

# Experiência 2

## Transmissão Serial Assíncrona

### Dicas e Recomendações

PCS3645

## Experiência 2

- Comunicação serial (padrão RS232C)

- Envio serial de *bits* por uma linha de comunicação

- Start bit
- Dado
- Paridade
- Stop bit

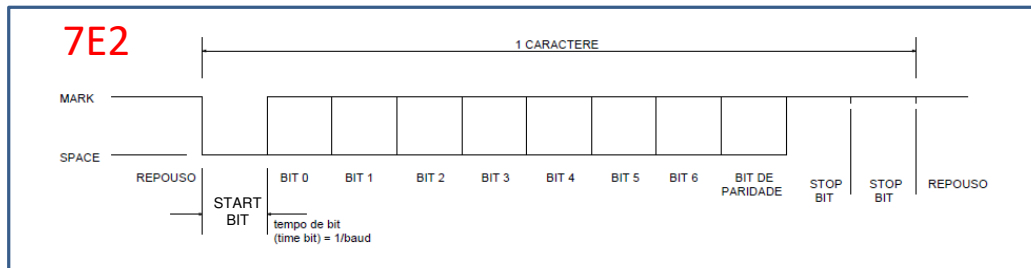


- Parâmetros:

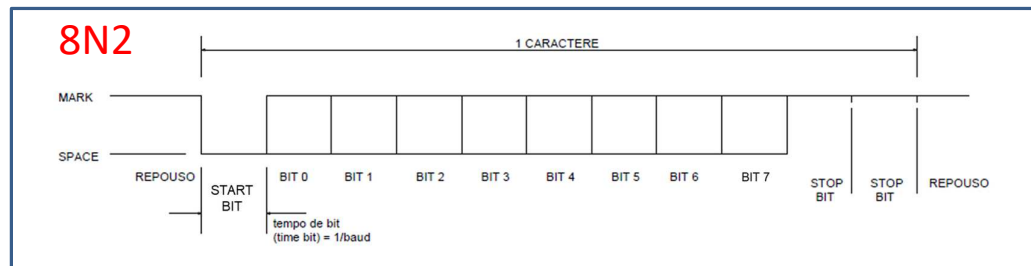
- Número de *bits* de dado (7 *bits*, 8 *bits*)
- Paridade (par, ímpar, nenhum)
- Número de *stop bits* (1, 2)

# Experiência 2

- Transmissão serial com RS232C



Projeto fornecido



Projeto do grupo

## Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

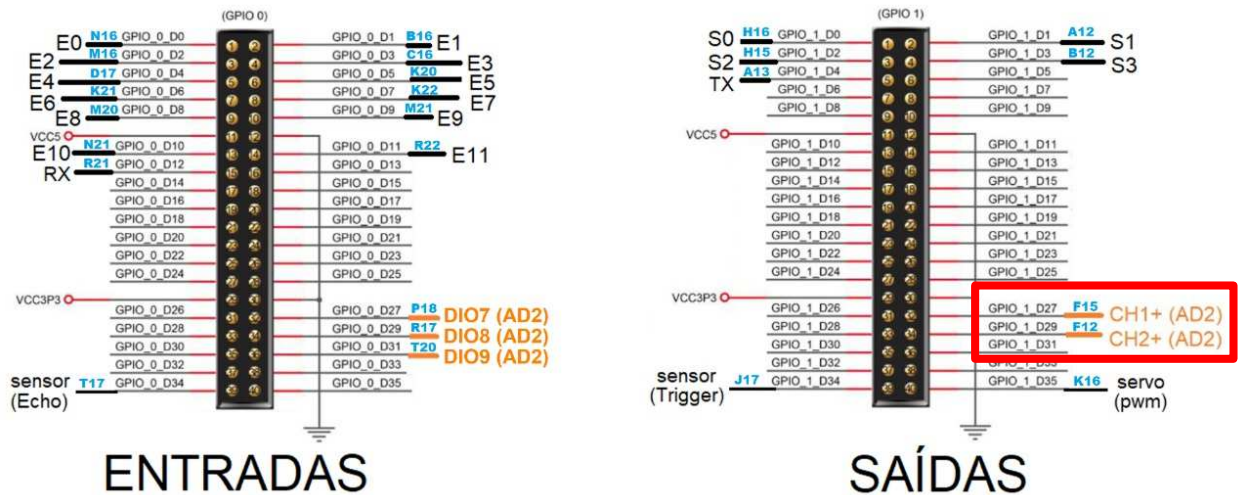
### PARTE 1: verificação com ferramenta *Scope*

- controle das entradas do circuito via MQTT Dash
- **saída serial** deve ser designada ao pino da GPIO que é conectada ao **canal CH1+ (osciloscópio)**
- as formas de onda devem ser verificadas para todos os valores de entrada usando a ferramenta *Scope* do Waveforms
- Ao final, o grupo terá a garantia que o circuito gera as formas de onda conforme a especificação da transmissão serial (8N2, 9600 *bauds*).

# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 1: verificação com ferramenta *Scope*

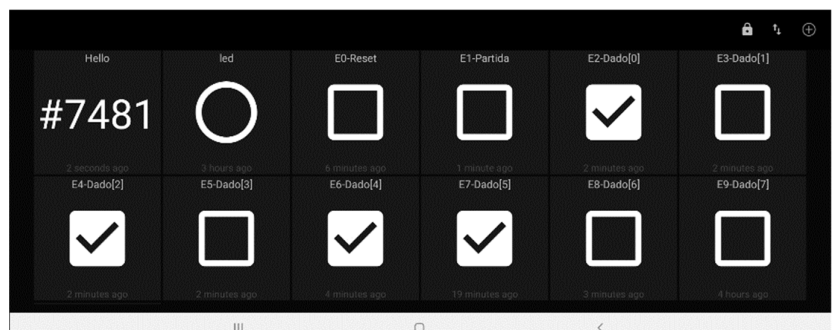


## Teste do circuito de transmissão

- MQTT Dash

controle das entradas

- reset
- partida
- dado ASCII



# Teste do circuito de transmissão

- Ferramenta *Scope*

captura de formas de onda e medidas de temporização

- canal CH1+
- ajustes (tempo e tensão)
- ajustes de *trigger*
- ferramentas de medida (cursors)



## Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

### PARTE 2: verificação com ferramenta *Protocol*

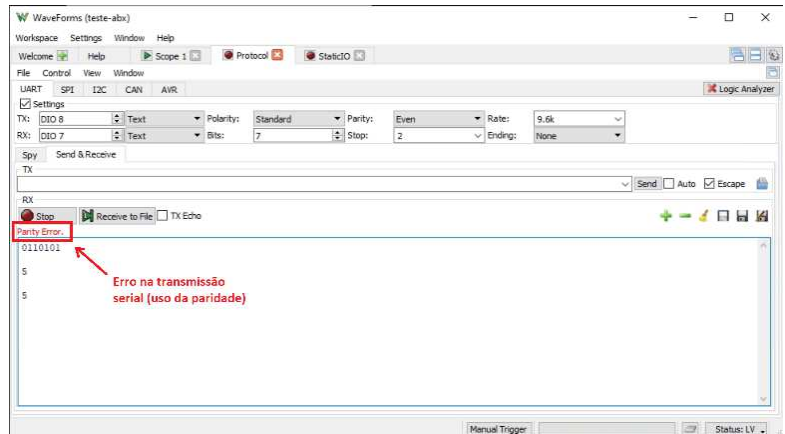
- controle das entradas do circuito via MQTT Dash
- **saída serial** deve ser designada ao pino da GPIO que é conectada ao **canal digital DIO7** e analisada com a ferramenta **Protocol (analisador de protocolos)** configurada para UART
- Ao final, o grupo terá a garantia que as formas de onda geradas pelo circuito conforme são interpretadas corretamente pelo analisador de protocolos.

# Teste do circuito de transmissão

- Ferramenta *Scope*

captura de formas de onda e interpretação do sinal serial

- canal digital DIO7
- configuração RS232-C
  - 8N2
  - 7E2
- análise da comunicação



## Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

### PARTE 2: verificação com ferramenta *Protocol*

– Testes com configuração 8N2

- Envio de diversos dados ASCII de 8 bits.
- Interpretação do dados pelo Analisador de Protocolos
  - Dados em formato Binário
  - Dados em formato Texto

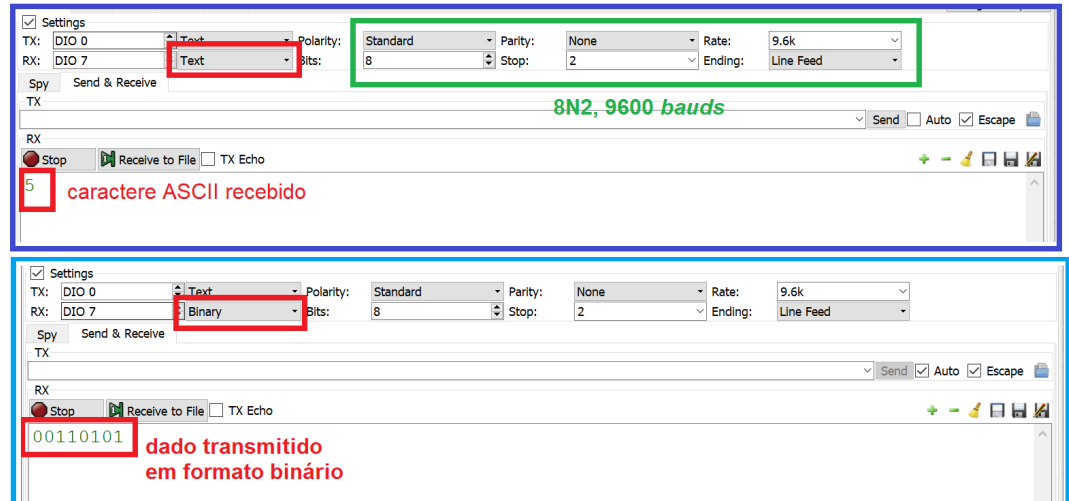
# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 2: verificação com ferramenta *Protocol*

