbre (CC#74)** 的逐音符表现力通道。每条通道都会显示为该音符单独录制的连续数据。

> **陷阱：**如果所有音符显示的是相同的表现力数据，而不是逐音符数据，请确认轨道输入设为 `Erae 2 MIDI (MPE)` 而不是 `Erae 2 MIDI`。标准的 MIDI 通道压力是逐通道消息；逐音符表现力需要 MPE 线缆。

20.5.4 第 4 步：使用 Generic Flexi 映射 CC

1. 在 **Settings -> Controllers** 中，点击 Flexi 脚本条目，然后打开其设置面板。
2. 点击绑定槽位旁的 **Learn**。
3. 在 Erae 2 上移动一个推子或 CC 源：Bitwig 会自动捕获 CC 编号。
4. 使用同一面板中的参数浏览器，将其分配给任意 Bitwig 设备参数。

20.5.5 MIDI 时钟同步（Bitwig -> Erae 2）

Bitwig 会向任何启用了 Sync 的输出端口发送 MIDI 时钟：

1. 在 **Settings -> Controllers** 中，选择 Erae 2 控制器条目。
2. 在控制器选项中启用 **Send Clock**。
3. 在 Erae 2 上，打开 **Settings** 并将 **Clock source** 设为 `USB-dev`。

20.6 Cubase / Nuendo

Cubase Pro（版本 10.5+）和 Nuendo 通过 **Expression Map** 系统和 **MIDI Polyphonic Expression** 设备声明支持逐音符 MIDI 表现力。Cubase 12+ 增加了专门的 MPE 轨道支持。

20.6.1 第 1 步：启用 MIDI 端口

1. 打开 **Studio -> Studio Setup**（Cubase）或 **Devices -> Device Setup**（旧版本）。
2. 在左侧面板中选择 **MIDI Port Setup**。
3. 在列表中找到 **`Erae 2 MIDI`**。确保 **In**（输入）列已勾选（可见/激活）。
4. 找到 **`Erae 2 MIDI (MPE)`**，同样勾选其 **In** 列。
5. 点击 **OK** 以应用。

> **陷阱：** Cubase 拥有独立的 **Visible** 和 **Active** 列。端口必须标记为 **Active** 才会出现在轨道输入选择器中。被标记为 **Visible** 但未标记为 **Active** 的端口会出现在列表中，但不会传递数据。

20.6.2 第 2 步：创建标准 MIDI 或乐器轨道

1. 创建一个新的 **Instrument Track**（或路由到机架乐器的 **MIDI Track**）。
2. 在轨道的 **Inspector**（左侧面板）中，将 **MIDI Input** 设为 **`Erae 2 MIDI`**。
3. 将 **Channel** 设为 **Any**，或设为您的 Erae 2 元素所使用的特定通道。

20.6.3 第 3 步：将 Erae 2 声明为 Poly Expression 设备（Cubase 12+）

Cubase 12 引入了显式的 MPE 支持。要启用它：

1. 打开 **Studio -> Studio Setup**。
2. 在 **MIDI Port Setup** 下，找到 `Erae 2 MIDI (MPE)` 并点击 **Edit**（铅笔）图标。
3. 将 **Device Type** 设为 **MPE Instrument**。
4. 将 **Lower Zone Master Channel** 设为 **1**（与 Erae 2 的默认 MPE 配置相匹配）。
5. 将 **Member Channels** 设为您所需的同时发声数（默认：Lower Zone 为 15，使用通道 2-16）。
6. 点击 **OK**。

20.6.4 第 4 步：创建 MPE 乐器轨道

1. 创建一个带有 MPE 兼容 VST3 乐器的新 **Instrument Track**。
2. 在 Inspector 中，将 **MIDI Input** 设为 **`Erae 2 MIDI (MPE)`**。
3. 将 Channel 设为 **Any**。
4. 打开乐器自身的设置并激活其 MPE 模式（通常标记为 **Poly Expression**、**MPE** 或 **Per-Note Modulation**）。

> **提示：**并非所有 VST3 乐器都支持 Note Expression API。支持的乐器（Komplete Kontrol、Equator2、Pigments、Serum 2）会在 MPE 输入激活时于 Key Editor 中显示逐音符调制通道。

20.6.5 第 5 步：逐 Layout 的 MPE 区域配置

如果您的 Erae 2 Layout 使用了不同的 MPE 区域配置（有些 Layout 在 Ch 1 上使用 Lower Zone，另一些可能在 Ch 16 上使用 Upper Zone），您可以在 Cubase 的 Studio Setup 面板中保存相匹配的 MIDI Device 设置，并按需在它们之间切换。大多数用户始终只需要 Lower Zone 默认配置。

20.6.6 MIDI 时钟同步（Cubase -> Erae 2）

1. 打开 **Transport -> Project Synchronization Setup**。
2. 确保 **MIDI Clock** 已启用。
3. 在 **MIDI Clock Destinations** 列表中，勾选 **`Erae 2 MIDI`**。
4. 在 Erae 2 上，打开 **Settings** 并将 **Clock source** 设为 `USB-dev`。

