

Comunicação Serial Assíncrona

Versão 2016

RESUMO

O objetivo desta experiência é projetar circuitos digitais para comunicação serial de dados (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 envia dados digitais (caracteres em código ASCII) para um terminal serial usando a linguagem de descrição de hardware VHDL e uma placa de desenvolvimento FPGA DE2 da Altera.

OBJETIVOS

Após a conclusão desta experiência, os seguintes tópicos devem ser conhecidos pelos alunos:

- Comunicação serial assíncrona (RS232C);
- Conversores de nível de tensão;
- Projeto com FPGA;
- VHDL
- Depuração com ferramentas.

1. PARTE EXPERIMENTAL

A parte experimental envolve o desenvolvimento de um circuito digital para a transmissão de dados para um terminal serial, usando a placa de desenvolvimento FPGA DE2 da Altera.

1.1. Atividades Pré-Laboratório

- a) **Pesquisa do manual da placa Altera DE2.** Estude a documentação da placa de desenvolvimento FPGA Altera DE2 para localizar os componentes de suporte a comunicação serial (circuito *transceiver* MAX232 e conector DSUB para comunicação RS-232C). A figura 1 ilustra o diagrama esquemático dos componentes relativos à interface serial da placa DE2 da Altera.

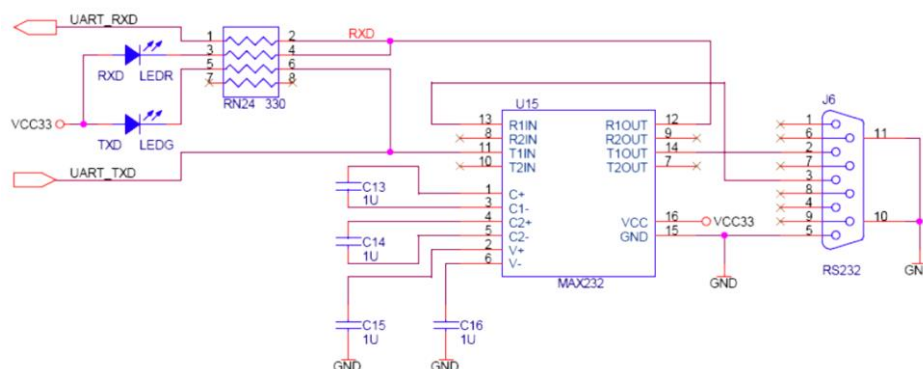


Figura 1 – Suporte a comunicação serial da placa DE2 da Altera (fonte: Altera 2008).

- b) Pesquisar a identificação dos pinos relacionados à comunicação serial na placa Altera DE2.

Nome do sinal	Pino	Descrição
UART_RXD		UART Receiver Data
UART_TXD		UART Transmitter Data

DICA: o sinal DADO_SERIAL do projeto deve ser designado para o pino do sinal UART_TXD.

- c) **Especificação do Projeto:** O projeto da parte experimental consiste em projetar, implementar e documentar um circuito digital em VHDL que faça uma transmissão serial assíncrona para um terminal serial, que estará disponível no Laboratório Digital. O dado a ser transmitido é um caractere em código ASCII, colocado em 7 chaves (**dados_ascii**) e deve ser apresentado no terminal, ao pressionar-se um botão de **partida**. Ao final da transmissão, o sinal **pronto** deve ser acionado.

O circuito deve ser projetado de forma a permitir a apresentação do caractere colocado nas chaves a cada acionamento do botão de partida. A carga dos dados, a geração do bit de paridade, a transmissão da sequência correta de bits e o término da transmissão devem ser realizadas automaticamente. A escolha da **paridade** – par ou ímpar – deve ser explicitada no planejamento.

A figura 1 mostra a interface do circuito, com os principais sinais de entrada e saída.

