



UFCOM 虚拟串口驱动程序用户指南

Added by 陈军, last edited by 陈军 on Mar 24, 2017

- 1. UFCOM 简介
- 2. UFCOM 与虚拟串口设备管理
 - 2.1. UFCOM 安装与卸载
 - 2.2. UFCOM 工作行为配置
 - 2.2.1. VCOM 生命期模式(Lifemode)
 - 2.2.2. 修改 VCOM 串口号
 - 2.2.3. 获知 VCOM 当前被哪个程序打开
 - 2.2.4. 清除不再使用的 VCOM 设备，释放串口号
 - 2.2.5. 串口号绑定模式(COM Port Bindmode)
 - 2.3. 故障诊断
 - 2.3.1. 驱动包安装过程故障
 - 2.3.2. VCOM 设备无法生成
 - 2.3.3. USB 数据传输异常诊断
- 3. VCOM 设备 API 行为说明
- 4. 本文档更新历史

本文档适用于 UFCOM 1.2.0 版本。

1. UFCOM 简介

UFCOM(USB Flexible COM Port 的缩写) 是由新大陆自动识别公司开发的全新的虚拟串口驱动程序，2017 年起向我们的客户提供，它配合虚拟串口模式的 USB 扫描枪使用，提供和 USB 扫描枪之间的双向通信功能。

UFCOM 提供的功能和 Windows 系统自带的通用虚拟串口驱动 `usbser.sys` 类似，可完全替代 `usbser.sys`，不但修正了 `usbser.sys` 的很多糟糕行为，还提供了更强大的功能。UFCOM 的功能亮点如下：

1. 多扫描枪支持。

支持多只 USB 设备（下文用“扫描枪”指代）同时工作，至少同时支持 8 只，每只扫描枪会在系统中生成一个虚拟串口设备（称 VCOM），各自对应不同的串口号。

2. VCOM Lifemode（虚拟串口生命期模式）。

通过设定 VCOM Lifemode，用户可以控制扫描枪被拔除后，VCOM 设备是否继续存在。在很多应用场合下，我们希望扫描枪意外拔除时 VCOM 能够持久存在，扫描枪插回后自动关联到原先的 VCOM，这样，使用 VCOM 的应用程序就不会因为 VCOM 句柄失效而停止工作。

3. Port number Bindmode（串口号绑定模式）。

通过设定 Bindmode，用户可以指定不同型号、不同序列号的扫描枪是关联到同一个串口号还是不同的串口号。共有 4 种绑定模式可选。

4. 动态设备显示名。

在 Windows 设备管理器中，VCOM 设备的名称会依据设备状态动态显示。比如正常工作的扫描枪，设备名称会有 [online] 前缀，拔除的扫描枪，设备名称会有 [offline] 前缀。

5. VCOM Settings 用户界面。

设备管理器中的 VCOM 设备属性对话框，提供了专门的用户界面来方便地设置 VCOM 的各项属性，可在此查看与修改 Lifemode, Bindmode 以及指定 VCOM 使用的串口号，检查 USB 链路故障等。

针对新大陆扫描枪老用户的一些背景信息

在 UFCOM 之前，我们给用户提供了两套虚拟串口驱动程序。

- USB Datapipe 驱动包: 当扫描枪通信模式被设为 USB Datapipe 或 com0com 虚拟串口时，你需要安装这个。
- CDC Virtual COM Driver: 当扫描枪通信模式被设为 USB-CDC 模式、且 PC 端是 Windows 8.1 或更早版本时，你需要安装这个。此驱动包实质上是引用了 Windows 系统上自带的 usbser.sys，对，仅仅是引用，这意味着我们无法跨大版本升级 usbser.sys，比如，无法将 Windows 10 的 usbser.sys 用到 Windows 7 上。

现在 UFCOM 可以完全替代那两套老的驱动程序，即，不论你将扫描枪设成三种“虚拟串口”模式的哪一种，UFCOM 都将为它们生成 Windows 上的虚拟串口。

UFCOM 同时提供了 Datapipe API 支持，意即，先前使用 udp_op.dll 的老式应用程序也可以配合 UFCOM 工作。注意：部分使用 Datapipe API 的老版本应用程序有 bug，需要更新后方可配合 UFCOM。

2. UFCOM 与虚拟串口设备管理

2.1. UFCOM 安装与卸载

【背景知识】

为了让我们的用户对于 Windows 系统的驱动程序安装有个清晰的认识，这里介绍一些关键知识。所谓的“安装驱动程序”其实包含了三个步骤：

1. 将驱动包(driver package)安装到称作 DriverStore 的系统目录中。可以认为这是个单纯的文件拷贝动作。在 Windows 7~10 上，该目录是 C:\Windows\System32\DriverStore\FileRepository。Driver package 的内容，通常由 inf, sys, cat 构成，也可能包含一些 dll 和 exe。其中的 inf 是关键，可以认为一个 inf 就代表了一个驱动包。
2. Windows 为探测到的硬件设备（也包括虚拟的软件设备）匹配驱动程序。意思是：一个设备被 Windows 探测到后，会向 Windows 通告自己的 hardware-id，Windows 在 DriverStore 中检索各个 .inf 文件的内容，看有没有哪些 inf 宣称自己匹配这个 hardware-id。有可能多个 inf 都能匹配当前的 hardware-id，但 Windows 同一时刻只会选择其中的一个来驱动当前这个设备。Windows 通常会选择最佳的那个，比如有数字签名的，且日期最新的那个。
3. 装载驱动程序代码(.sys)到内存中执行。硬件设备至此真正地开始和 Windows 交换数据。

当 Windows 为一个具体设备匹配了一个驱动包后，该匹配关系被 Windows 记录于注册表中（记录于该设备的 hardware-key 下方），即使设备拔除(Unplug)也不会被删除。当设备第二次插入时，Windows 就无需重新搜索 DriverStore 进行驱动包匹配，而是直接装载之前记录的 .sys 开始执行。只有当用户在设备管理器中对一个设备实例进行卸载(Uninstall)操作后，设备实例信息从注册表中删除（hardware-key 被删除），此后再插入设备就会重新触发驱动包搜索与匹配动作。

【UFCOM 安装过程】

UFCOM 安装包目前以 zip 方式提供，用户解压 zip 后看到一个 install.bat，双击执行之即可进行 UFCOM 安装动作。Install.bat 内部调用微软提供的安装工具 DPInst.exe 来达成目的。Install.bat 具体做了如下事情：