20.7 Reaper

Reaper（版本 6.0+）原生支持 MPE。MPE 数据在 MIDI 编辑器中以独立的 Pitch Bend 和 CC 包络的形式逐音符存储，并且逐音符表现力会在录制、播放和编辑过程中得以保留。

20.7.1 第 1 步：启用 MIDI 设备

1. 打开 **Options -> Preferences -> Audio -> MIDI Devices**。
2. 在 **MIDI Inputs** 列表中，找到 **`Erae 2 MIDI`** 并双击它以启用（显示为勾选标记）。
3. 找到 **`Erae 2 MIDI (MPE)`** 并双击以同样启用它。
4. 点击 **Apply**。

> **陷阱：** Reaper 要求设备已连接后才会出现在 MIDI Devices 列表中。如果您是在打开 Preferences 之后才插入 Erae 2，请点击 **Reset** 按钮以重新扫描。

20.7.2 第 2 步：创建用于标准用途的 MIDI 轨道

1. 插入一个新轨道（**Track -> Insert New Track**）。
2. 点击轨道的 **ARM** 按钮（红色圆点）。
3. 轨道输入默认为所有 MIDI。要专门限制为 Erae 2，请点击轨道的输入选择器并选择 **MIDI Input -> `Erae 2 MIDI` -> All Channels**。

20.7.3 第 3 步：在轨道上启用 MPE 输入

1. 插入一个轨道并为其启用录音待命。
2. 点击轨道输入选择器并选择 **MIDI Input -> `Erae 2 MIDI (MPE)` -> All Channels**。
3. 在该轨道上插入一个 VST3 MPE 兼容乐器（例如 Pigments、Equator2）。
4. 在乐器的设置中，启用其 MPE 模式。

Reaper 会在单次录制中录入全部 16 个 MIDI 通道，保留完整的逐音符表现力。在 MIDI 编辑器中，启用 **View -> Show per-note pitch** 即可查看每个音符事件单独的 Pitch Bend 曲线。

20.7.4 第 4 步：MPE Pitch Bend 范围匹配

Erae 2 的 MPE 键盘元素默认的 Pitch Bend 范围为 48 个半音。请确保您的乐器与之匹配：

1. 在所录制区域的 MIDI 编辑器中，右键单击 Pitch Bend 通道。
2. 将 **Pitch Bend Range** 设为 **±48 个半音**（或与 Erae 2 元素设置相匹配的任意值）。
3. 在乐器插件中，将其 Pitch Bend 范围设为相同的值。

> **提示：** Pitch Bend 范围不匹配是 MPE 设置中最常见的音高跟踪错误原因。如果向上演奏一个八度听起来像两个八度，那就是范围不一致。请将两侧都设为 48 个半音。

20.7.5 第 5 步：MIDI 时钟同步（Reaper -> Erae 2）

1. 打开 **Options -> Preferences -> MIDI Devices**。
2. 启用 `Erae 2 MIDI` 输出设备。
3. 打开 **File -> Project Settings -> Notes, Media**，启用 **Send MIDI clock to** Erae 2 MIDI 输出。

4. 在 Erae 2 上，打开 **Settings** 并将 **Clock source** 设为 `USB-dev`。

20.8 连接到硬件合成器

Erae 2 专为直接的硬件集成而设计。两个物理端口可将 MIDI 传送到合成器和鼓机，信号路径中无需计算机。

20.8.1 TRS MIDI 输出

后面板包含一个 3.5 mm 的 TRS MIDI 输出。包装盒内附有两个 TRS 转 DIN 5 针的转接器。

接线类型： Erae 2 的 TRS MIDI 插孔采用 **Type A** 接线，并且没有可更改它的设置。带 Type A TRS 输入的设备使用普通 TRS 线缆即可连接；对于需要 Type B 的设备，请使用 Type A 转 Type B 的转换线。请查看合成器的手册以确认其预期的 TRS 类型：接线不匹配是硬件 MIDI 无法通过 TRS 工作的最常见原因。

连接步骤：

1. 将附带的 TRS 转接器连接到 Erae 2 后部的 MIDI Output A 接口。
2. 用一根标准的 5 针 DIN MIDI 线缆，从转接器连接到合成器的 MIDI In 插座。
3. 在 Erae Lab（或 LCD 映射屏幕）中，打开您想要路由到硬件合成器的元素。
4. 在 **MIDI Output Destination** 中，启用 **MIDI A**（如果使用第二个输出则启用 **MIDI B**）。
5. 将元素的 **MIDI Channel** 设为与合成器的接收通道相匹配。
6. 演奏控制面：音符应该会通过硬件合成器发声。

