UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Programação de um nó Contiki

Disciplina:
Internet das Coisas
(TECS1)

Prof. Hermano Pereira

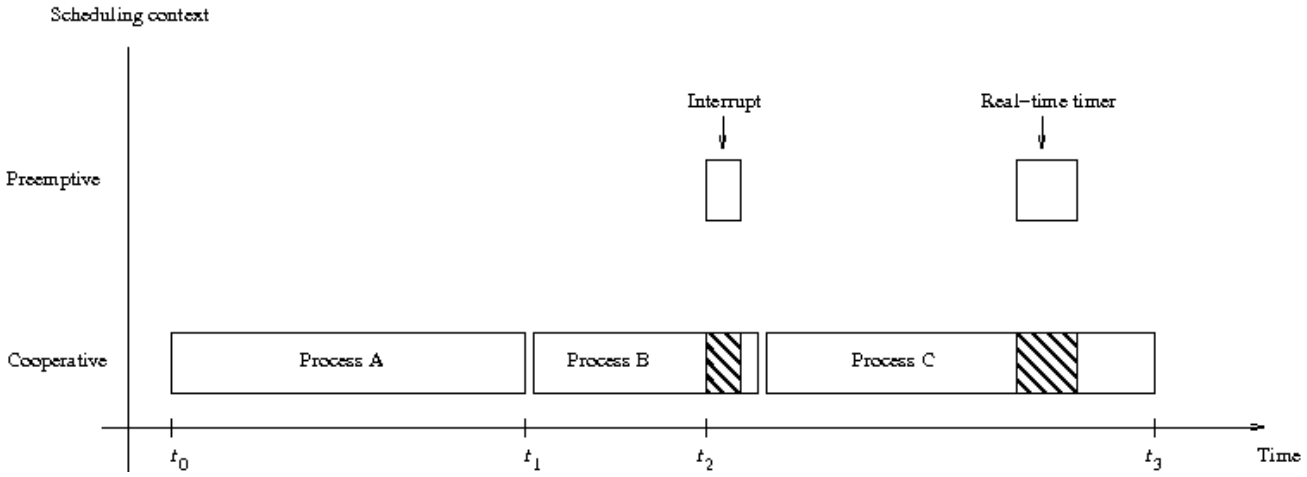
TSI – UTFPR - GP

Introdução

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

- → Processos/Threads
- → Estruturas de Repetição e de Condição
- → Variáveis
- → Timers

Processos/Threads



Processos/Threads

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

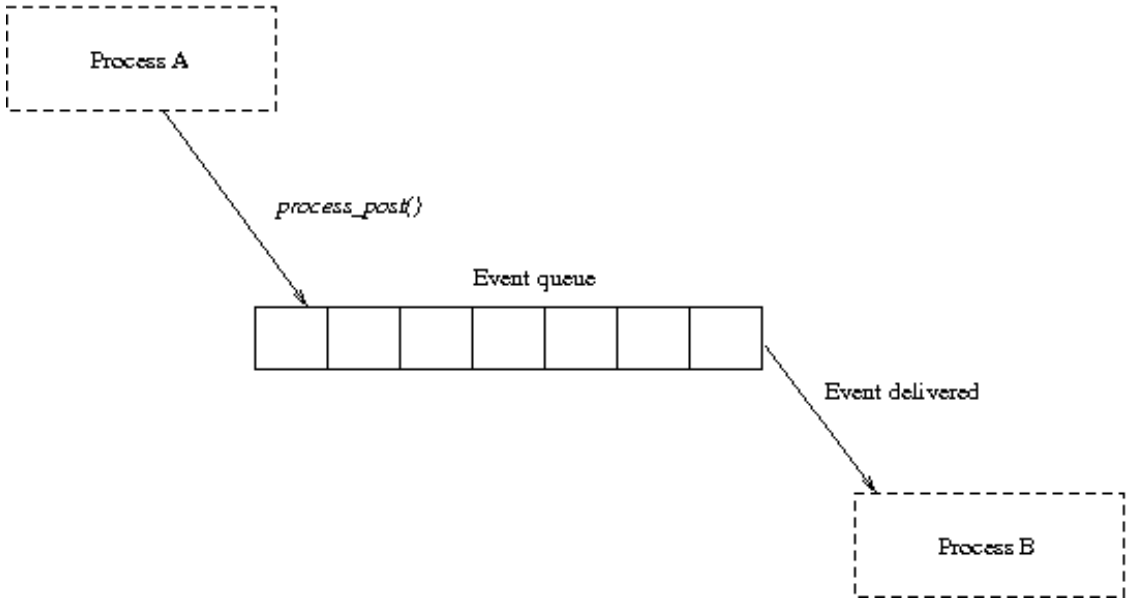
```
PROCESS_THREAD(hello_world_process, ev, data)
{
  PROCESS_BEGIN();

  printf("Hello, world\n");

  PROCESS_END();
}
```