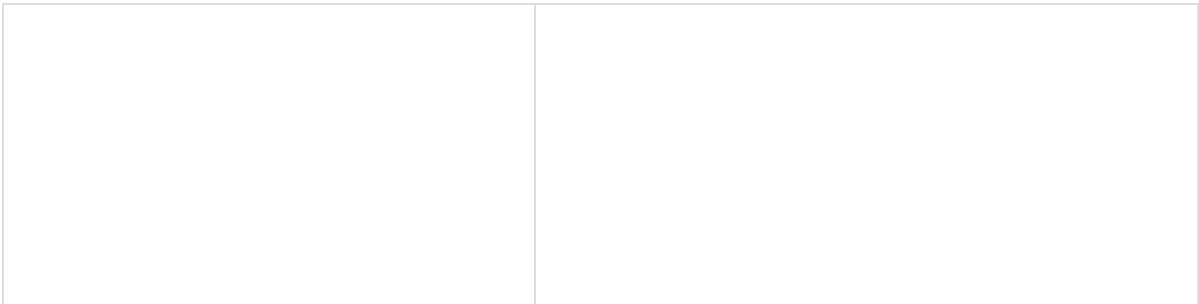
- 将 UFCOM driver package 拷贝到 DriverStore 中。
- 如果当前 Windows 上已经存在有能够匹配 UFCOM 驱动包的设备实例（包括当前插入的和拔除的），这些设备实例所匹配的驱动包将会被更新为使用 UFCOM 驱动包。
- 如果旧版本的驱动代码(.sys)无法从内存中被卸载（比如设备串口正被打开着），Windows 会提示用户重启整个系统。

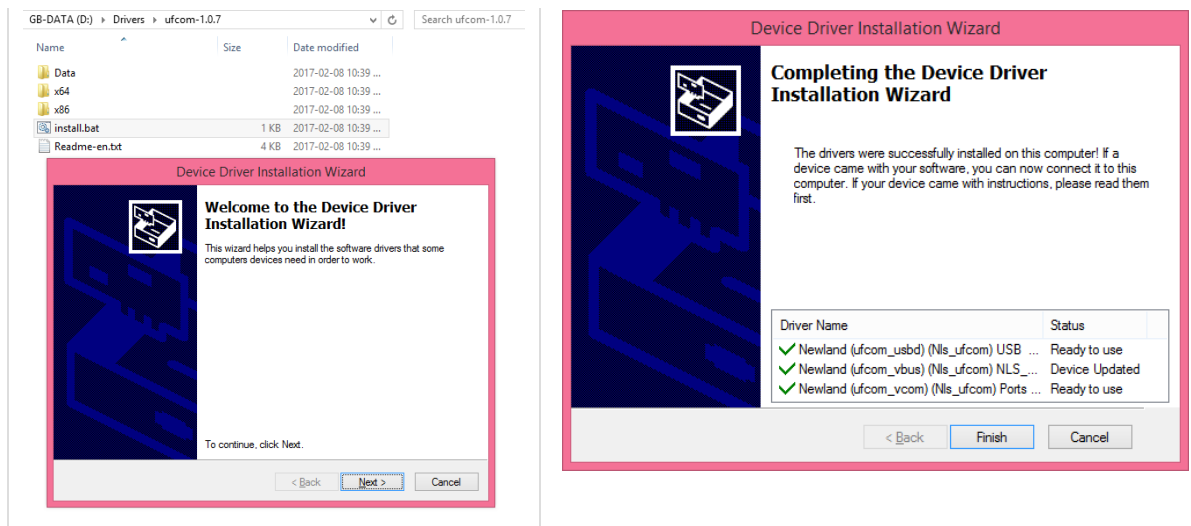
对于首次执行安装包的情况，不用担心插入扫描枪和执行安装包的先后顺序，两者皆可。

对于升级安装，我们建议关闭所有使用操作扫描枪的应用程序后再执行安装包（不必拔除扫描枪），这样可以最大程度减小需要重启 Windows 的概率。

安装向导对话框点击 Next 按钮，等待一会儿，安装向导将报告安装结果。如果看到三个绿色的勾，表示安装成功。

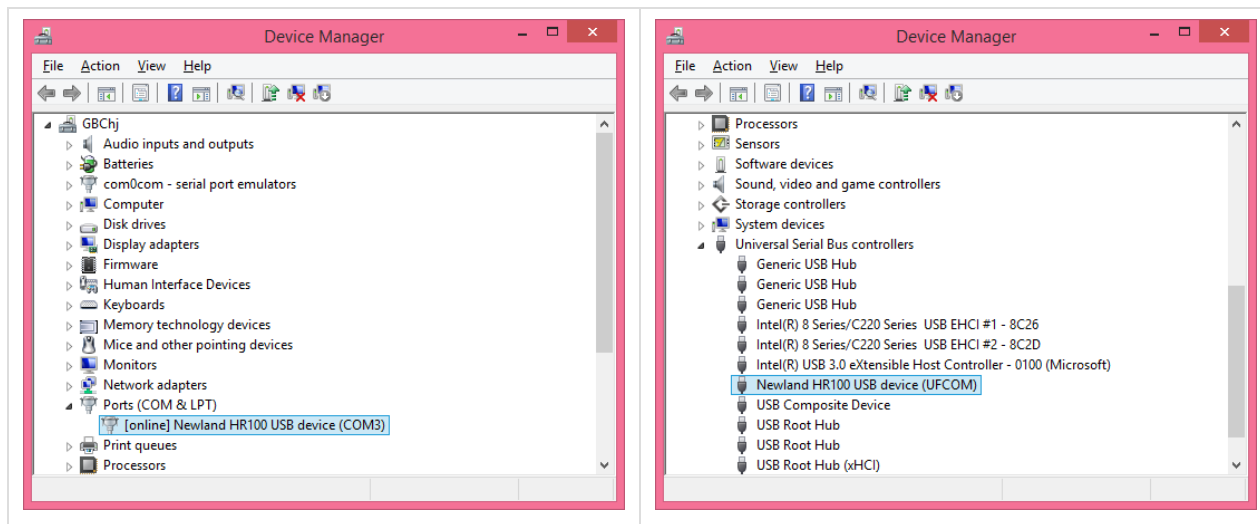
如果 Windows 提示需要重启，请重启 Windows 以保证安装/升级完成。





现在插入 **USB 扫描枪**（如果你还没这么做的话），**Windows** 将为扫描枪匹配并装载驱动程序。第一次插入扫描枪，匹配驱动程序可能会花费几秒至几十秒时间（依 **Windows** 运行速度而异）。

为了确认扫描枪被成功地驱动，需要打开设备管理器进行验证。在设备管理器中，我们实际上会看到两个新设备生成。其中一个虚拟串口（左图，称 **VCOM**），另一个是真实的 **USB 设备**（右图，暂称其物理设备）。