> **提示：**您可以同时启用多个目标。一个同时勾选了 **USB Device** 和 **MIDI A** 的元素，会同时向您的 DAW 和硬件合成器发送数据。

20.8.2 USB Host 端口

USB Host 端口允许 Erae 2 充当 USB 主机，为符合类规范的 USB MIDI 设备供电并与之通信，而无需计算机。

功率预算： USB Host 端口可提供高达 500 mA。需求更高的设备（例如某些带背光的 USB 键盘）可能需要外部供电。

连接步骤：

1. 将 USB MIDI 合成器、鼓机或 USB 转 DIN 转接器连接到后面板上的 USB Host 端口。
2. 已连接的设备会在 Erae 2 的 MIDI 路由中显示为 **USB Host** 端口。
3. 在 Erae Lab 中，打开该元素的设置，在 **MIDI Output Destination** 位掩码中启用 **USB Host**。
4. 将 MIDI 通道设为与合成器的接收通道相匹配。

> **提示：**如果已连接的 USB 设备未被识别，请确认它符合类规范（在 macOS/Windows 上无需驱动）。需要专有驱动的设备与 USB Host 模式不兼容。

20.8.3 硬件设置的 MIDI 路由矩阵

路由矩阵（在 LCD 上通过 **Settings -> Routing** 访问）控制哪些端口将传入的 MIDI 转发到其他端口。对于硬件设置：

有用的路由	原因
USB Device -> MIDI A	通过 Erae 2 将 DAW MIDI 发送到 MIDI A 上的硬件合成器
USB Device -> MIDI B	将 DAW MIDI 发送到 MIDI B 上的第二台硬件设备
MIDI -> USB Device	将硬件 MIDI 转发到您的 DAW 中以进行录制
USB Host -> MIDI A	将 USB MIDI 设备输出链接到 TRS 设备
MIDI -> MIDI A	无需计算机的硬件 MIDI thru

实例演练：将 Erae 2 作为 USB MIDI 接口

要将 Erae 2 用作 DAW 与硬件合成器之间的双向 MIDI 接口：

1. 启用 **MIDI -> USB Device**：硬件合成器的 MIDI 输出流入您的 DAW。
2. 启用 **USB Device -> MIDI A**：您的 DAW 通过 Erae 2 将 MIDI 发送给合成器。
3. 该合成器现在可作为受 DAW 控制的乐器使用，无需单独的 MIDI 接口。

实例演练：将控制面音符路由到硬件合成器

1. 在 Erae Lab 中，选中 Layout 上的键盘元素。
2. 在 **MIDI Output Destination** 中，启用 **MIDI A**（如果您不希望音符也发送到 DAW，请禁用 USB Device）。
3. 将 MIDI 通道设为合成器的接收通道（例如通道 1）。
4. 用一根 TRS 转接器和线缆，从 MIDI Output A 连接到合成器的 MIDI In。
5. 演奏控制面：音符会直接路由到合成器，全程无需计算机。

> **提示：**路由设置按项目（Project）保存。可以创建一个专门的 “硬件” 项目，将路由矩阵预先配置为适合您的合成器设置，再创建一个单独的 “DAW” 项目用于录音室使用。

20.9 Eurorack 集成

Erae 2 通过三种互补的方式连接到 Eurorack 模块化合成器：原生 CV 输出、经由 TRS 连接到 MIDI-CV 模块的 MIDI，以及经由 USB Host 连接到 USB MIDI-CV 转换器模块。

20.9.1 原生 CV 输出

Erae 2 在 3.5 mm 接口上提供 **24 个独立的 CV 输出通道**，直接兼容 Eurorack 和 1 V/octave 音高标准。电压范围和分辨率列于第 10 章。

详细的 CV 配置在第 10 章：CV 输出 中介绍。以下是 Eurorack 连接的摘要：

20.9.1.1 第一个声部：音高与 Gate

1. 在 Erae Lab 中，创建或编辑一个 Isomorphic Keyboard 元素。
2. 将 **CV Num Voice** 设为 1。
3. 分配基准输出通道（例如通道 1）。通道 1 承载音高（1 V/oct）；通道 2 承载 Gate（0 V / 5 V）。
4. 将通道 1 连接到您的 VCO 的 **V/oct** 输入。
5. 将通道 2 连接到您的包络发生器的 **Gate** 输入。
6. 演奏一个音符：音高以 1 V/octave 跟踪，并且包络会在每次音符开启时触发。

> **提示：**中央 C（MIDI 音符 60）输出 2 V，高八度的 C 输出 3 V。如果在您的 VCO 上八度跟踪略宽或略窄，请在 **Settings** 中调整 **CV Gain (cent/oct)**，或按 **Run CV calib** 重新校准输出。

