

```

inline uint64_t tickbegin() {
    int dummy[4];
    __cpuid(dummy, 0);

    // Prevent the compiler from optimizing away the whole Serialize function:
    volatile int DontSkip = dummy[0];
    return __rdtsc();
}

```

```

inline uint64_t tickend() {
    int dummy[4];
    uint64_t r = __rdtsc();
    __cpuid(dummy, 0);

    // Prevent the compiler from optimizing away the whole Serialize function:
    volatile int DontSkip = dummy[0];
    return r;
}

```

```

//*****

```

```

inline uint64_t tickstartscp() {
    int dummy[4];
    unsigned int a;
    __cpuid(dummy, 0);
    volatile int DontSkip = dummy[0];
    uint64_t r = __rdtscp(&a);
    return r;
}

```

```

inline uint64_t tickstopscp() {
    int dummy[4];
    unsigned int a;
    uint64_t r = __rdtscp(&a);
}

```

```

    __cpuid(dummy, 0);

    volatile int DontSkip = dummy[0];

    return r;
}

//*****

// Измерение тактов

// Вариант 1:

uint64_t tb, te, ticks;

    tb = tickbegin();

    for ( i=0; i<N; ++i ){

        //Код, время выполнения которого измеряется

    }

    te = tickbend();

    ticks = (te - tb)/N;

//Вариант 2:

uint64_t tb, te, ticks, mt, dt;

mt =1000000000LL;

for ( i=0; i<N; ++i ){

    tb = tickbegin();

    //Код, время выполнения которого измеряется

    te = tickbend();

    dt = te - tb;

    if ( dt < mt ) mt = dt;

}

ticks = mt;

```