Escola Politécnica da USP

Departamento de Engenharia de Computação e Sistemas Digitais

PCS2021 Laboratório Digital II

Turma 1 – Prof. Edson Midorikawa

Experiência 3

UART

Felipe Hamamoto Toyoda

Mi Che Li Lee

grupo 5 – bancada A-2

Data: 16/09/2013

## objetivos

O objetivo desta experiência é projetar e implementar circuitos digitais para comunicação de dados serial (transmissão de dados) com um terminal de dados, utilizando a norma EIA-RS-232C e o código ASCII (American Standard Code Information Interchange).

A parte prática consiste no projeto e na implementação de um circuito digital que troca (envia e recebe) dados (caracteres em código ASCII) de um terminal serial usando uma placa de desenvolvimento FPGA.

## projeto

O circuito digital que será implementado realizará a transmissão e recepção serial assíncrona de dados para um terminal. A figura abaixo apresenta a interface de comunicação da placa FPGA e o terminal serial.



Figura 1 - Diagrama de Blocos do Circuito do UART

Um sinal de reset (botão 2) está disponível para reiniciar o circuito e zerar os registradores internos que armazenam dados de transmissão e de recepção.

### Transmissão serial

Os dados são caracteres ASCII e serão especificados em 7 chaves. A transmissão se inicia, após a definição dos dados nas chaves, ao pressionar-se um botão de partida (botão 1).

O terminal serial será operado por pulsos de tensão entre +12V e -12V, portanto será usado um conversor de nível (do circuito integrado MAX232 da placa Altera DE2).

A figura abaixo mostra o diagrama de blocos a ser seguido para o módulo de transmissão do circuito.



Figura 2 - Diagrama de Blocos do Circuito de Transmissão de Dados

### Recepção serial

O segundo módulo é o circuito de recepção de dados enviados a partir do terminal serial. Este deve detectar o acionamento de uma tecla no teclado do terminal e mostrar o código ASCII nos 4 displays hexadecimais (D0 a D3). Além disso, deve testar o bit de paridade e indicar, por meio de um LED (L0), que ocorreu um erro de transmissão. A comunicação deve ser projetada e configurada com uma velocidade de transmissão de 110 bauds, 7 bits de dados, paridade ímpar e 2 stop bits.

Os dois últimos caracteres digitados devem ser armazenados para serem mostrados, sendo que o último caractere deve ser mostrado no primeiro e segundo display, enquanto o penúltimo caractere deve ser mostrado no terceiro e no quarto display (da esquerda para direita).

A projeção do circuito de recepção seguirá o diagrama de blocos abaixo:



Figura 3 - Diagrama de Blocos do Circuito de Recepção de Dados

### implementação

A montagem seguida a seguinte configuração:

|  |  |  |
| --- | --- | --- |
| **Nome do sinal** | **Pino** | **Descrição** |
| UART\_RXD | PIN\_C25 | UART Receiver |
| UART\_TXD | PIN\_B25 | UART Transmitter |

Dados TX: chaves SW0 a SW6

Partida: botão KEY2

Reset: botão KEY1

Displays: displays HEX0 a HEX3

O circuito integrado MAX232 contém um único encapsulamento para a conversão de tensão para ambos os sentidos (TTL para RS232C e RS232C para TTL), com a seguinte pinagem:



Figura 4 - Pinagem e esquema lógico do MAX232



Figura 5 - Suporte a comunicação serial da placa DE2 da Altera (fonte: Altera 2008)

## Atividades experimentais

Primeiramente sintetizamos o circuito do UART para a placa Altera DE2.

Para testar o circuito, realizamos alguns testes com diferentes caracteres. Espera-se que os resultados estejam de acordo com os resultados previstos com as cartas de tempo abaixo:

Após a exibição correta dos valores, variamos a taxa de comunicação do terminal serial para ver a reação do circuito:

### PERGUNTAS

1. Há alguma limitação de funcionamento do circuito projetado? Elabore uma discussão sobre as frequências mínima e máxima de funcionamento do circuito.

2. O módulo de transmissão serial sofre alguma influência do módulo de recepção? Se sim, qual é esta influência. Explique.

3. Os dados de teste (caracteres a serem mostrados no circuito) influem nos testes efetuados? Caso afirmativo explique por que.

4. É possível a transmissão e a recepção de dados operarem em frequências diferentes? Como isto pode ser realizado?

## conclusões