20.9.1.2 额外的逐声部表现力

每个声部都可以将补充的表现力流作为 CV 输出。这些输出在每个声部上各占用一个 CV 通道：

CV 输出	电压范围	Eurorack 用途
Velocity	0-5 V	包络初始电平、VCA 偏移
Pressure	0-5 V	滤波器截止、VCA 增益、LFO 深度 CV
X Position	0-5 V	任何响应水平位置的参数
Y Position	0-5 V	任何响应垂直位置的参数
X Slide	以 2.5 V 为中心	颤音深度、经由 CV 的音高偏移
Y Slide	以 2.5 V 为中心	音色变化、共振峰位置

要启用这些输出：

1. 在 Erae Lab 中选中该元素，或在设备上打开它的 Mapping 界面。
2. 在 CV 部分中，启用所需的输出（Velocity、Pressure、X Position 等）。
3. 每个启用的输出在每个声部上占用一个 CV 通道。在启用较高声部数之前，请先规划好您在所有元素上的通道分配。

> **警告：**为 4 个声部启用全部六个可选输出，会占用 4 × 8 = 32 个 CV 通道：超过 Erae 2 可用的 24 个输出。请保持声部数与可选输出数量之间的平衡。

20.9.1.3 用于 Eurorack 时钟同步的 CV 时钟输出

Erae 2 的 CV Clock 输出会以可配置的 ppqn 速率向某个 CV 通道发送脉冲信号，并锁定到项目速度。

1. 按下前面板上的 **Settings** 按钮。
2. 导航到 **CV Clock**。
3. 将 **Enabled** 设为 On。
4. 将 **Beat Division** 设为 **24 ppqn**（标准 MIDI 时钟速率，与大多数 Eurorack 时钟分频器和受时钟驱动的模块兼容）。
5. 将 **Clock Output** 设为一个未使用的通道（例如通道 3）。
6. 将通道 3 连接到您受时钟驱动的模块的 **Clock In** 或时钟分频器输入。
7. 将相邻的复位输出连接到模块的 **Reset** 输入，以实现采样精度的启动对齐。

可用的 ppqn 速率：1、2、4、8、24、48 ppqn。在兼容模块上使用 48 ppqn 可获得高分辨率的 LFO 同步。

20.9.2 USB Host 连接到 Eurorack MIDI-CV 模块

若要获得比原生 CV 输出所允许的更高复音数，或在您希望由专用 Eurorack 模块处理 MIDI-CV 转换的设置中，请将该模块直接连接到 Erae 2 的 USB Host 端口。

该模块从 Erae 2 接收 MIDI（如果模块支持，包括 MPE），并在路径中无需计算机的情况下将其转换为 CV。

建议的模块：

模块	最适用于
Expert Sleepers FH-2	高复音数、MPE 支持、可深度配置
Intellijel uMIDI 1U	带 USB 输入的紧凑单声部 MIDI 转 CV
Mutable Instruments Yarns	最高 4 声部复音、琶音器、MIDI 时钟
Endorphin.es Shuttle Control	16 声部 USB MIDI、丰富的 CV 映射
Befaco MIDI Thing	简单、实惠的 TRS MIDI 转双路 CV

> **提示：** Expert Sleepers FH-2 是用于 Erae 2 集成的最强大选项。它可直接从 USB Host 端口接收 USB MIDI，支持 MPE 输入（将每个成员通道分配给一对单独的 CV 输出），并且可以从其基于浏览器的编辑器中完全配置。

Expert Sleepers FH-2 的连接步骤：

1. 用一根 USB-A 线缆从 Erae 2 的 USB Host 端口连接到 FH-2 的 USB 端口。
2. FH-2 会在 Erae 2 的 USB Host 上枚举为一个 USB MIDI 设备。

3. 如果您希望来自 FH-2 的 MIDI 到达您的电脑，请在 **Settings -> Routing** 中启用 **USB Host -> USB Device**。
4. 在 Erae Lab 中，将您的键盘元素的 **MIDI Output Destination** 设为包含 **USB Host**。
5. 在 FH-2 的浏览器编辑器中配置它：将 MIDI 通道分配给 CV 输出对，并为 MPE 声部将 Pitch Bend 范围设为 48 个半音。

20.9.3 示例补丁：完整的表现力声部

此示例将 Erae 2 通过一个完整的 Eurorack 声部进行路由，使用原生 CV 输出，并提供独立的逐音符表现力。

硬件： - Erae 2（经由 3.5 mm 接口的 CV 输出）- 任意带有 V/oct 和线性 FM 输入的 Eurorack VCO - 任意 ADSR 包络发生器 - 任意带 CV 截止输入的 VCF - 任意带 CV 增益输入的 VCA

补丁：

Erae 2 输出	Eurorack 目标	效果
CV Ch 1（音高）	VCO V/oct	音高跟踪触摸位置
CV Ch 2（Gate）	ADSR Gate In	触摸时包络触发
CV Ch 3（Velocity）	ADSR Initial Level	起音电平反映敲击力度
CV Ch 4（Pressure）	VCF Cutoff CV	滤波器随压力增加而打开
CV Ch 5（Pressure）	VCA Gain CV	音量随压力而渐强
CV Ch 6（X Slide）	VCO Linear FM	水平滑移 = 音高偏移（颤音）

