

CH372 动态库说明文档 V1.0

库文件必须放在系统库文件夹内（/usr/local/lib）

Windows 系统下，设备操作以 Index 或者 Handle 的方式进行，而在 MAC OS 下，设备则是以 locationID 参数来进行操作。在苹果系统下，CH372 设备的区分是以位置来区分的，同一个 USB 接口位置下的 CH372 设备的地址位置即 locationID 是一样的。CH375GetDeviceLocation 这个函数可以用来获取计算机上存在的所有 CH372 的设备位置信息。其他操作与 Windows 下基本类似。

int CH375GetDeviceLocation(UINT32** locationID)此函数返回值表示 CH375 设备的个数，locationID 则返回一段 UINT32 的数组内存，存贮了设备的位置信息，此段内存需要自己调用 free 来释放。操作可见下图

```
UINT32* locationID = NULL;
int nCount = CH375GetDeviceLocation(&locationID);
NSLog(@"found %d device.\n", nCount);
if (nCount != 0) {
    m_locationID = locationID[0];
    free(locationID);
}
```

Bool CH375SetDeviceNotify(pCH375NotifyRoutine routine) 此函数用来设置设备插拔通知，routine 的申明必须是以 void __stdcall functionName(int32_t parameter)的方式。当回调函数被调用时，parameter 的值既是设备的位置信息(locationID)，如果是正数，则代表设备插上，如果为负数则代表设备被拔除。

```
CH375SetDeviceNotify(NotifyRoutine);
```

```
void __stdcall NotifyRoutine(int32_t nEvent)
{
    NSLog(@"notifyRoutine %d.\n", nEvent);
    if (nEvent > 0) {
        m_locationID = nEvent;
    } else {
        m_locationID = 0;
    }
}
```

Windows 下 CH375SetExclusive 这个函数在 MAC 下作废，在 MAC 下，所有的操作都是以独占的方式进行的，一旦打开成功以后，其他程序无法打开此设备进行操作。