

# QU-BIT

客户：  
Qu-Bit Electronix

行业：  
专业消费者

国家/地区：  
美国

意法半导体技术：  
• [STM32H7 MCU](#)  
• 低压降稳压器  
• 步降开关稳压器  
• MEMS音频传感器



## 推动新一轮的数字声音合成革命

当Qu-Bit Electronix于2020年3月为他们旗下的Daisy平台开展Kickstarter众筹活动的时候，音乐行业的面貌已悄然发生了巨大的变化。在合成器这一领域，“模块化”与“模拟”的含义几乎完全相同。因此当该公司打造出全新类别的数字模块化合成器时，广大音乐人和音响设计师都不同程度地对其抱有怀疑态度。

如今在许多人的心中，“数字”一词仍然等同于“耗电”和“高延迟”，而“延迟”是所有音乐人在录音过程中都不希望出现的现象。

为了打破这种刻板印象，Qu-Bit在设计环节十分注重控制设备的数字参数和功耗，并设法为客户提供新鲜的独创音频样本。

Qu-Bit所在的行业到处都弥漫着独立制作的氛围；为了使行业内的所有客户都能得到良好的产品使用体验，该公司必须确保能够为小规模的工作室带来难以置信的出色性能。在意法半导体的协助下，Qu-Bit说服了Kickstarter众筹活动的支持者，并有望将其首批解决方案授权给竞争对手。



STM32H7 MCU具有出色的速度、ADC分辨率和功率效率表现，能够为嵌入式音乐硬件的设计空间带来颠覆性的变化。

Andrew Ikenberry,  
Qu-Bit Electronix首席执行官

### 挑战

- 确保合成器不会出现任何意外延迟的迹象
- 设计出高品质的旋钮，不会使声音出现“跳跃性”变化或是降低分辨率
- 在新冠疫情造成供货短缺之际，得到10000颗能够满足Kickstarter活动支持者需求的芯片
- 在创新型的DSP算法与耗电的处理循环之间寻找到合适的平衡点

### 解决方案

- 重新设计16位旋钮，使用户无法仅凭外观分辨出产品是否采用了数字技术
- 利用STM32H7针对大量音频样本运行处理循环，运行次数为该公司旗下硬件可运行循环数的两倍
- 找到一款兼具微控制器通用功能与DSP独有功能的芯片

### 影响

- 即使在生产规模较小的情况下，也能通过解决方案授权的方式与竞争对手展开合作
- 输出不再受到硬件的限制，由此激励了一大批新的艺术家参与进来
- 新增了大量原创音效，从而推动了下一波创新浪潮



意法半导体的介入十分迅速，他们在两天内就交付了我们所需的芯片。

Andrew Ikenberry,  
Qu-Bit Electronix首席执行官

诚然，Qu-Bit Electronix在全新声音合成技术领域的不懈投入无疑是该公司取得成功的重要原因之一，但该公司所采用的高性能旋钮设计同样在一定程度上帮助客户克服了对数字模块化合成器的抗拒心理。当前的产品系列采用了分辨率可达65536步的16位控制器，即便是最挑剔的设备发烧友也无法分辨出该产品与模拟解决方案的差异。

STM32H7具备设计低延迟旋钮和完成大量数据采样所需的高处理能力，因而成为了该公司的理想选择。该器件兼具GPIO和USB连接的通用功能、DSP的独有特性以及480MHz的时钟频率。

意法半导体的芯片凭借更为出色的处理性能，帮助该公司获得了音响工程师和实验音乐人等各类客户最为重视的创新型声音表现。

在这一点上，Qu-Bit已经充分证明了数字并不逊色于模拟技术。该公司的相位声码器支持时间拉伸，这是一种模拟解决方案无法实现的全新效果。在过去，模块化合成器一直无法实现混响、延迟、时间拉伸和采样器等其他效果。

然而，打动Qu-Bit Electronix的并不仅仅是意法半导体产品过硬的技术规格。当时形势非常紧迫，大量宅在家中的设备爱好者纷纷出来参与Kickstarter上的众筹活动，因此Qu-Bit急需10000颗芯片来解他们的燃眉之急。意法半导体通过快速的产品交付促进了Qu-Bit Electronix的发展，并帮助行业内的其他公司与之共同成长。

未来Qu-Bit Electronix将继续与意法半导体开展合作，并继续通过创新手段大力优化产品功耗，从而帮助音乐人从容应对他们在日常工作中遇到的各种挑战。所有的创新成果都会在下一波声效和音乐创新浪潮中发挥巨大的推动作用。

### 关于Qu-Bit

Qu-Bit Electronix总部位于加利福尼亚州的圣克莱门特，自2013年以来一直专注于制造模块化数字合成器。多年来Qu-Bit一直深耕音乐行业，他们与竞争对手积极合作，所推出的解决方案曾为全球无数的实验音乐人、音响工程师和业余爱好者带来启发和力量。

Qu-Bit将旗下的产品视为公司的骄傲，并尽最大的努力在内部精心制作每一台设备。  
[访问Qu-Bit Electronix的网站。](#)