– Análise da comunicação serial

exemplo:  
dígito 5  
código ASCII 35H



# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 2: verificação com ferramenta *Protocol*

– Testes com configuração 7E2

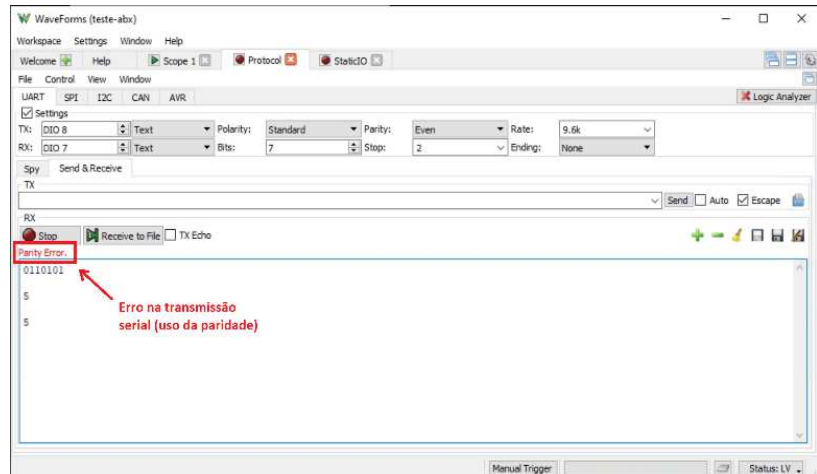
- Comunicação serial configurada com dados ASCII de 7 bits e paridade par.
- Envio dos bits seriais pelo circuito da experiência (8N2).
- Interpretação do dados seriais pelo Analisador de Protocolos
  - Dados com paridade correta
  - Dados com paridade incorreta

# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 2: verificação com ferramenta *Protocol*

- Análise da comunicação serial (erro de paridade)



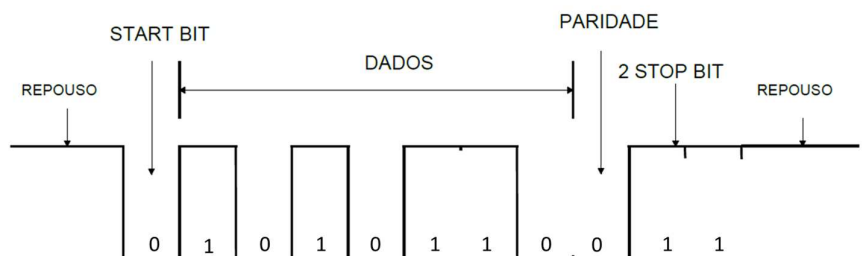
# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 2: verificação com ferramenta *Protocol*

Como usar o circuito 8N2 para testar uma comunicação 7E2?

- 7E2 e 8N2: mesmo número de bits (11 *bits*)
- Exemplo: dado ASCII 35H
- Para uma transmissão 7E2 correta basta enviar dado de 8 bits: 00110101



dado 35H com configuração 7E2

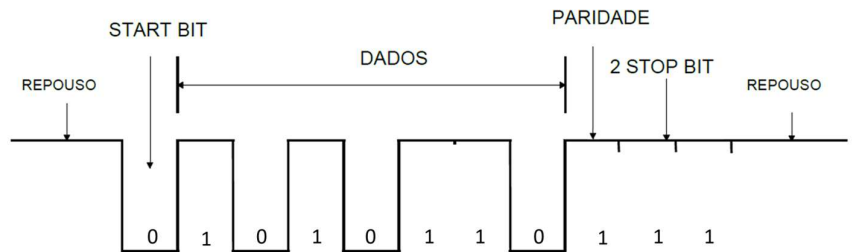
# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 2: verificação com ferramenta *Protocol*

Como usar o circuito 8N2 simular um erro na comunicação 7E2?

– Exemplo: dado ASCII 35H



dado 35H com configuração 7O2

- Para uma transmissão 7E2 incorreta com bit de paridade ímpar basta enviar dado de 8 bits: 10110101

# Dicas para a Parte Experimental

- Atividade 3 – Implementação do Projeto na Placa FPGA DE0-CV

## PARTE 2: verificação com ferramenta *Protocol*

Como usar o circuito 8N2 simular um erro na comunicação 7E2?

– Exemplo: dado ASCII 35H

Para simular uma outra transmissão 7E2 incorreta basta enviar dado de 8 bits: 00110100 (erro de transmissão do *bit* menos significativo do dado)

O que ocorre se 2 *bits* forem transmitidos com erro?



# Experiência 1

- Atividade 4 – Desafio
  - Modificação proposta pelo professor.