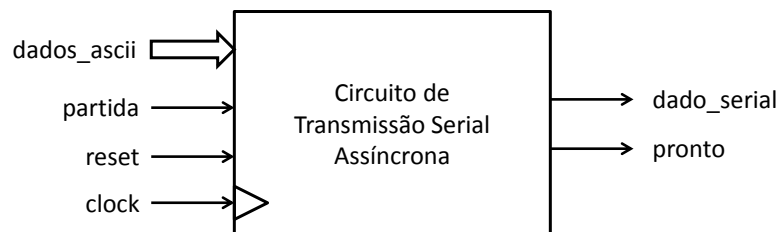


Figura 1 – Interface do Circuito de Transmissão Serial Assíncrona.

A Unidade de controle pode ser especificada através do diagrama de transição de estados da figura 2. O circuito deve permanecer no estado **Inicial** enquanto o sinal de **partida** não for acionado. Com o acionamento do sinal de partida, muda-se o circuito para o estado de **Preparação** que deve carregar o **dado_ascii** a ser transmitido e iniciar um contador interno. Em seguida, o circuito muda para o estado de **Transmissão** que envia todos os bits de comunicação serial. O término da transmissão é sinalizado pelo sinal **Fim**, e o circuito passa para o estado **Final**, que aciona o sinal **Pronto**.

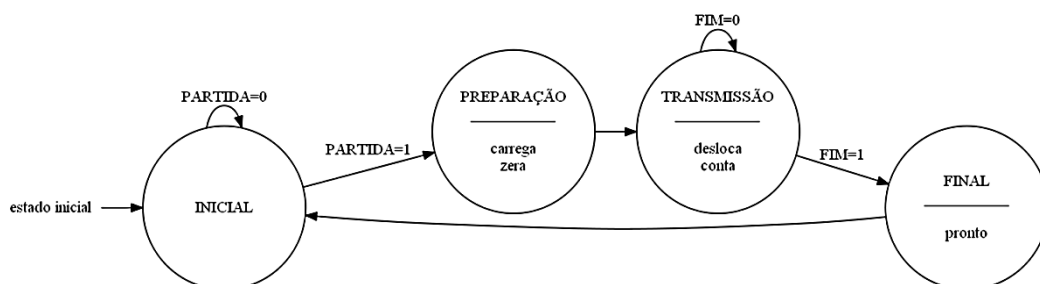


Figura 2 – Diagrama de transição de estados da Unidade de Controle do Circuito de Transmissão Serial Assíncrona.

O Fluxo de dados do circuito de transmissão serial assíncrona é responsável pelo armazenamento dos dados do caractere ASCII de 7 bits a ser transmitido, cálculo do bit de paridade, a serialização dos bits para comunicação serial, e o controle da quantidade dos bits transmitidos. A interface do Fluxo de dados é apresentada na figura 3.

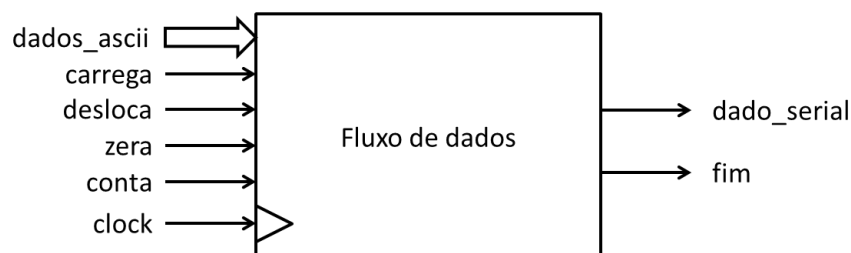


Figura 3 – Interface do Fluxo de dados do Circuito de Transmissão Serial Assíncrona.

Internamente o Fluxo de dados do circuito de transmissão serial assíncrona contém alguns componentes. Dois destes componentes são um registrador de deslocamento e um contador módulo-m.

O registrador de deslocamento tem uma interface conforme ilustrada na figura 4. O bits do dado paralelo são ligados na entrada de **dados**, e temos dois sinais de controle, **carrega** e **desloca**, que são responsáveis pela carga e deslocamento dos bits no deslocador, respectivamente.



Figura 4 – Interface do Registrador de deslocamento.