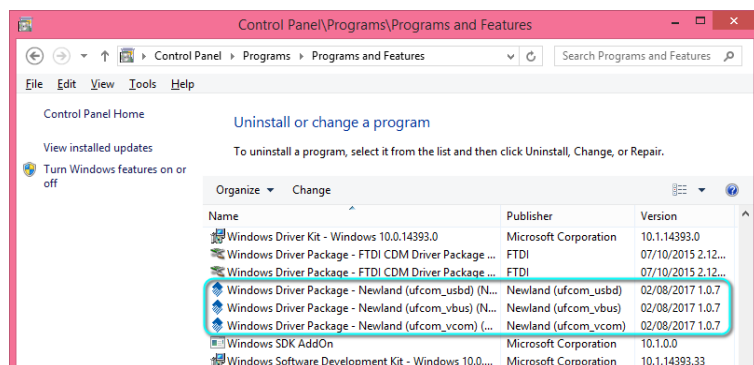


如果将扫描枪拔除，物理设备必定会消失(处于 **unplugged** 状态)，但 **VCOM** 设备不一定会消失。**VCOM** 是否跟随物理设备消失，由 **VCOM Lifemode** 来决定，用户可以自行设定 **VCOM Lifemode**。

【卸载 UFCOM】

在控制面板中用户可以对 **UFCOM** 进行卸载。

每一个 **UFCOM** 版本被安装后，会在 **Programs and Feature** 中生成三个项目，名字中带有 **ufcom** 字样。如下图：



如果需要，可以将它们逐个卸载。此处的卸载动作意味着：

- 驱动包从 **DriverStore** 中删除。
- 当前匹配 **UFCOM** 驱动包的所有设备节点将重新进行驱动包匹配，也许匹配到 **Windows** 自带的一个驱动包或一个 **UFCOM** 老版本的驱动包。

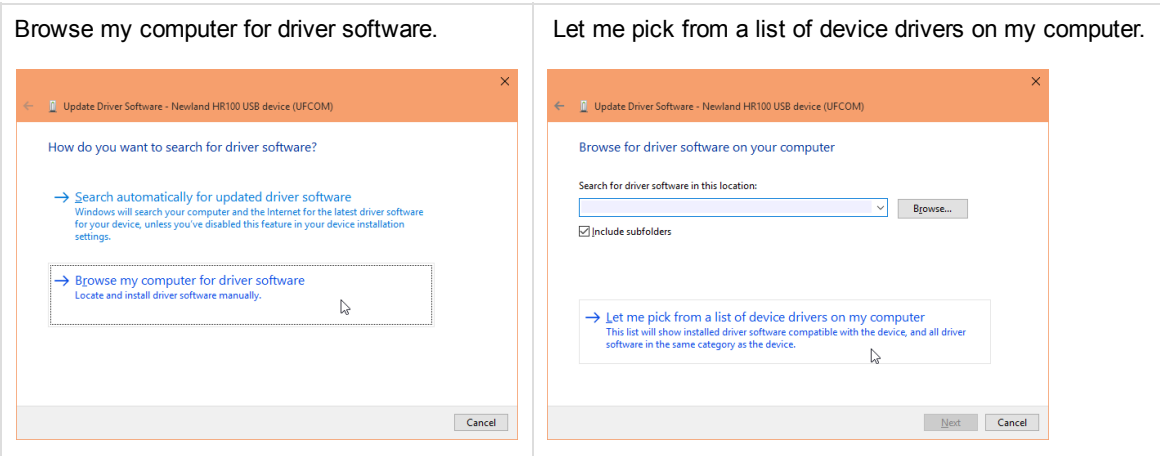
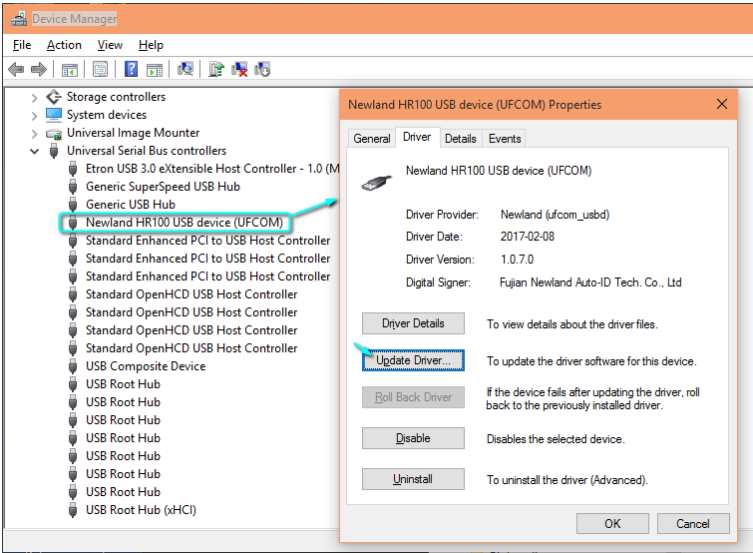
注：每安装一个新版本的 **UFCOM** 驱动包，并不会自动卸载老版本驱动包，老版本驱动包需要手工卸载。但这通常不是大问题，因为 **install.bat** 安装过程 **Windows** 会帮我们匹配最新版的驱动包。

【 为 USB 设备手动切换驱动程序包 】

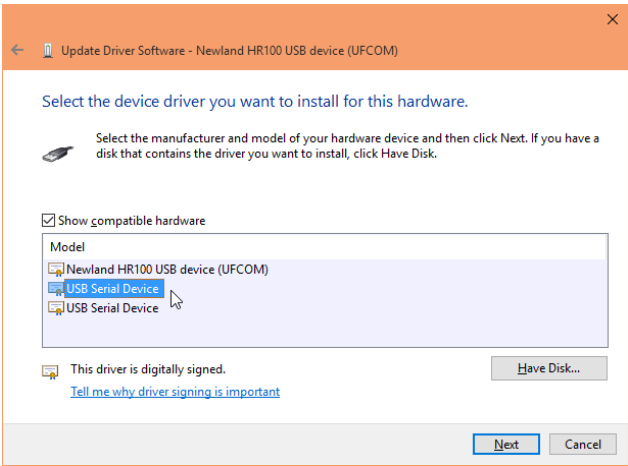
如果因为某种原因你需要针对某个 **USB** 设备使用老式的 **Datapipe** 驱动、老版本的 **UFCOM**，或更换为 **Windows** 自带的 **usbser.sys**，你不需要卸载 **UFCOM** 驱动包，只需要针对具体设备更改所装载的驱动程序即可，这是 **Windows** 自身提供的功能。