在 Erae Lab 中的设置：

1. 创建一个 Isomorphic Keyboard 元素。
2. 将 **CV Num Voice** 设为 1。
3. 将基准通道设为 1。
4. 启用 **Velocity**、**Pressure** 和 **X Slide** 额外输出。
5. 通道 1-6 现已分配完毕；按表中所示进行连接。
6. 在元素设置中将 **Pressure Type** 设为 `PolyPressure`：这会将单根手指的压力映射到每个音符对应的正确 CV（适用于复音设置）。

> **提示：**先从单音补丁（CV Num Voice = 1）开始，验证所有 CV 信号是否按预期工作，然后再转向复音配置。在接入 VCO 之前，先用电压表在音高输出上确认 1 V/octave 的跟踪。

20.10 相关主题

- MIDI 配置：USB 端口布局、通道分配、MPE 区域、路由矩阵

- CV 输出：完整的 CV 输出参考、校准、逐声部分配
- 设置：时钟源、MIDI 路由、CV 时钟配置
- 故障排除：MIDI 端口未出现、音符不发声、路由回环
- 附录 B：MIDI 实现表：完整的消息表、MPE 细节、SysEx 协议

21 附录 D：开发者 API

Erae 2 为高级集成提供了一个可编程的表面：自定义 Max/MSP 补丁、TouchDesigner 设置、Web 工具、定制软件乐器。该 API 通过 USB MIDI 系统专有消息（SysEx）传输，并作用于 Layout 中放置的 **API Zone** 元素。宿主应用程序可以：

- 从一个或多个 API 区域接收原始多点触控数据（X、Y、压力），无需经过 MIDI 抽象层。
- 直接绘制每个 API 区域的 LED：清空、设置像素、填充矩形、绘制图像。
- 查询每个 API 区域的尺寸以及 API 协议版本。

本附录记录线路协议。如需从 Layout 作者的视角了解 API Zone 元素的概览，请参阅5.9 节：API Zone。

> **兼容性。**Erae 2 只应答发往设备 ID `0x01` 的 Erae 2 SysEx 头部（系列成员代码 `00 02`）。为 Erae Touch 编写的宿主软件使用成员代码 `00 01`，必须更改该代码才能与 Erae 2 通信。两者的命令 ID 和载荷布局相同。

21.1 D.1 传输

所有 API 消息都通过主 USB MIDI 端口传输。

方向	USB MIDI 端口
设备 -> 宿主（手指数据流、回复）	`Erae 2 MIDI`
宿主 -> 设备（绘制、清空、查询）	`Erae 2 MIDI`

`Erae 2 MIDI (MPE)` 端口只承载 MPE 输出，API 不使用它。

> **连接检查。**在 WebMIDI 宿主中，打开名为 `Erae 2 MIDI` 的输入和输出，即名称中不含 `MPE` 的那一个。

> **Linux。**当 Erae 2 处于 MIDI 2.0 模式时，Linux 内核将其旧版 MIDI 端口转换为 MIDI 2.0 数据包的过程，可能会损坏长度超过 6 个数据字节的 SysEx 消息，因此设备会收到错乱的命令而不作应答。如果您的工具通过旧版 ALSA 端口（例如使用 `amidi -p hw:...`）得不到回复，请将 MIDI 2.0 数据包直接发送到设备的 UMP 端口（`/dev/snd/umpC<card>D0`）。这不影响 macOS 或

> Windows。

21.2 D.2 SysEx 头部

每条宿主到设备的 API 消息都以一个固定的 11 字节头部（包括 F0 起始字节）开始，然后是子服务字节，并以 F7 结束：

```
F0  00 21 50  00 01  00 02  01    01   04   <sub>  <payload...>  F7
^   ^^^^^^^^  ^^^^^  ^^^^^  ^^    ^^   ^^   ^^^^^
|   Embodme   Erae   Erae 2 ID    Svc  API  Sub-
SysEx  manuf  family model        grp  svc  service
start  ID
```

字节	含义
F0	SysEx 起始
00 21 50	Embodme 厂商 ID
00 01	Erae 硬件系列代码
00 02	Erae 2 型号代码
01	设备 ID。Erae 2 只应答 0x01；API 会忽略广播 ID 0x7F
01	服务组
04	服务：API
<sub>	子服务 ID（参见D.4）
<payload...>	子服务专用字节
F7	SysEx 结束

设备到宿主的消息使用不同的封装格式，见D.3.4。

21.3 D.3 约定

21.3.1 D.3.1 坐标轴

Erae 表面是一个宽 42、高 24 的 LED 网格（1008 个 LED）。在 API 区域内，坐标是**区域局部**的：(0, 0) 是区域的一个角，(width-1, height-1) 是对角。X 从左到右递增。