O contador módulo-m tem sua interface apresentada na figura 5. Este contador apresenta uma contagem de 0 até m-1, controlado pelos sinais de controle **zera** e **conta**. A saída adicional **fim** é ativada sempre que o contador atinge seu valor máximo.

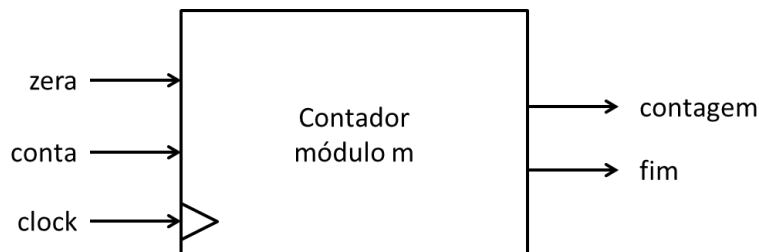


Figura 5 – Interface do Contador módulo-m.

- d) O Planejamento deve conter a descrição do projeto em VHDL de cada um dos componentes e módulos, uma explicação da integração destas partes e, finalmente, do funcionamento do projeto completo. Sugere-se que sejam realizadas simulações para descrever o funcionamento de cada entidade VHDL.
- e) Defina os casos de testes devem ser executados para assegurar o correto funcionamento do circuito completo. Defina alguns sinais de depuração. Acrescente formas de onda no planejamento.
- f) O terminal serial que será utilizado no laboratório deve ser operado por tensão, isto é, com pulsos de tensão entre +12V e -12V. É necessário usar um conversor de nível para a interface com a linha serial. A placa altera DE2 dispõe de um *transceiver* MAX232 (TTL para RS-232C) para realizar esta funcionalidade. A Figura 6 mostra o diagrama de blocos da montagem experimental a ser executada.

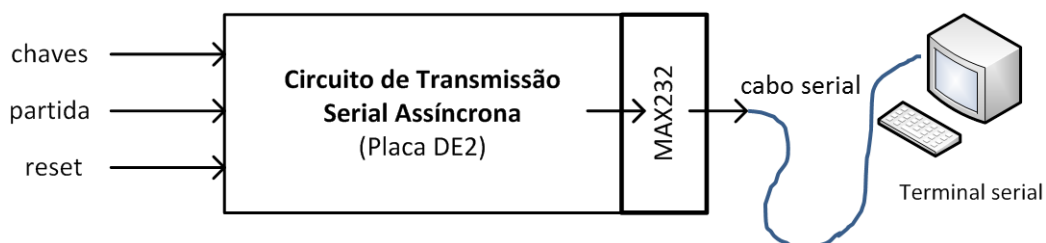


Figura 6 – Esquema geral da Montagem do Circuito de Transmissão Serial Assíncrona.

1.2. Implementação do Projeto na Placa Altera DE2

Neste item vamos implementar, no Laboratório Digital, o projeto do circuito de transmissão serial na placa Altera DE2.

- g) Sintetizar o circuito de transmissão serial para a placa Altera DE2.
Usar a seguinte designação de pinos:

- DADOS_ASCII: chaves SW0 a SW6
- PARTIDA: pino GPIO_1[0]
- RESET: pino GPIO_1[1]
- DADO_SERIAL: UART_TXD
- PRONTO: LEDG[0]

- h) Testar o circuito de transmissão serial para vários valores de taxa de transmissão e dados seriais.
- i) Use a ferramenta **SignalTap II** para a visualização dos sinais digitais para fins de depuração. Adicione figuras com as saídas da ferramenta no Relatório.
- j) Analise o funcionamento do circuito transmissor com a variação da frequência do relógio (*clock* de transmissão) e sua relação com o dado enviado.

1.3. Desafio

Neste item deverão ser estudados e implementados alguns melhoramentos no circuito de transmissão serial na placa DE2.

- k) Uma **modificação** no projeto do circuito de transmissão serial assíncrona será solicitada pelo professor. Estude esta modificação e verifique qual parte do projeto inicial deverá ser alterado.
- l) Projete o circuito e sintetize na placa DE2. Explique a designação de pinos.
- m) Documente os resultados obtidos nos testes realizados.