以 **Windows 10** 为例，操作方法如下：

在设备管理器中找到真实的物理设备节点（特别提示：非 **VCOM** 设备节点），打开其属性对话框，点击 **Update Driver** 。



在 **Show compatible hardware** 列表框中，选择一个驱动包。此处，驱动包是用 **inf** 中的设备名字来标识的。本例中，**Newland HR100 USB device(UFCOM)** 代表 **UFCOM** 驱动包，**USB Serial Device** 是 **Windows** 自带的 **usbser.sys** 驱动包。



最后点击一次 **Next**，确定使用选中的驱动包。

当然，你可以用同样的方法让 USB 设备换回 UFCOM 驱动包。

2.2. UFCOM 工作行为配置

对于 UFCOM 生成的 VCOM，你可以用任何的串口通信工具打开它，通过标准的 Windows 串口操作 API 来读写串口数据，达到跟 USB 设备通信的目的。常用的串口工具如 Windows XP 自带的超级终端，Windows 7 上没有超级终端了，可以用免费的 PuTTY。如果需要对扫描枪进行高级设置，可以用新大陆提供的 EasySet。

由于 Windows 的串口操作 API 最初是贴合 RS-232 物理串口设计的，它没有考虑到 USB 设备特有的行为（比如可以被随时拔除插入、节能管理等），这些行为无法用串口 API 来控制，因此，UFCOM 提供了专门的手段来控制这些特有的行为。

2.2.1. VCOM 生命期模式(Lifemode)

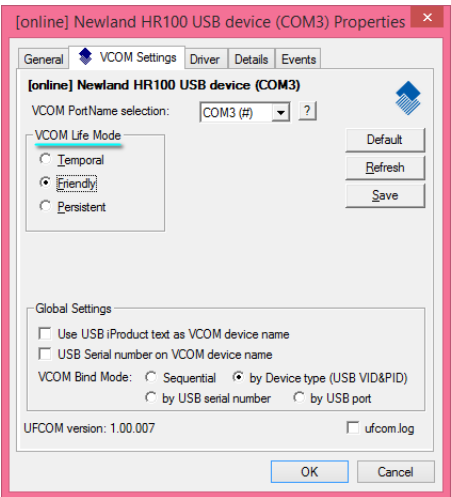
VCOM Lifemode 指出 VCOM 设备什么情况下存在，什么情况下消失。此处，“消失”的意思是，在设备管理器中无法看到 VCOM 设备（除非要求设备管理器显示隐藏的设备）。

Lifemode 名称	VCOM 显示名称前缀	VCOM 存在的条件
短暂模式 (Temporal)	<online>	扫描枪插入时 VCOM 出现，扫描枪拔除后 VCOM 立即消失。
友好模式 (Friendly)，此为默认设定	[online]	扫描枪插入时 VCOM 出现。
	[offline]	扫描枪拔除时，若已有应用程序正在使用这个 VCOM（应用程序持有该 VCOM 的打开句柄），VCOM 不会消失，原先的句柄也仍旧有效。在 VCOM 句柄保持打开的情况下，若扫描枪被插回，则扫描枪恢复和原先 VCOM 的连通。意即，获取了 VCOM 句柄的应用程序，不会受到物理扫描枪短暂拔除的影响，达到一种“自动故障恢复”的效果。 当“扫描枪拔除”和“其对应的 VCOM 句柄被关闭”这两个条件同时成立时，VCOM 消失。
持久模式 (Persistent)	[[online]]	VCOM 永远存在。
	[[offline]]	持久模式使得 VCOM 像一个物理串口般永远存在于系统中，不论扫描枪是否插入系统。

VCOM 设备显示名称 online, offline 前缀的含义：

- online 表示扫描枪当前插在 Windows 机器上且驱动程序工作正常。
- offline 表示扫描枪已被拔除(physically unplugged)。

设置 Lifemode 的操作方法：在设备管理器找到 VCOM 设备节点（不是找物理设备节点），双击打开设备属性对话框。切换到 "VCOM Settings" 选项卡。此处你可以调整 VCOM 的 Lifemode。

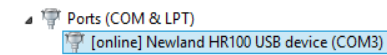


特别提示：插入两把扫描枪将会生成两个 VCOM 设备节点，它们的 VCOM Lifemode 是各自独立的。

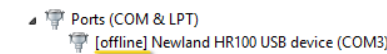
你需要 Windows 系统的管理员权限方可修改 VCOM Settings。

【观察 VCOM 设备 "online/offline" 显示名前缀的变化】

扫描枪插入 Windows ，对应的 VCOM 设备在设备管理器中将被附上前缀 online 。



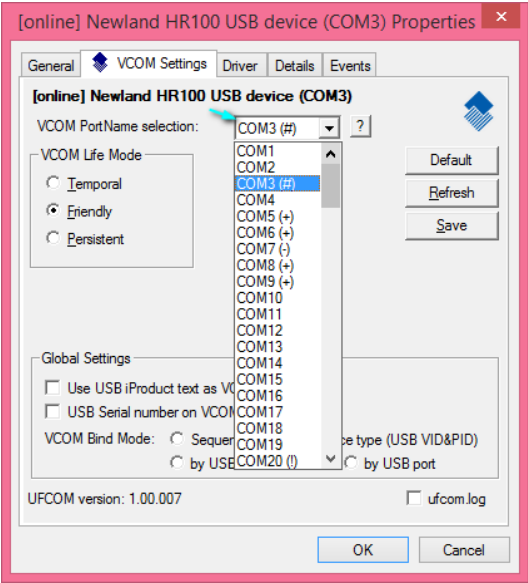
扫描枪拔出，且根据 VCOM Lifemode 规则判定 VCOM 设备应继续存在的情况下，前缀变为 offline 。



2.2.2. 修改 VCOM 串口号

这是一个常用功能。

当 Windows 生成一个新的 VCOM 设备时，系统会为 VCOM 自动选取一个空闲的串口号，通常是分配一个数值最小（从 COM3 开始）且未占用的。当我们对系统自动分配的串口号不满意时，可以通过 VCOM Settings UI 修改它。



在选用新的串口号前，请留意下拉列表中各个备选串口号尾部的 + - ! 后缀，它们有如下含义：

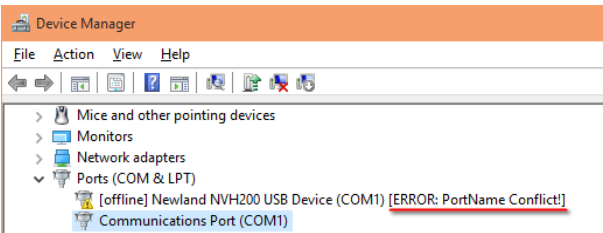
符号	含义
#	此串口号正在被自己使用。
!	此串口号正在被其他串口设备使用，这些串口设备当前真实存在于系统中（且是 plugged-in 状态）。
+	有其他某个 VCOM 设备已使用了此串口号，只不过这些 VCOM 设备处于 unplugged 状态。
-	Windows 的 ComDB 数据库标记该串口号已被某个串口设备使用，这些串口设备暂处于 unplugged 状态。

