Informatique Embarquée
M2 / 2014

Quelle heure est-il ?

Différents Temps

- Temps « horloge murale »
  - Avance sans arrêt, quelque soit l'activité des processus, processeurs
- Temps CPU
  - N'augmente que lorsque le processus (la thread) est actif (active)
    - CPU en mode utilisateur
      - Exécute votre code, ou le code des bibliothèques que vous utilisez
    - CPU en mode système
      - Quand votre processus, fait un appel système (fork, read, ...)

Sources de Temps

- PIT (8253 ou 8254)
  - Timer périodique génère une IT quand son compteur atteint une valeur donnée (programmable)
    - Valeur détermine « la résolution »
  - One-shot ou périodique
  - 3 timers en fait
    - 0 utilisé comme horloge système par Windows, Linux
- RTC (Real Time Clock)
  - Gère l'heure « réelle », même quand la machine est éteinte (batteries)

Sources de Temps

- HPET: High Precision Event Timer
  - Supposé remplacer le PIT
  - Jusqu'à 32 comparateurs
  - Jusqu'à 8 HPET chips sur une carte de PC...
  - Plus grande fréquence que le PIT
  - Compteurs sur 64 bits
  - Voir http://en.wikipedia.org/wiki/High_Precision_Event_Timer

Sources de Temps

- TSC : Time Stamp Counter
  - Compte le nombre de « ticks » depuis le boot
  - Attention en multipro/multicoeur
    - Pas synchronisés entre les coeurs
    - Sur certains processeurs: exécution possible « Out of Order »
    - Attention aux variations de vitesse de l'horloge!
    - (Speed stepping)
  - Voir http://en.wikipedia.org/wiki/Time_Stamp_Counter

Linux Ticks

- Le noyau utilise une fréquence d'horloge configurable (mais bon..)
- HZ
  - Historiquement à 100HZ (sur x86)
    - Cent tick d'horloge par seconde
    - 1 tick = 10 millisecondes
  - Maintenant, en général à 1000HZ
    - 1 tick = 1 milliseconde

gettimeofday

```
#include <sys/time.h>
int gettimeofday(struct timeval *tv, struct timezone *tz);
struct timeval {
    time_t      tv_sec;     /* seconds */
    suseconds_t tv_usec;    /* microseconds */
};
```

=> usec ne contient que les microsecondes additionnelles

=> la résolution n'est pas la micro seconde! On ne peut pas mesurer un intervalle d'1 micro seconde

=> 540 micro secondes (Debian / VirtualBox)

Résolution gettimeofday

```
gettimeofday(&t, NULL);

printf("since 01/01/1970  !d sec (0"!")  "
       " !d #$sec (0"!")%n",
       t&t'(sec, t&t'(sec,
       t&t'(#sec, t&t'(#sec);

gettimeofday(&t), NULL);

*+i,e ((t)&t'(sec $ t&t'(sec -- 0) &&
       (t)&t'(#sec $ t&t'(#sec -- 0)) .
  t&t'(sec - t)&t'(sec;
  t&t'(#sec - t)&t'(#sec;
  gettimeofday(&t), NULL);
/
```

time

```
#include <time.h>
time_t time(time_t *t);
```

=> secondes depuis le 01/01/1970

- **Exemple**:
- Sun Oct 17 18:07:41 CEST 2010
- secondes depuis le O1/01/1970: 1287331661 (0x4cbb1f4d)
- Intervalle observable : 1 seconde
  - Delta trouvé: 1 (1287331941, 1287331942)

Code « time »

```
0inc,#de 1time&+2
0inc,#de 1std,i3&+2
0inc,#de 1stdio&+2

main()
.
  time(t t -  time(NULL);

  printf("sec O1/O1/1970  !d (O"!")%n",t, t);
  time(t t) -  time(NULL);
  4+i,e (t) $ t -- O) .
    t - t);
    t) - time(NULL);
  /
  printf("5e,ta tro#'6  !d (!d, !d)%n", t) $t, t, t));
/
```

clock_gettime

```
0inc,#de 1time&+2
int c,oc7(getres(c,oc7id(t c,7(id,
                    str#ct timespec 8res);
int c,oc7(gettime(c,oc7id(t c,7(id,
                    str#ct timespec 8tp);
str#ct timespec .
   time(t    t'(sec;  /8 seconds 8/
   ,ong      t'(nsec; /8 nanoseconds 8/
/;
-2 c,oc7(id   CLOCK_REALTIME
-2 n6cessite ,a 3i3,iot+9:#e $,rt
```

clock_gettime

```
c,oc7(gettime(;LO;<(=>?L@AB>, &t);

c,oc7(gettime(;LO;<(=>?L@AB>, &t));
4+i,e ((t)&t'(sec $ t&t'(sec -- 0) &&
       (t)&t'(nsec $ t&t'(nsec -- 0)) .
    t&t'(sec - t)&t'(sec;
    t&t'(nsec - t)&t'(nsec;
    c,oc7(gettime(;LO;<(=>?L@AB>, &t));
/
printf("5e,ta  !d sec  !,d n$sec(!d, !d (,!d !,d))%n",
       t)&t'(sec $ t&t'(sec, t)&t'(nsec $ t&t'(nsec,
       t)&t'(nsec, t)&t'(nsec, t&t'(nsec, t&t'(nsec);

c,oc7(getres(;LO;<(=>?L@AB>, &t);
printf("reso,  !d sec (0"!")   !,d n$sec (0"!,")%n",
     t&t'(sec, t&t'(sec, t&t'(nsec, t&t'(nsec);
```

rdtsc

- rdtsc
  - Read Time Stamp Counter
  - Voir les réserves déjà exprimées

```
static in,ine #intCD(t rdtsc ('oid)
.
    #intCD(t tsc;
    /8 cp#id needed to drain pipe,ine 8/
    asm 'o,ati,e ("cp#id; rdtsc"   "-?" (tsc));
    ret#rn tsc;
/
```

Ce matin

- Je suis arrivé à la vitesse de 42 !

Ce matin

- Je suis arrivé à la vitesse de 42 !
  - UNITE ?
  - Quel moyen ?
- Pour le TP
  - Unités... sec, millisec, µsec
  - Environnement matériel :
    - processeur, vitesse, caches, mémoire, vitesse,
  - Environnement logiciel :
    - Système, compilateur...