Instituto Tecnológico de Aeronáutica - ITA

Laboratório 2

Sprint 01

LUCAS NICOLLI TOSI

Sembro de 2019

Instituto Tecnológico de Aeronáutica – ITA
Divisão de Ciência da Computação - IEC
Pós-Graduação em Eng. Eletrônica e Computação

CE-235 Sistemas Embarcados de Tempo Real
Prof. Denis Loubach

Roteiro de Laboratório 02

1 Objetivo

1.1 Objetivos Gerais

Modelagem e implementação de máquinas de estados para as tarefas do RTOS utilizado na disciplina.

2 Desenvolvimento

2.1 Desenvolvimento do laboratório

Os seguintes itens deverão ser exercitados neste laboratório:

1. Utilização de modelos SysML;
   - Modelar duas máquinas de estados em SysML para controle de cores em LED RGB;
2. Criar duas tarefas no RTOS, utilizando o *Processo Expert*; e
3. Implementar as máquinas de estados modeladas.
   - Desenvolver o código da tarefa, ou seja, a parte de inicialização e parte do *loop infinito*
   - Utilizar o LED RGB da placa *target* para visualizar as trocas de estados, utilizando código fonte fornecido pelo professor
   - Utilizar os `delays` para controle de tempo das tarefas

2.2 Sobre o código fonte fornecido

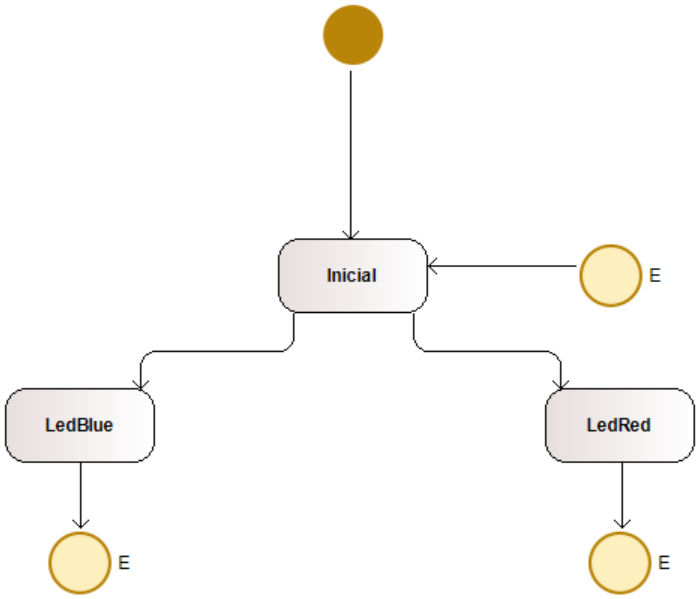
Os códigos para gerenciamento do LED RGB do kit fornecidos neste lab utilizam chamadas da KSDK *library*.

Caso houver algum problema na fase do *cross-compiler* ou *linker* relacionados à utilização do código fornecido, certifique-se que o projeto em questão esteja utilizando a *library* (arquivo `.a`) do KSDK.

2.3 Sobre a modelagem

Poderão ser utilizadas ferramentas gratuitas para modelagem das máquinas de estado tais como *Modelio SysML Architect*, *Papyrus SysML* ou qualquer outra de preferência do aluno.

1. ***State Machine Diagram*:**



2. **Código da Task 1, utilizando a estrutura switch:**

```
void Task1_task(os_task_param_t task_init_data)
{
  /* Write your local variable definition here */

    uint8_t estadoLedAzul=0;

#ifdef PEX_USE_RTOS
  while (1) {
#endif
      /* Write your code here ... */

      /* OSA_TimeDelay(1000);                  Example code (for task release) */


      switch (estadoLedAzul){
      case 0:
          ledrgb_setBlueLed();
          estadoLedAzul=1;
          OSA_TimeDelay(1000);
          break;
      case 1:
          ledrgb_clearBlueLed();
          estadoLedAzul=0;
          OSA_TimeDelay(1000);
          break;

      }


#ifdef PEX_USE_RTOS
  }
#endif
}
```

3. **Código da Task 2, utilizando a estrutura switch:**

```
void Task2_task(os_task_param_t task_init_data)
{
  /* Write your local variable definition here */

    uint8_t estadoLedVermelho=0;

#ifdef PEX_USE_RTOS
  while (1) {
#endif
      /* Write your code here ... */

      /* OSA_TimeDelay(1000);                 Example code (for task release) */


      switch (estadoLedVermelho){
      case 0:
          ledrgb_clearRedLed();
          estadoLedVermelho=1;
          OSA_TimeDelay(1000);
          break;

      case 1:
          ledrgb_setRedLed();
          estadoLedVermelho=0;
          OSA_TimeDelay(1000);
          break;

      }



#ifdef PEX_USE_RTOS
  }
#endif
}
```

4. **Main_Task com a rotina de inicialização:**

```
void main_task(os_task_param_t task_init_data)
{
    /* Write your local variable definition here */

    /* Initialization of Processor Expert components (when some RTOS is active). DON'T REMOVE THIS CODE!!! */
#ifdef MainTask_PEX_RTOS_COMPONENTS_INIT
    PEX_components_init();
#endif
    /* End of Processor Expert components initialization.  */

    ledrgb_init( );

#ifdef PEX_USE_RTOS



    while (1) {
#endif
       /* Write your code here ... */

       OSA_TimeDelay(10);                /* Example code (for task release) */


#ifdef PEX_USE_RTOS
    }
#endif
}
```

5. **Upload do LAB2 no GitHub: https://github.com/lntosi/CE-235**