

# aaudio 范例: 高精度计时

```
//高精度计时
import console;
import time.performance;
import time.period;

/*
计时器有关的函数，
参数虽以毫秒为单位或返回毫秒值，但实际上并非真的以毫秒为精度。
存在十几毫秒的误差是正常的。

对于分时操作限制，这种设计本身是合理而且必要的。
默认允许一定的计时精度误差有利于提升性能的稳定性、性能并降低功耗。
任何地方都有限制，任何东西都不可能十全完美。
*/

//得到高精度计时，以毫秒为单位（千分之一秒），精度为微秒（千分之一毫秒）
var tick = time.performance.tick;
var tk = tick();

//高精度延时，占用 CPU，而 sleep 函数则可以释放 CPU（存在精度误差）。
time.performance.delay(1);

var tk = tick() - tk;

console.log( tk / 1000 , "秒" );

/*
临时修改 sleep 函数精度。
Win10 2004 前会影响系统全局设置，Win11 之后窗口不可见时不保证提升精度。
*/
time.period(1,function(){

    var tk = tick()

    sleep(1)

    var tk2 = tick()

    console.log("sleep 精度" , (tk2-tk)/ 1000 , "秒" );
})

console.pause();
```