

Experiência 1

Controle de um Servomotor

PCS3645

PCS3645 - ETM / 2020

1

Experiência 1

- Controle de um servo motor



PCS3645 - ETM / 2020

2

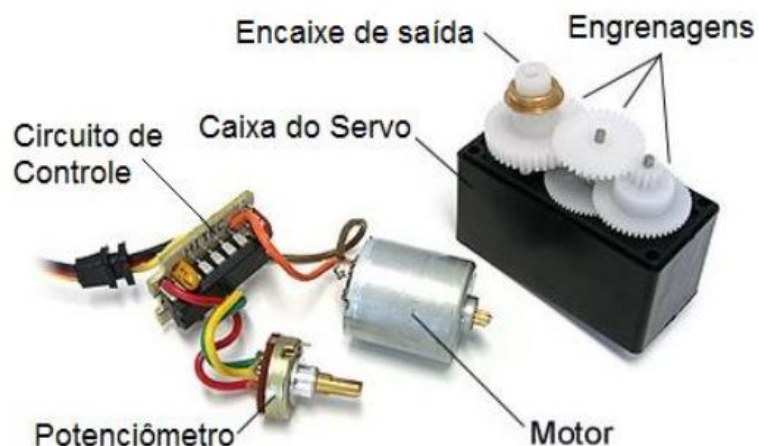
Experiência 1

- Controle de um servo motor



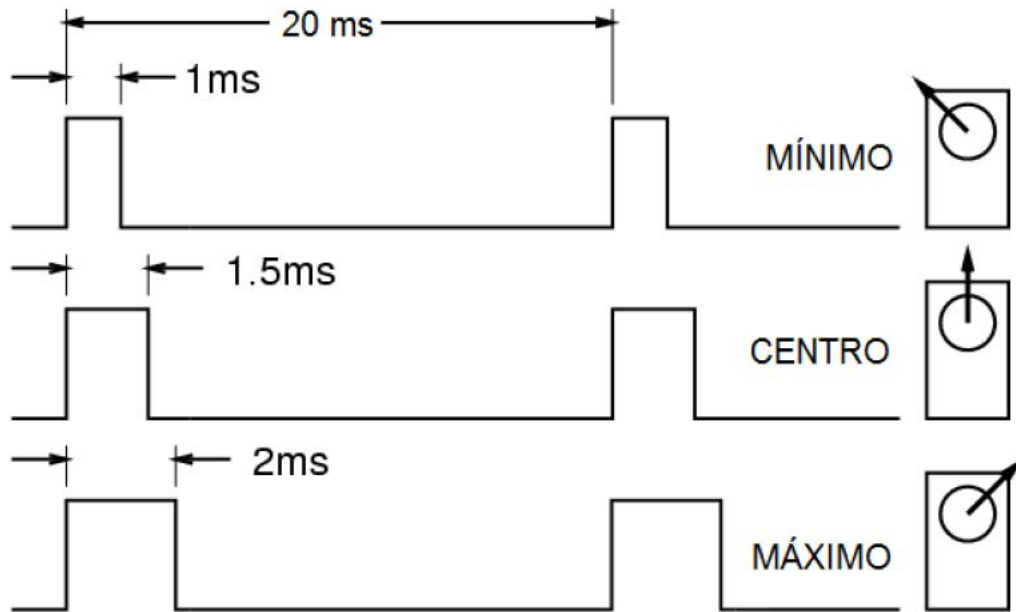
Experiência 1

- Controle de um servo motor



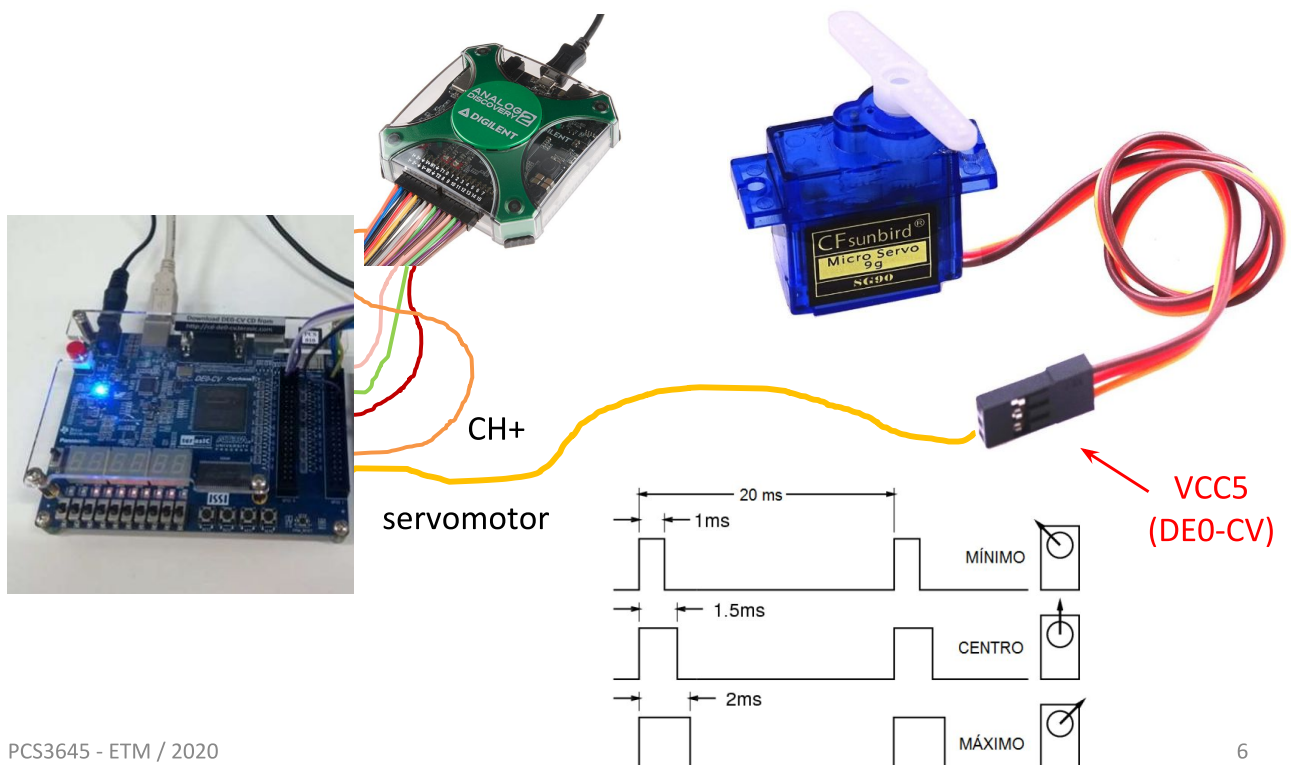
Experiência 1

- Controle de um servo motor



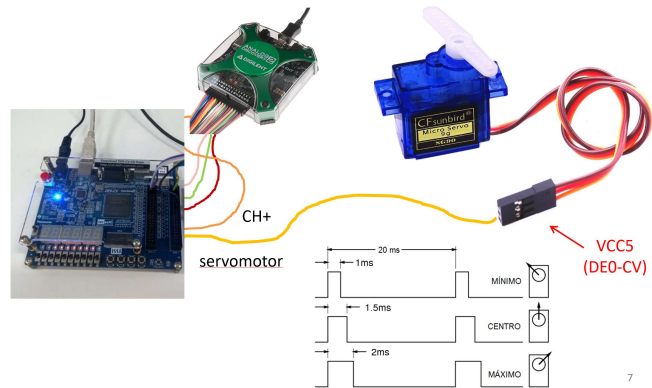
Experiência 1

- Atividade 1 – Familiarização



Experiência 1

- Atividade 1 – Familiarização



Usar designação abaixo:

Sinal	Ligação na placa FPGA	Pino na FPGA	Analog Discovery
clock	CLK_50	PIN_M9	-
reset	GPIO_0_D27	PIN_P18	DIO0
largura[0]	GPIO_0_D29	PIN_R17	DIO1
largura[1]	GPIO_0_D31	PIN_T20	DIO2
pwm	GPIO_1_D27	PIN_F15	CH1+

} sinais gerados pelo Analog Discovery
saída no osciloscópio

Experiência 1

- Atividade 1 – Familiarização

– circuito_pwm.vhd

– Estudo

– Síntese na DE0-CV

– Formas de onda no Analog Discovery

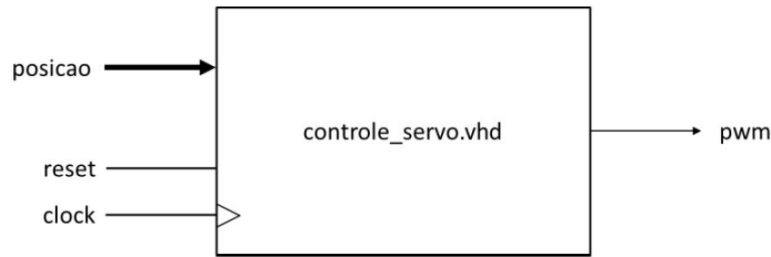
```