不必记忆那些符号的含义，点击下拉框右侧的小问号即会得到解释。

我们建议新选用的串口号是不带任何后缀标记的(表示空闲可用)，以防止干扰其他串口设备的工作。

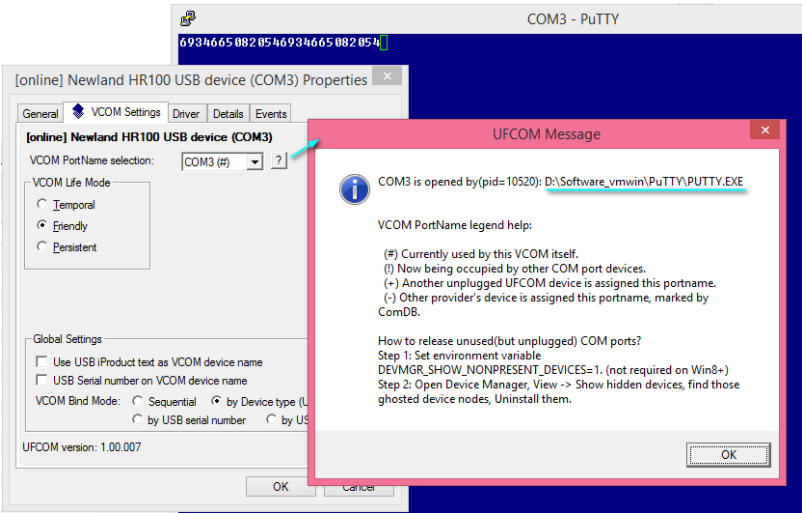
- 如果你选择 (!) 后缀的串口号，必定会遭遇失败。
- 选择 (+) (-) 后缀的串口号是可以的，比如图中的 COM7 ；只不过，当原先使用 COM7 的那个设备被插回时，原先那个设备可能会工作异常、也可能它会另行选取一个新串口号后正常工作，具体行为由该设备的驱动程序决定。

如果选择了 (!) 标注的串口号，UFCOM 会让该设备节点报错，出错原因会显示在设备名称上。比如，在物理串口 COM1 存在的情况下，强行将 VCOM 串口号设为 COM1，设备管理器将提示 [ERROR: PortName Conflict!] 如图：



2.2.3. 获知 VCOM 当前被哪个程序打开

点击 PortName selection 右侧的小问号，我们能得知当前 VCOM 是否被某个程序打开着。如下图，告知 COM3 正被 PUTTY.EXE 打开。

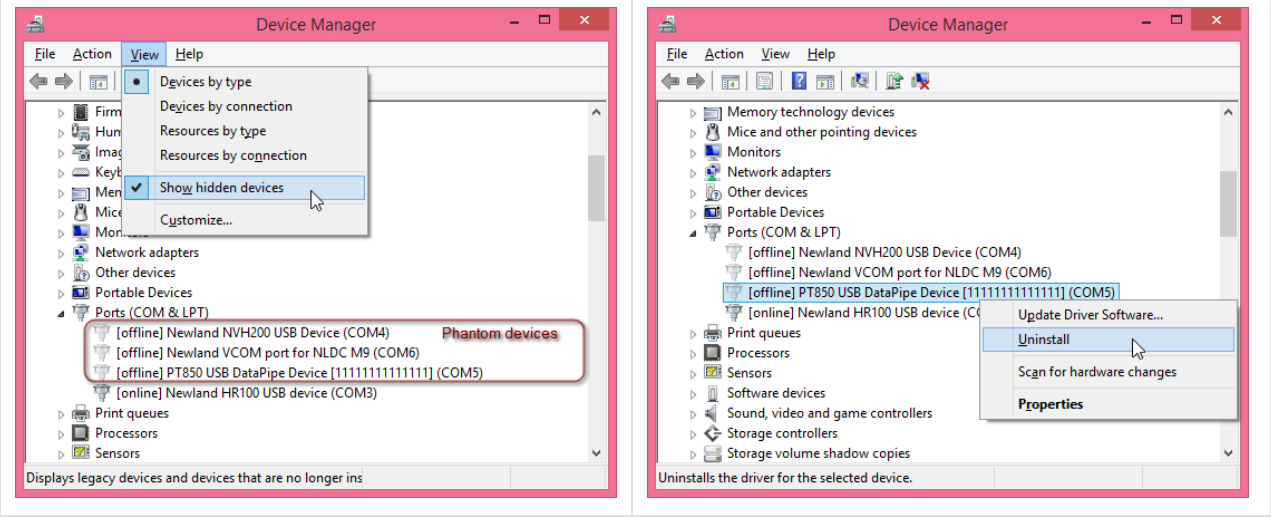


2.2.4. 清除不再使用的 VCOM 设备，释放串口号

当我们拔除 USB 扫描枪、且关闭其 VCOM 设备句柄后，我们会看到扫描枪对应的 VCOM 设备节点会从设备管理器中消失（除非 Lifemode=Persistent），但事实上，这种消失更像是隐藏——因为 VCOM 设备节点的信息还存在于注册表中，这些信息包括：此 VCOM 的串口号、Lifemode 等。当“同一个”VCOM 设备实例再次出现时，Windows 就能记得这个 VCOM 之前用的是哪个串口号、哪种 Lifemode，并应用那些设定，这是非常合理的行为。（注：VCOM 设备实例是用所谓的“设备实例路径(Device instance path)”字串来标识的，设备属性 Details 选项卡能够查到当前设备的这个字串）

当我们往系统中插入多把不同型号的扫描枪后，可能会有多个串口号被系统标识为“已分配”。若我们不再使用之前插过的扫描枪、想回收那些串口号，即，让系统重新认为它们是空闲串口号，我们得将将占用那些串口号的设备从系统中卸载。

操作方法是：让设备管理器显示隐藏设备 View → Show hidden devices。此时我们将看到之前插入但现在处于 unplugged 状态的设备（左图）。这种设备被称为影子设备(phantom device)，它的设备图标显示成残影的样子。对影子设备执行卸载动作，可回收它们占用的串口号（右图）。



提示：在 Windows XP~Windows 7 系统上，为了看到影子设备，你得事先设置环境变量 DEVMGR_SHOW_NONPRESENT_DEVICES=1 再打开设备管理器。Windows 8.0 以上的系统无需此额外步骤。