> **Y 轴警告：请读一遍并牢记。**
>
> API 一直使用两种相反的 Y 约定，固件为了兼容而保留了它们：
>
> | 数据流 | Y 约定 | `y = 0` 表示 |
> |---|---|---|
> | 手指数据（设备 -> 宿主） | **以底部为原点** | 区域的最底行 |
> | 绘制命令（宿主 -> 设备） | **以顶部为原点** | 区域的最顶行 |
>
> 要点亮位于 `(x, y)` 处所报告手指下方的 LED，请在 `SetPixel(x, (height - 1) - y, color)` 处绘制。同样的翻转也适用于 `DrawImage`：在 `y = 0` 处绘制的图像会绘制在区域顶部，而手指报告中的 `y = 0` 是底部。

21.3.2 D.3.2 颜色编码

`SetPixel` 和 `DrawRectangle` 以三个 7 位 MIDI 字节（R、G、B）表示颜色。设备会将每个分量左移一位：

```
display_R = sysex_R << 1   // 0..254 in steps of 2
display_G = sysex_G << 1
display_B = sysex_B << 1
```

因此全白像素为 `7F 7F 7F`；纯红为 `7F 00 00`。

`DrawImage` 则不同：它的像素是完整的 8 位数值（`0x00` 到 `0xFF`），通过下文的 bitize-7 打包方式在 MIDI 中传输，并且不做任何移位直接使用。

21.3.3 D.3.3 Bitize-7（图像和手指数据编码）

MIDI 禁止数据字节设置第 7 位。因此二进制块使用 Embodme 的 7 位打包方式发送：每 7 个源字节变成 8 个 MIDI 字节。每组的第一个 MIDI 字节承载其后 7 个源字节各自的最高有效位（源字节 0 位于第 6 位，依次到源字节 6 位于第 0 位），随后的每个字节承载一个源字节的低 7 位。

打包器的伪代码（宿主端）：

```
for chunk of 7 source bytes:
   out.push(((src[0] >> 7) << 6) | ((src[1] >> 7) << 5) | ... | ((src[6] >> 7) << 0))
    for i in 0..7:
        out.push(src[i] & 0x7F)
# A trailing partial chunk emits its MSB byte first, then only the bytes present.
```

`N` 个源字节编码后的大小为 `ceil(N * 8 / 7)`。一个块的**校验和**是编码后（bitize 后）块中每个字节的异或，包括每组开头的 MSB 字节。由于每个编码字节都是 7 位的，校验和也是 7 位的。

> **常见错误。不要**对解码后的源字节做校验和。对于所有分量都不超过 0x7F 的图像，它恰好相符，但对于更明亮的图像则不然，而固件会悄悄丢弃校验和不匹配的 DrawImage。请对您放到线路上的字节做校验和。

> **参考实现。**一个公开的 API Python 示例（包括 bitize-7 打包器、校验和例程和图像上传）可在 erae_api_sysex 仓库（run_erae_api.py）中获取，同时附有 Erae_API_V2.pdf 协议规范。

21.3.4 D.3.4 流式传输激活与回复封装

只有在宿主发送 StartFingerDataStreaming（0x01）**之后**，才会发送手指数据和 ZoneBoundaryRequest 回复。流式传输状态对整个设备是全局的，而不是按区域区分。

每条设备到宿主的 API 消息都封装为：

```
F0  <prefix data>  <body>  F7
```

这些消息中没有 Embodme 头部：宿主提供的**前缀数据**（随 StartFingerDataStreaming 提供，或者对于版本回复，随 VersionRequest 提供）会被原样复制到 F0 之后。可利用它使回复成为对您的宿主有效的 SysEx，并区分多台 Erae 设备，例如让前缀以厂商 ID 字节（如 7D，保留给非商业用途）开头，后接一个会话标签。

21.4 D.4 宿主 -> 设备命令

除非另有说明，所有数值均为十六进制。<header> 表示 F0 00 21 50 00 01 00 02 01 01 04，如D.2 所示。

21.4.1 D.4.1 StartFingerDataStreaming（0x01）

启用手指数据和区域边界回复。可选的 <prefix data>（0 到 16 字节）会被复制到每条设备到宿主消息的开头（参见D.3.4）。再次发送会替换该前缀。

```
<header>  01  <prefix data 0..16 bytes>  F7
```

21.4.2 D.4.2 EndFingerDataStreaming（0x02）

禁用手指数据和区域边界回复，并清除前缀。无载荷。

```
<header>  02  F7
```

21.4.3 D.4.3 ZoneBoundaryRequest（0x10）

查询某个 API 区域的尺寸。设备会回复一条Zone Boundary Reply。若流式传输已禁用，则不会发送回复。

```
<header>  10  <zoneIdx>  F7
```

