**Sistemas Embarcados – UTFPR – 2019/2**

**Programação concorrente com RTOS – Controle de um motor elétrico**

1. **Requisitos do Sistema**
   1. **Requisitos Funcionais**

* **RF1:** O Sistema deve realizar o controle da velocidade e do sentido de um motor elétrico;
* **RF2:** O sistema deve permitir a configuração da velocidade e sentido desejados pelo usuário;
* **RF3:** O sistema deve possuir dois modos de operação: O modo de configuração e o modo de execução;
  + **RF3.1:** O modo de configuração deve permitir ao usuário selecionar as configurações de funcionamento do sistema, no caso, velocidade e sentido de rotação do motor.
  + **RF3.2:** O modo de execução deve executar o sistema com base nas configurações iniciais.
* **RF4:** O sistema deverá possuir uma interface visual (display, tela, visor, etc.) para o usuário.
  + **RF4.1:** A interface visual deve mostrar instruções de como utilizar o sistema;
  + **RF4.2:** A interface visual deve mostrar as configurações de velocidade e sentido iniciais do motor, quando o sistema estiver em modo de configuração;
  + **RF4.3:** A interface visual deve exibir a velocidade e sentidos reais do motor, quando o sistema estiver em modo de execução;
* **RF5:** O sistema deve fazer uso de botões (push-buttons, teclado, etc.) para permitir a transição entre os modos de operação.
  1. **Requisitos Não-Funcionais**
* **RNF1:** O sistema deve possuir um motor, que será o elemento atuador do sistema que será controlado; um *encoder* de quadratura, que será o elemento sensor do sistema e de um hardware (SDK), que implementará a técnica de controle;
* **RNF2:** O Sistema deve fazer uso de um driver L298N para interface do hardware com o motor;
* **RNF3:** O hardware do sistema deverá ser baseado no microcontrolador TM4C1294;
* **RNF4:** O ambiente de desenvolvimento do sistema deverá ser o IAR EWARM V8.40;
* **RNF5:** O sistema deve utilizar UART para exibir e obter informações;
* **RNF6:** O sistema deve fazer uso de um computador com interface UART para troca de informações com o usuário.
  + **RNF6.1:** O sistema deve utilizar o monitor do computador para exibir informações;
  + **RNF6.2:** O sistema deve utilizar o teclado do computador para obter informações;
* **RNF7:** O hardware ou SDK deverá fazer uso de pinos, periféricos e interrupções para gerar o firmware;
* **RNF8:** O hardware ou SDK deve utilizar CMSIS-RTOS;
  + **RNF8.1:** O sistema deve possuir uma tarefa destinada à configuração inicial do sistema;
  + **RNF8.2:** O sistema deve possuir uma tarefa destinada à leitura do sensor;
  + **RNF8.3:** O sistema deve possuir uma tarefa destinada a implementar o controle e gerar o sinal que controlará o motor;
  + **RNF8.4:** O sistema deve fazer uso de Mutexes, Semáforos, Filas de mensagens e Flags para organizar as tarefas e controlar acesso a recursos externos.
* **RNF9:** O sistema deve utilizar a técnica de controle PID;
  + **RNF9.1:**  O sistema deve utilizar os sinais do sensor para gerar a velocidade e o sentido atuais;
  + **RNF9.2:** O sistema deve gerar um sinal PWM, que será usado para controlar o motor;
  + **RNF9.3:** O sistema deve possuir variáveis e definições que permitam a calibragens e ajustes referentes à técnica de controle;

1. **Arquitetura Física do Sistema**

O sistema físico apresenta a seguinte estrutura:

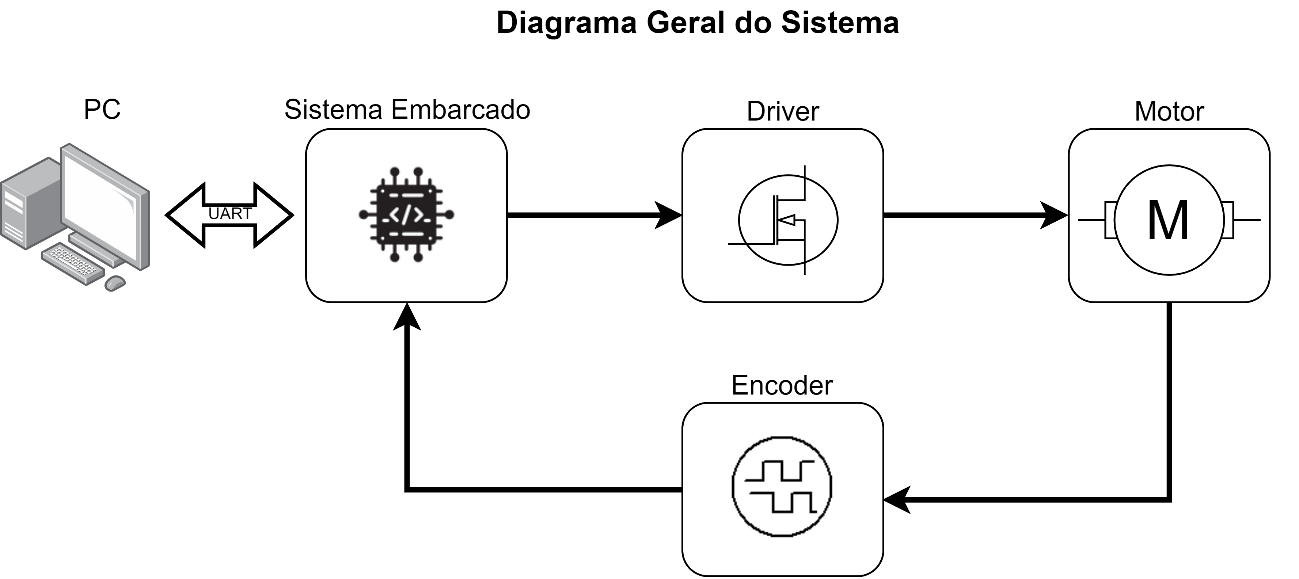
****

Figura 1: Esboço da arquitetura física do sistema.

O sistema embarcado é responsável por interpretar os sinais do sensor, gerar um sinal PWM utilizando-se uma regra de controle e se comunicar via UART com uma interface serial do PC. O Driver é responsável por transferir o sinal PWM do sistema embarcado para o motor, além de selecionar o seu sentido de rotação. O motor será o atuador do sistema, o qual terá a velocidade e o sentido de rotação controlados através dos sinais vindos do driver. O encoder, que funcionará como o sensor do sistema, é responsável por criar sinais para o microcontrolador que representam a velocidade e o sentido de rotação do motor.