

---

Subject: Problem with GetTickCount on Linux

Posted by [steffen](#) on Wed, 30 May 2012 09:38:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

On my embedded system I'm using a GPS to correct the system time and I found that adjusting the realtime clock on a posix system, cripples all the timer functions in Upp.

The Posix standard has added some clock\_xxx functions to handle timers.

This gives a much better solution than the gettimeofday call used today.

Here is an updated function with fallback to the current gettimeofday call.

In Core/Util.cpp:

```
#ifdef PLATFORM_POSIX
dword GetTickCount() {
    #if _POSIX_C_SOURCE >= 199309L
        struct timespec tp;
        if (clock_gettime(CLOCK_MONOTONIC, &tp) == 0)
        {
            return (dword)((tp.tv_sec * 1000) + (tp.tv_nsec / 1000000));
        }
        return 0; // ?? (errno is set)
    #else
        struct timeval tv[1];
        struct timezone tz[1];
        memset(tz, 0, sizeof(tz));
        gettimeofday(tv, tz);
        return (dword)tv->tv_sec * 1000 + tv->tv_usec / 1000;
    #endif
}
#endif
```

In Core/TimeDate.cpp there is a GetSysTime, but I was missing a SetSysTime also.

Here is one that can be called from a program with administrator privileges:

```
bool SetSysTime(Time &ATime)
{
    #ifdef PLATFORM_POSIX
        struct tm    tmp_time;
        tmp_time.tm_sec = ATime.second;
        tmp_time.tm_min = ATime.minute;
        tmp_time.tm_hour = ATime.hour;
        tmp_time.tm_mday = ATime.day;
        tmp_time.tm_mon = ATime.month-1;
        tmp_time.tm_year = ATime.year-1900;
```

```
time_t raw_time = mktime(&tmp_time);

struct timespec sys_time;
sys_time.tv_sec = raw_time;
sys_time.tv_nsec = 0;

int result = clock_settime(CLOCK_REALTIME, &sys_time);
return (result == 0);
#endif
#ifdef PLATFORM_WIN32
SYSTEMTIME systime;
systime.wYear = ATime.year;
systime.wMonth = ATime.month;
systime.wDay = ATime.day;
systime.wHour = ATime.hour;
systime.wMinute = ATime.minute;
systime.wSecond = ATime.second;
systime.wDayOfWeek = 0;
systime.wMilliseconds = 0;
return SetSystemTime( &systime );
#endif
}
```

---