字节	字段	范围	备注
0	zoneIdx	0x00 .. 0x7F	区域索引，在 Erae Lab 中设置

21.4.4 D.4.4 Clear（0x20）

清空一个区域的 LED（以黑色填充）。

```
<header>  20  <zoneIdx>  F7
```

21.4.5 D.4.5 SetPixel（0x21）

写入单个 LED。

```
<header>  21  <zoneIdx>  <x>  <y>  <R>  <G>  <B>  F7
```

字节	字段	范围	备注
0	zoneIdx	0x00 .. 0x7F	目标区域
1	x	0 .. width-1	超出范围的写入会被悄悄丢弃
2	y	0 .. height-1	**以顶部为原点**（参见D.3.1）
3	R	0x00 .. 0x7F	内部左移 1 位
4	G	0x00 .. 0x7F	
5	B	0x00 .. 0x7F	

21.4.6 D.4.6 DrawRectangle（0x22）

用一种颜色填充一个矩形（7 位分量，与 SetPixel 相同）。

```
<header>  22  <zoneIdx>  <x>  <y>  <w>  <h>  <R>  <G>  <B>  F7
```

该矩形覆盖单元 (x .. min(x+w, width) - 1, y .. min(y+h, height) - 1)。如果 x >= width 或 y >= height，则不绘制任何内容。否则矩形会被裁剪到区域范围内。

21.4.7 D.4.7 DrawImage（0x23）

绘制一张 w x h 的彩色图像。像素按**行优先**顺序读取：第 0 行从左到右，然后是第 1 行，依此类推。每个像素为三个 8 位字节（R、G、B，各为 0x00 到 0xFF），按原样使用。

```
<header> 23 <zoneIdx> <x> <y> <w> <h> <bitized RGB data...> <chksum> F7
```

字节	字段	备注
0	zoneIdx	目标区域
1	x	图像在区域内的左上角 X
2	y	左上角 Y（以顶部为原点）
3	w	图像宽度（像素）
4	h	图像高度（像素）
5..N-1	bitized RGB	w * h * 3 个源字节，按D.3.3 编码
N	chksum	bitize 后数据每个字节的异或；参见D.3.3

固件会检查四项内容，任何一项不通过都会悄悄丢弃该消息：

1. 解码后的大小是 3 的倍数（只允许完整像素）。
2. 解码后的大小恰好等于 w * h * 3。
3. 图像能放入表面（w * h <= 1008）。
4. 校验和与 bitize 后字节的异或相符。

图像会被裁剪到区域的右边缘和下边缘（xmax = min(x+w, width)，ymax = min(y+h, height)）。区域外的像素不会被绘制；区域内的像素取自源图像中对应的位置。如果 x 或 y 位于区域之外，则不绘制任何内容。

21.4.8 D.4.8 VersionRequest（0x7F）

查询 API 协议版本。无论流式传输是否启用，设备都会回复一条Version Reply。

```
<header>  7F  <prefix data>  F7
```

此请求的 <prefix data> 会被复制到回复中（参见D.3.4），因此即使存在多台设备，宿主也能将回复与其请求配对。

21.5 D.5 设备 -> 宿主消息

所有设备到宿主的消息都以 F0 和宿主前缀数据开头，如D.3.4 所述。

21.5.1 D.5.1 Zone Boundary Reply

对ZoneBoundaryRequest 的回复。

```
F0  <prefix data>  7F  01  <zoneIdx>  <width>  <height>  F7
```

字节	字段	备注
0	0x7F	标记这是一条回复（而非手指数据）
1	0x01	区域边界回复
2	zoneIdx	从请求中复制
3	width	区域宽度（像素），若没有 API 区域使用该索引则为 0x7F
4	height	区域高度，若没有则为 0x7F

21.5.2 D.5.2 Version Reply

对VersionRequest 的回复。

```
F0  <prefix data>  7F  02  <apiVersion>  F7
```

<apiVersion> 为 0x02。它标识设备到宿主的线路格式，其中最重要的是Finger Stream 主体，仅当该格式变化时它才会改变。应答 VersionRequest 的设备会发送下文所述的 Finger Stream 主体。Erae Touch 固件和早期的 Erae 2 固件不会应答该请求。

21.5.3 D.5.3 Finger Stream

流式传输启用期间，API 区域内的每一次触摸事件都会发送。所有区域共享同一个数据流，并通过 zoneIdx 区分。

```
F0 <prefix data> <action> <zoneIdx> <bitized fingerIdx (uint64)> <bitized X,Y,Z (3 x fl
```

字节	字段	备注
0	`action`	`0x00` Click（手指按下），`0x01` Slide（移动），`0x02` Release（手指抬起）
1	`zoneIdx`	来源区域
2..11	`fingerIdx`	64 位无符号手指 ID，经 bitize-7 编码（8 字节在线路上变为 10 字节）
12..25	`x, y, z`	三个小端序 `float32` 值，作为一个 12 字节块进行 bitize-7 编码（线路上为 14 字节）：以像素为单位的区域局部位置（**Y 以底部为原点**，参见D.3.1）以及归一化压力 z（约 `0.0` 到 `1.0`）
26	`chksum`	`x`，`y`，`z` 块的 14 个 bitize 后字节的异或