Processos/Threads

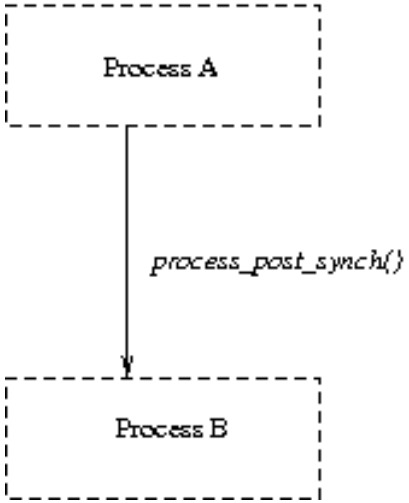
```
PROCESS(t1_process, "Mote T1");
PROCESS(t2_process, "Mote T2");
AUTOSTART_PROCESSES(&t1_process,&t2_process);
PROCESS_THREAD(t1_process, ev, data)
{
  PROCESS_BEGIN();
  printf("Thread 1 de Fulano!(barra-n)");
  PROCESS_END();
}
PROCESS_THREAD(t2_process, ev, data)
{
  PROCESS_BEGIN();
  printf("Thread 2 de Fulano!(barra-n)");
  PROCESS_END();
}
```

Eventos

Assíncrono



Síncrono



ProtoThreads

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

```
PROCESS_BEGIN(); // Declares the beginning of a process'
protothread.

PROCESS_END(); // Declares the end of a process'
protothread.

PROCESS_EXIT(); // Exit the process.

PROCESS_WAIT_EVENT(); // Wait for any event.

PROCESS_WAIT_EVENT_UNTIL(); // Wait for an event, but with
a condition.

PROCESS_YIELD(); // Wait for any event, equivalent to
PROCESS_WAIT_EVENT().

PROCESS_WAIT_UNTIL(); // Wait for a given condition; may
not yield the process.

PROCESS_PAUSE(); // Temporarily yield the process.
```

Estruturas de Repetição/Condição

```
PROCESS_THREAD(t1_process, ev, data)
...

while(1)
{
   PROCESS_WAIT_EVENT();
   if(ev == sensors_event && data == &button_sensor)
   {
     printf("Botão foi acionado!\n”);
   }
}
```

… básico de programação.

Variáveis

Variáveis locais:

```
void x() {
   int variavel;
   variavel = 10; // escopo ok
}

void y() {
   variavel = 20; // erro
}
```

Variáveis globais:

```
int variavel;
void x() {
   variavel = 10; // escopo ok
}

void y() {
   variavel = 20; // escopo ok
}
```

Variáveis

Variáveis automáticas:

```
while (1) {
    int variavel = 0;
    variavel = variavel + 1;
    printf("%d\n", variavel); // imprime 1 1 1 1 1 ...
}
```

Variáveis estáticas:

```
while (1) {
    static int variavel = 0;
    variavel = variavel + 1;
    printf("%d\n", variavel); // imprime 1 2 3 4 5 ...
}
```

Timers

```
...
  static struct etimer et;
  etimer_set(&et, CLOCK_SECOND * 60);
  PROCESS_WAIT_EVENT();
  if (etimer_expired(&et))
  {
        printf("Tempo expirou\n");
        etimer_reset(&et);
  }
...
```

Vamos trabalhar?

Referências

Processes: https://github.com/contiki-os/contiki/wiki/Processes

Timers: https://github.com/contiki-os/contiki/wiki/Timers

Explanação variáveis em C:
http://gd.tuwien.ac.at/languages/c/programming-bbrown/c_046.htm