1.4. Atividades Pós-Laboratório

Após a conclusão das atividades programadas, responda as perguntas abaixo:

1. Há alguma limitação de funcionamento do circuito projetado? Elabore uma discussão qualitativa sobre as frequências mínima e máxima de funcionamento do circuito.
2. A placa DE2 contém sinais de clock internos de 27MHz e de 50MHz. Como o projeto do grupo poderia ser modificado para usar um sinal de clock interno da placa DE2.
3. O que acontece se a frequência do circuito de transmissão tiver um erro de x% em relação à taxa nominal do terminal serial? Por exemplo, se o terminal for configurado a 9600 bauds e a transmissão tiver um clock de 9120Hz (erro de 5%).

CONFIGURAÇÃO DA COMUNICAÇÃO SERIAL COM O PC

Um computador tipo PC será usado para emular um terminal serial. Conecte o cabo serial na saída da placa FPGA no adaptador USB-serial e utilize o software de comunicação serial *HyperTerminal* do Windows (ou outro compatível). Crie uma nova conexão usando os parâmetros abaixo (configurações de porta), como na figura 3.

- Conexão: COMx (determinar porta correta usando o Painel de Controle do Windows)
- Bits por segundo: 300
- Bits de dados: 7
- Paridade: ímpar
- Bits de parada: 2
- Controle de fluxo: Nenhum

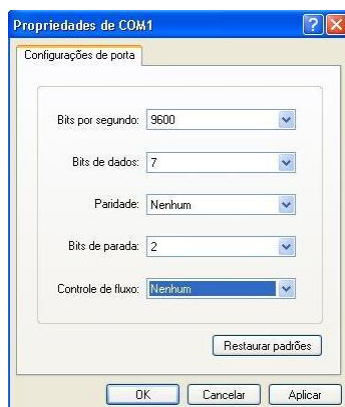


Figura 3 – Exemplo de configuração do HyperTerminal.

Para testar a conexão da porta serial antes de conectá-lo na placa FPGA, faça um curto-circuito com os pinos¹ TX e RX do cabo serial e veja se o que for digitado é ecoado no terminal.

¹ Os pinos TX e RX do cabo serial são, respectivamente, os pinos 2 e 3 do conector DB9 ou DE9.

2. BIBLIOGRAFIA

- ALTERA. **Altera DE2 Development and Education Board User Manual**. 2008.
- CCITT - Fifth Plenary Assembly. Green Book. Vol. VIII, Geneve, December 1972.
- Electronic Industries Association. **Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Date Interchange EIA-RS-232-C**, Washington, August 1969.
- FREGNI, E.; SARAIVA, A. M. **Engenharia do Projeto Lógico Digital: Conceitos e Prática**. Editora Edgard Blücher, 1995.
- HELD, G. **Understanding Data Communications**. 6th ed., New Riders, 1999.
- TOCCI, R. J.; WIDMER, N.S.; MOSS, G.L. **Sistemas Digitais: Princípios e Aplicações**. Prentice-Hall, 11^a ed., 2011.
- WAKERLY, John F. **Digital Design Principles & Practices**. 4th edition, Prentice Hall, 2006.

3. EQUIPAMENTOS NECESSÁRIOS

- 1 computador com interface serial e software de comunicação.
- 1 computador com software Altera Quartus II.
- 1 dispositivo Analog Discovery da Digilent.
- 1 placa de desenvolvimento FPGA DE2 da Altera com o dispositivo Altera Cyclone II EP2C35F672C6.

Histórico de Revisões

E.S.G. e F.N.A/2001 – revisão
 E.T.M./2004 – revisão
 E.T.M./2005 – revisão
 E.T.M./2008 – revisão
 E.T.M./2011 – revisão
 E.T.M./2012 – revisão
 E.T.M./2013 – revisão da parte experimental
 E.T.M./2014 – revisão
 E.T.M./2015 – revisão do texto
 E.T.M./2016 – revisão