按 (`zoneIdx`, `fingerIdx`) 重建每根手指的轨迹，并将 `Release` 视为一条轨迹的结束。Looper 回放的手势会以带有其自身 ID 的手指形式出现。

21.5.4 D.5.4 Max Num Fingers 与 Finger Data Rate

Erae Lab 还会显示两个 API Zone 参数：**Max Num Fingers**（`0` 到 `10`）和 **Finger Data Rate**。它们会保存在 Layout 中，但当前的 Erae 2 固件并不应用它们：区域内的每一次触摸都会被报告，`Slide` 消息以设备自身的扫描速率跟随手指。请将宿主的轨迹表按每个区域最多 10 根同时触摸的手指来设计；如果需要更少的更新，请在宿主端进行节流。

21.6 D.6 完整示例

所有示例都以区域 `0x00`、设备 ID `0x01` 为目标。

21.6.1 D.6.1 将像素（3，5）点亮为红色

`SetPixel，R=0x7F，G=0，B=0：`

```
F0 00 21 50 00 01 00 02 01 01 04 21  00 03 05 7F 00 00  F7
```

21.6.2 D.6.2 清空区域

```
F0 00 21 50 00 01 00 02 01 01 04 20  00  F7
```

21.6.3 D.6.3 以前缀 7D 01 启动手指数据流

```
F0 00 21 50 00 01 00 02 01 01 04 01  7D 01  F7
```

此后设备发送的每条消息都以 F0 7D 01 开头。例如，以相同前缀发送的 VersionRequest 的回复为 F0 7D 01 7F 02 02 F7。

21.6.4 D.6.4 在 (0, 0) 处绘制一张 [红, 绿] 的 2x1 图像

原始数据（完整 8 位）：FF 00 00 00 FF 00（6 字节，一个不完整分组）。

```
MSB byte = bit7(FF)<<6 | bit7(00)<<5 | bit7(00)<<4 | bit7(00)<<3 | bit7(FF)<<2 | bit7(00)<<
payload  = 44 7F 00 00 00 7F 00
```

校验和 = bitize 后字节的异或 = 44 XOR 7F XOR 00 XOR 00 XOR 00 XOR 7F XOR 00 = 0x44。

完整消息：

```
F0 00 21 50 00 01 00 02 01 01 04 23  00 00 00 02 01  44 7F 00 00 00 7F 00  44  F7
                                       |  |  |  |  |  -----------------  |
                                     zo x  y  w  h  bitized payload    chksum
```

21.7 D.7 错误处理与限制

条件	行为
消息发往 0x01 以外的设备 ID	被忽略
区域索引与当前 Layout 中的任何 API 区域都不匹配	命令被悄悄丢弃（无错误回复）
x 或 y 超出区域范围	命令被悄悄丢弃（SetPixel）或被裁剪（DrawRectangle、
DrawImage 校验和不匹配	命令被悄悄丢弃
DrawImage 解码后的大小不等于 w * h * 3	命令被悄悄丢弃
DrawImage 大于表面（1008 像素）	命令被悄悄丢弃
StartFingerDataStreaming 的前缀长于 16 字节	流式传输**不会**被启用
流式传输期间 API 区域从 Layout 中被移除	流式传输继续；发往该索引的命令会被丢弃，直到再次出现具
多个 API 区域使用相同的索引	只有第一个区域接收绘制命令；请避免重复的索引

限制：

限制	数值
前缀数据	16 字节
`DrawImage` 像素数	1008（整个表面）
传入的 SysEx 消息	8 KB。消息在到达过程中会被保留，无论到达得多慢；更长的消息会被丢弃
设备到宿主的消息	128 字节（包括前缀）

21.8 D.8 整合应用

一次典型的会话如下：

1. 打开 `Erae 2 MIDI` 输入和输出。
2. 使用您的前缀发送 `VersionRequest`，并检查回复是否报告 `0x02`。
3. 使用相同的前缀（例如一个厂商 ID 字节和一个会话标签）发送 `StartFingerDataStreaming`。
4. 对您使用的每个区域发送 `ZoneBoundaryRequest`，并保存 `(width, height)` 回复。
5. 使用 `Clear`、`DrawImage`、`DrawRectangle` 和 `SetPixel` 进行绘制。一张覆盖整个表面的图像在线路上约为 3.5 KB；请控制大图像的发送节奏，使宿主的 MIDI 输出跟得上，或将其拆分为较小的矩形。
6. 按 `(zoneIdx, fingerIdx)` 对传入的手指消息进行分类，在将触摸与绘制的像素相匹配时，请记住Y 轴翻转。
7. 当您的宿主断开连接时，发送 `EndFingerDataStreaming`。