

Laboratório 4 - Comunicação Serial (UART)

Aluno: Lucas Silva Beneti

Para o Laboratório 4 foi requisitado o gerenciamento de dados transmitidos pela interface UART para a Tiva, onde deve ocorrer um processamento e retornar o resultado por essa mesma porta. O objetivo é converter letras maiúsculas recebidas pela porta UART nas mesmas letras minúsculas para, sem alterar os outros bytes da mensagem.

Planejamento do programa para a comunicação serial

1. Definição do problema

O processador da Tiva receberá, através da porta de comunicação UART alguns bytes com letras maiúsculas, e deverá converter esses caracteres em letras minúsculas e mandar esses bytes de volta sem alterar outros bytes da mensagem.

2. Levantamento de requisitos

O projeto tem os seguintes requisitos:

Funcionais:

RF1 - A comunicação deve ocorrer através da porta UART da placa, tanto para receber dados quanto para mandar.

RF2 - O processador deve converter as letras maiúsculas recebidas em suas correspondentes minúsculas

RF3 - O processador deve retornar a mensagem com letras minúsculas através da comunicação UART para o terminal simulador

RF4 - As interrupções devem ser registradas manualmente no vetor de interrupção da Tiva, para essa finalidade, a biblioteca TivaWare não pode ser utilizada.

Não funcionais:

RFN1 - A comunicação deve ser feita por meio de um emulador de comunicação serial UART, como por exemplo com o uso de softwares como Tera Term ou Putty.

RFN2 - Devem ser realizados testes para comprovar o funcionamento do programa e da comunicação UART.

3. Plataforma de Hardware

A plataforma determinada:

- Placa Tiva EK-TM4C1294XL
- Utiliza o processador TM4C1294NCPDT

Não há como requisito a utilização de nenhum periférico nesse experimento.

4. Plataforma de Software

Para a interface com a placa, será utilizada a biblioteca TivaWare, junto com algumas configurações iniciais necessárias no vetor de interrupções. Para a

configuração da placa no IAR, deverá ser usado o arquivo `startup_ewarm.c` que contém a configuração do vetor de interrupções.

5. Estudo da plataforma de Software

A biblioteca TivaWare possui implementações de funções úteis para esse experimento que lida com a transmissão de dados seriais. Há diversas configurações necessárias para serem feitas, escolhendo uma das portas possíveis de UART. Será necessário habilitar as portas GPIO necessárias como receptora de dados, habilitar para interrupções entre outras configurações.

Em relação a conversão de caracteres maiúsculos para minúsculos, acredito que um simples AND do valor correspondente ao caractere em binário com o número 32 (10000) deva realizar a conversão. Claro que após conseguirmos o novo número, devemos “transformar” o número em seu respectivo equivalente *char* da tabelas ASCII antes de mandá-lo de volta pela serial.

Para o terminal que simulará a entradas de dados, será utilizado o software Putty, muito usado para acesso remoto a computadores por meio de conexões SSH, neste caso, será utilizado como entrada serial configurado na porta COM correta que identifica a conexão USB da placa TivaC que utilizo com meu PC

6. Design da Solução

Supondo que os caracteres a serem convertidos venham do terminal emulador de USBUART:

- Configurar vetor de interrupções para as funções handlers designadas para agir quando houver mensagem na UART
- Inicialização da interface UART da placa
- Configurações necessárias para para a interrupção pela UART ser habilitada
- Ao receber algo na porta UART, será necessário avaliar se o caractere precisará de conversão (se for um maiúsculo)
- Se for um maiúsculo, pegamos o equivalente em decimal na tabelas ASCII e somamos 32, esse novo número, transformado para char, corresponderá à mesma letra porém minúscula
- Se não for maiúscula, apenas devolve pela porta UART para o remetente
- O programa ficará em loop infinito, apenas esperando as interrupções recebidas pela UART

Conclusões

Os requisitos tanto funcionais quanto não funcionais foram implementados. O programa ao iniciar, fica esperando a entrada de algum caractere pelo terminal serial remoto (representado pelo software Putty configurado para comunicar-se com a placa TivaC na porta COM3). Ao receber um char minúsculo, o mesmo caractere é retornado para o remetente, se for maiúsculo, ocorre a conversão e o caractere também é enviado pelo

remetente, sempre após um “\n” pois é importante que a porta saiba quando a mensagem acabou.

A *interrupt handler* da UART foi registrada manualmente no arquivo *startup_ewarm.c* presente no projeto, que é o arquivo que contém a configuração do vetor de interrupções para o processador utilizado.