2.2.5. 串口号绑定模式(COM Port Bindmode)

串口号绑定模式是一个全局设定，它会影响系统给 USB 扫描枪分配串口号的规则。

UFCOM 提供了四种绑定模式供选择，你需要发挥一些想象力来理解它。

绑定模式	含义
------	----

顺序绑定 (Sequential)	<p>UFCOM 对插入系统的所有扫描枪都不作差别对待，最终效果是，串口号是按顺序分配的。</p> <p>举例，我们手头有两只扫描枪，插入第一只得 COM3，同时插入第二只得到 COM4。将两只扫描枪都拔除，之后无论先插入哪一只，都会先得到 COM3。</p>
按型号绑定 (by Device type)，默认值	<p>简言之，UFCOM 会关心扫描枪的型号，不同型号的扫描枪会分配到不同的串口号。扫描枪的型号是用 USB 设备描述符中的 VID,PID 二元组标识的。</p> <p>举例，我们手头有一只 HR100 和一只 HR200（它们是不同的型号），第一次插入 HR100 时得到 COM5，第一次插入 HR200 得到 COM6。之后，插入 HR200 将总是得到 COM6——不论 HR100 是否已插入；同理，插入 HR100 总是得到 COM5。意即，COM5 被绑定到 HR100，COM6 被绑定到 HR200；这种绑定关系一直持续到你将相应的 VCOM 设备卸载为止。</p>
按序列号绑定 (by USB serial number)	<p>不同序列号的扫描枪将得到各自不同的串口号。USB 扫描枪的序列号是由 USB 设备描述符中通告的的序列号 (iSerialNumber 字符串)决定的。</p> <p>当我们需要让同型号的两把扫描枪绑定到不同的 VCOM 时，可以用此设定。但请注意，有些型号的扫描枪自身并未烧录序列号，这些扫描枪就只能回退至按型号绑定的行为。</p>
按 USB 端口绑定 (by USB port)	<p>插入不同 USB 端口的设备将得到不同的串口号。先后插入同一个 USB 端口的不同设备，不论设备型号，都将得到相同的串口号。</p> <p>此绑定模式在 Windows 7 或更新版本上可用。</p>

额外提示：

- 目前，切换绑定模式会使得当前已经打开的 VCOM 句柄失效（无论何种 Lifemode），应用程序必须重新打开它们。
- 切换绑定模式后，在前一种模式时已经保留的串口号并不会被自动回收。这意味着，对于同一支扫描枪，在不同的绑定模式下它将得到不同的串口号——除非你手动将它们设成相同的。

2.3. 故障诊断

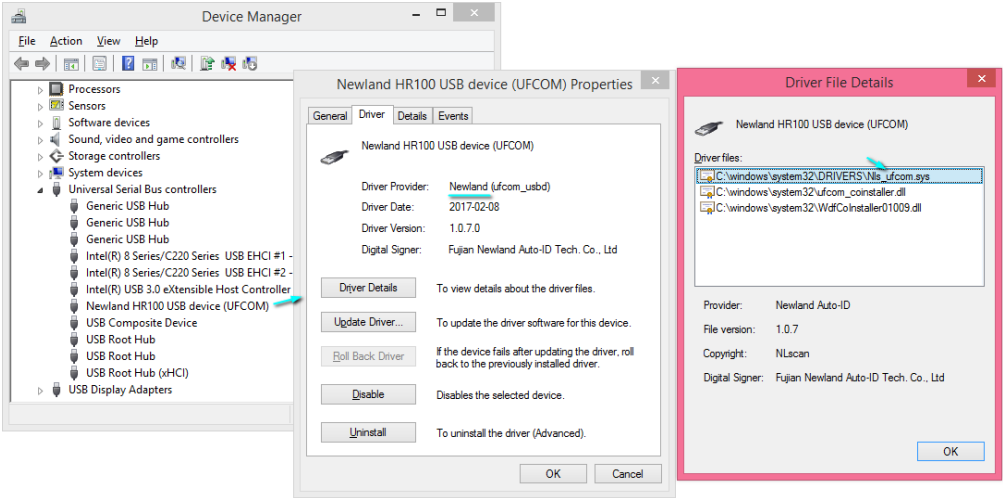
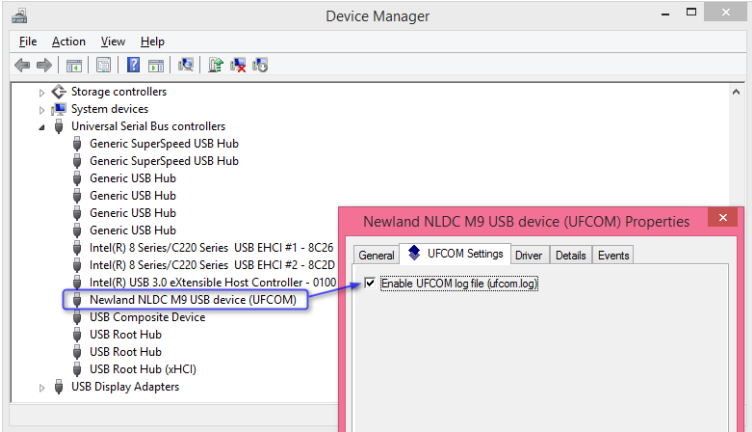
2.3.1. 驱动包安装过程故障

暂无。

2.3.2. VCOM 设备无法生成

有多种情况可能会导致 VCOM 设备无法生成

故障分类	现象、原因与解决方法
USB 物理设备没有被 Windows 检测到	<p>现象：在设备管理器中，"通用串行总线控制器(Universal Serial Bus controllers)" 类别下方，没有看到新设备节点的出现，在设备管理器其他设备类别下也没有看到新设备节点出现。</p> <p>可能的原因：</p> <ul style="list-style-type: none">• 扫描枪没有插入 Windows 主机。• 扫描枪 USB 功能未开启。• 扫描枪有硬件故障。• Windows 上未正确安装 USB 系统驱动（指 USB 控制器驱动）。 <p>检测物理设备的存在是由 Windows 自身进行的，设备的控制权还未转移到 UFCOM 手上。</p>
USB 扫描枪当前为键盘模式	<p>现象：扫描枪能够工作，扫描到的条码可以显示于文本编辑器（比如记事本）中。</p> <p>这说明扫描枪的 USB 通信模式是 USB keyboard（大多数扫描枪的出厂默认）。你需要让扫描枪扫描一个特殊设置码来将其切换为 UFCOM 可识别的模式，比如 USB-CDC。</p>

故障分类	现象、原因与解决方法
Windows 检测到 USB 物理设备，但使用了其他的驱动程序	<p>现象：USB 设备显示名称不是以 "(UFCOM)" 结尾。</p> <p>双击 USB 物理设备查看 Driver Provider，正常应该显示有 Newland 字样，且 Driver Details 中有 Nls_ufcom.sys。</p>  <p>如果发现使用了其他的驱动程序（比如使用了 Windows 自带的 usbser.sys 或使用了老式的 Datapipe 驱动），请将其切换为使用 UFCOM。操作方法前头 "为 USB 设备手动切换驱动程序包" 小节。</p>
USB 物理设备已检测到，但驱动程序工作不正常	<p>现象：USB 设备图标上有黄色感叹号。</p> <p>先确认 USB 设备当前是否使用了其他的驱动程序，若是，应该先切换成使用 UFCOM。已确认使用 UFCOM 后，若仍旧出现</p> <p>可能的原因：</p> <ul style="list-style-type: none">扫描枪的固件太旧，无法配合 UFCOM 工作。请联系新大陆更新固件。USB 硬件故障，故障可能出现在 PC、连接线或扫描枪自身。USB 设备属性对话框，General 选项卡会报告简要的故障原因，请将故障原因文字发给我们进行诊断。
USB 物理设备正常，但 VCOM 设备未出现	<p>此情况比较罕见。为了帮助我们诊断原因，你可以开启 UFCOM 的日志。开启日志后，让故障重现，将日志文件发给我们诊断。</p>  <p>日志文件位置：C:\Windows\Temp\ufcom.log。</p> <p>日志文件在驱动程序每次重新装载时会清空重建（比如重启 Windows 后），因此不用担心日志文件无限增长。</p>
VCOM 设备正常展现，但应用程序和扫描枪通信不正常	<p>该问题比较复杂，见下一节描述。</p>

2.3.3. USB 数据传输异常诊断

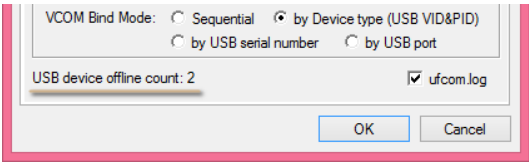
如果我们遭遇虚拟串口应用程序和 USB 外设数据通信异常，比如，虚拟串口无法读到条码，扫描枪配置程序（EasySet 等）无法检测到扫描枪，EasySet 获取扫描枪图像失败等，在寻求技术支持前，我们可以自己初步诊断故障原因。大致而言，故障可能出现在如下几个环节：

- (A) 扫描枪自身的软硬件。

- (B) USB 数据传输线路。
- (C) PC 端的 USB 硬件和 USB 主控(host controller)驱动程序（由 Windows 提供或 Host Controller 厂商提供）。
- (D) PC 端的 USB client driver（指 UFCOM）。
- (E) 使用 VCOM 的应用程序自身。

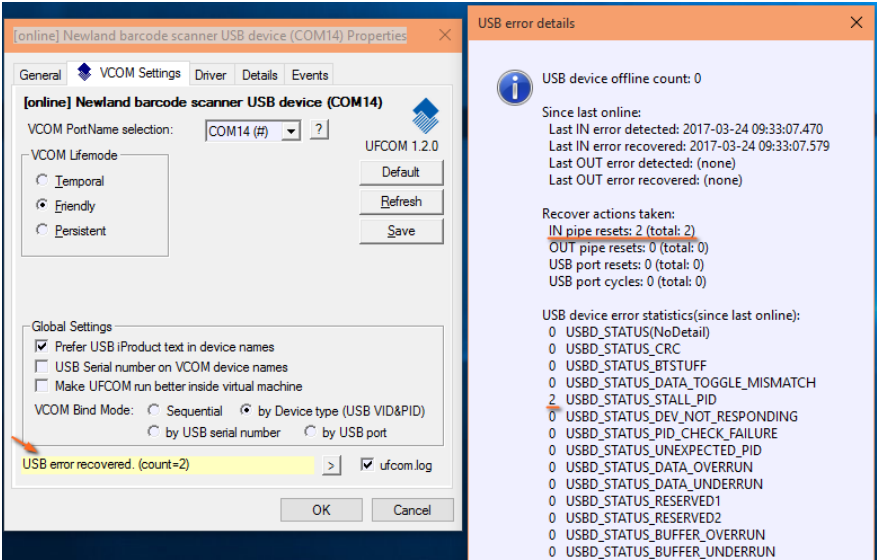
对于普通用户来说，要分辨故障出在哪个环节比较困难。幸运的一点是，UFCOM 可以帮助你直观地判断 B 环节是否有故障。当使用不良的 USB 线缆、有 bug 的 USB hub、USB 端口供电不足、或是 USB 信号受到干扰时，B 环节将暴露出故障。

VCOM Settings UI 的底部状态栏会告知 VCOM 设备对象存在期间发生设备 offline 的次数。当然，我们人为拔除扫描枪将导致 offline 计数增加，但如果我们正常使用过程却发生了 offline 增加的情况，说明 USB 物理信号传输环节很可能有故障。



还有些情况下，USB 传输故障并不表现为 offline，而是表现为 USB 主控硬件以软件形式向 Windows 报告一个 UsbdStatus 错误，Windows 进而将该错误告知 UFCOM，UFCOM 会将这种错误通过 VCOM Settings UI 呈现给最终用户。在一个工作良好的系统中，我们不期望看到任何的 UsbdStatus 错误。如果我们发现了这种错误，则暗示 USB 线路存在不稳定的情况。偶发的此类错误可以被软件清除，使得最终用户感知不到传输过程被打断，但如果 UsbdStatus 错误发生很频繁，意味着着传输系统非常不可靠，比较大的可能是硬件不稳定导致的，需要进一步排查。

当 UFCOM 检测到 UsbdStatus 错误时，VCOM Settings UI 会将其呈现出来。如下图，UFCOM 报告检测到了 2 个 错误，并且已经自动恢复。点击黄色状态条右侧的小箭头，可查看详细信息，得知发生了两次 USBD_STATUS_STALL_PID 错误。



注意：在 USB 线路质量很差的情况下，即使 UFCOM 提示错误已经恢复，也不能完全保证 USB 后续通信功能正常。换言之，对于偶发的错误效果才比较好。

提示：VCOM 的 USB 错误统计数字是各个 VCOM 独立的。一个 VCOM 的统计数字在 VCOM 被“拔除”（从设备管理器中消失）后清零。

如果仍旧无法定位故障，请联系我们解决。

3. VCOM 设备 API 行为说明

UFCOM 较为完整的实现了 Windows 串口 API 的功能。

如果你是 Windows 串口通信程序员，你需要了解本节的信息。

功能类别	实现说明
RS-232 通信参数模拟	不作模拟。用户可设置任意波特率，但设置值被忽略，和扫描枪的通信速度按 USB 线路能够达到的最大速度进行。 数据位、停止位、奇偶校验位的设置被忽略。

功能类别	实现说明										
Modem 信号线模拟	<p>输出信号：RTS, DTR 的设置被忽略。</p> <p>输入信号：GetCommModemStatus 总是报告 RING 和 RLSD 为 off 。DSR 和 CTS 信号则进行了模拟。</p> <p>DSR: 当扫描枪插入时，DSR=On ； 扫描枪拔出后，DSR=Off 。</p> <p>CTS: 当扫描枪已插入、且下行数据流未受阻时，CTS=On ； 扫描枪拔出或下行数据流受阻，CTS=Off 。</p>										
硬件流控行为	<p>不支持用模拟的 modem 信号进行上行流控（输入流控），上行流控完全由 USB 线路自身的流控功能决定。</p> <p>支持下行流控（输出流控）。VCOM 被应用程序打开后，默认不启用流控；当应用程序以 DCB.fOutxCtsFlow=1 调用 SetCommState() 后，下行流控开始起作用，意即， CTS=off 时会使得 WriteFile 暂时阻塞。</p>										
软件流控行为	不支持 XON/XOFF 方式的软件流控。										
WriteFile 行为	UFCOM 保证 WriteFile 的原子写入行为。即，对于应用层请求写出的数据，WriteFile 或者全部接受，或者一个字节也不接受。										
串口超时	<p>完整实现了 COMMTIMEOUTS 超时参数的语义。针对 USB 设备的特点有一些特殊的设计：</p> <ul style="list-style-type: none">• 当 WriteTotalTimeoutMultiplier=0，WriteTotalTimeoutConstant=0，且未启用输出流控的情况下，UFCOM 保证 WriteFile 不会永久阻塞。若扫描枪被拔除、或下行数据被抑制，WriteFile 会在持续 1 秒无法写出数据后返回，并告知 WriteFile 成功，此举是为了模拟向物理串口 WriteFile 不会永久阻塞的效果。• 当 WriteTotalTimeoutMultiplier=0，WriteTotalTimeoutConstant=0，且启用输出流控的情况下，将所有数据写出前，WriteFile 会永久阻塞。当然，用户可以用 Canceled 来取消 WriteFile，Canceled 后 WriteFile 会立即完成。• 当 WriteTotalTimeoutMultiplier 或 WriteTotalTimeoutConstant 任意一者非 0 时，不管是否启用下行流控，用户指定的超时将生效。										
串口事件等待	<p>实现了如下四种事件的通知：</p> <table><tr><th>事件</th><th>触发条件</th></tr><tr><td>EV_RXCHAR</td><td>当输入缓冲区中有字符时，发生该事件，是电平触发的，意即，只要没把输入缓冲区内容读空，WaitCommEvent 会一直报告这个事件。</td></tr><tr><td>EV_TXEMPTY</td><td>当输出缓冲区变空时，发生该事件，是边缘触发的，意即，当应用程序得到一次 EV_TXEMPTY 通知后，立即再次等待 EV_TXEMPTY 事件，将不会再次得到通知，要经历过一次输出缓冲区填入内容（WriteFile）再排空的过程，才会再次通知 EV_TXEMPTY 。</td></tr><tr><td>EV_DSR</td><td>EV_DSR 是个边缘触发事件，反映 DSR 信号的变化。</td></tr><tr><td>EV_CTS</td><td>EV_CTS 也是个边缘触发事件，反映 CTS 信号的变化。</td></tr></table> <p>另，UFCOM 允许用户使用两个线程进行操作，一个线程调用 WaitCommEvent，另一个线程进行 ReadFile，WriteFile 读写数据。</p>	事件	触发条件	EV_RXCHAR	当输入缓冲区中有字符时，发生该事件，是电平触发的，意即，只要没把输入缓冲区内容读空，WaitCommEvent 会一直报告这个事件。	EV_TXEMPTY	当输出缓冲区变空时，发生该事件，是边缘触发的，意即，当应用程序得到一次 EV_TXEMPTY 通知后，立即再次等待 EV_TXEMPTY 事件，将不会再次得到通知，要经历过一次输出缓冲区填入内容（WriteFile）再排空的过程，才会再次通知 EV_TXEMPTY 。	EV_DSR	EV_DSR 是个边缘触发事件，反映 DSR 信号的变化。	EV_CTS	EV_CTS 也是个边缘触发事件，反映 CTS 信号的变化。
事件	触发条件										
EV_RXCHAR	当输入缓冲区中有字符时，发生该事件，是电平触发的，意即，只要没把输入缓冲区内容读空，WaitCommEvent 会一直报告这个事件。										
EV_TXEMPTY	当输出缓冲区变空时，发生该事件，是边缘触发的，意即，当应用程序得到一次 EV_TXEMPTY 通知后，立即再次等待 EV_TXEMPTY 事件，将不会再次得到通知，要经历过一次输出缓冲区填入内容（WriteFile）再排空的过程，才会再次通知 EV_TXEMPTY 。										
EV_DSR	EV_DSR 是个边缘触发事件，反映 DSR 信号的变化。										
EV_CTS	EV_CTS 也是个边缘触发事件，反映 CTS 信号的变化。										
Canceled	<p>完全支持。用户可用 Canceled 取消进行中的 ReadFile，WriteFile，WaitCommEvent，Canceled 将立即成功。</p> <p>特别提示：针对 ReadFile 发起 Canceled 时，若 VCOM 已经收取了部分的字节，ReadFile 将报告成功，用户能取到已收取的字节。此行为比 Windows 自带的 serial.sys 和 usbser.sys 优秀；后者对于 Canceled 的情况，只会报告结果为 995(ERROR_OPERATION_ABORTED)。</p>										

【针对 Temporal Lifemode 的特别说明】

对于 Temporal Lifemode 的 VCOM，对应的 USB 物理设备拔除将使得 VCOM 句柄处于失效状态，你用失效句柄再次进行 API 调用，ReadFile、WriteFile、WaitCommEvent 等，都将立即返回失败，即使重新插回扫描枪，失效句柄仍旧保持失效。此时你能做的只有关闭失效句柄，再调用 CreateFile 重新获得一个 VCOM 句柄。

4. 本文档更新历史

UFCOM 版本	文档更新内容
1.2.0	USB 故障自动检测和恢复。
1.1.0	USB device 设备属性对话框中可开启 ufcom.log。支持 Datapipe API。

UFCOM 版本	文档更新内容
1.0.7	首个版本。