1  -- circuito_pwm.vhd
2  --
3  -- gera saída com modulação pwm
4  -- parametros: CONTAGEM_MAXIMA e largura_pwm
5  -- (clock a 50MHz ou 20ns)
6
7  library ieee;
8  use ieee.std_logic_1164.all;
9  use ieee.numeric_std.all;
10
11 entity circuito_pwm is
12 port (
13   clock : in std_logic;
14   reset : in std_logic;
15   largura : in std_logic_vector(1 downto 0);
16   pwm : out std_logic;
17 end circuito_pwm;
18
19 architecture rtl of circuito_pwm is
20
21   constant CONTAGEM_MAXIMA : integer := 1250; -- 4Khz ou 25us
22   constant CONTAGEM_MAXIMA : integer := 100; -- teste
23
24   signal contagem : integer range 0 to CONTAGEM_MAXIMA-1;
25   signal largura_pwm : integer range 0 to CONTAGEM_MAXIMA-1;
26   signal largura_pwm : integer range 0 to CONTAGEM_MAXIMA-1;
27
28 begin
29
30 process (clock, reset)
31 begin
32   if (reset='1') then
33     contagem <= 0;
34     pwm <= '0';
35   elsif (rising_edge(clock)) then
36     saida
37     if (contagem < largura_pwm) then
38       pwm <= '1';
39     else
40       pwm <= '0';

```

Experiência 1

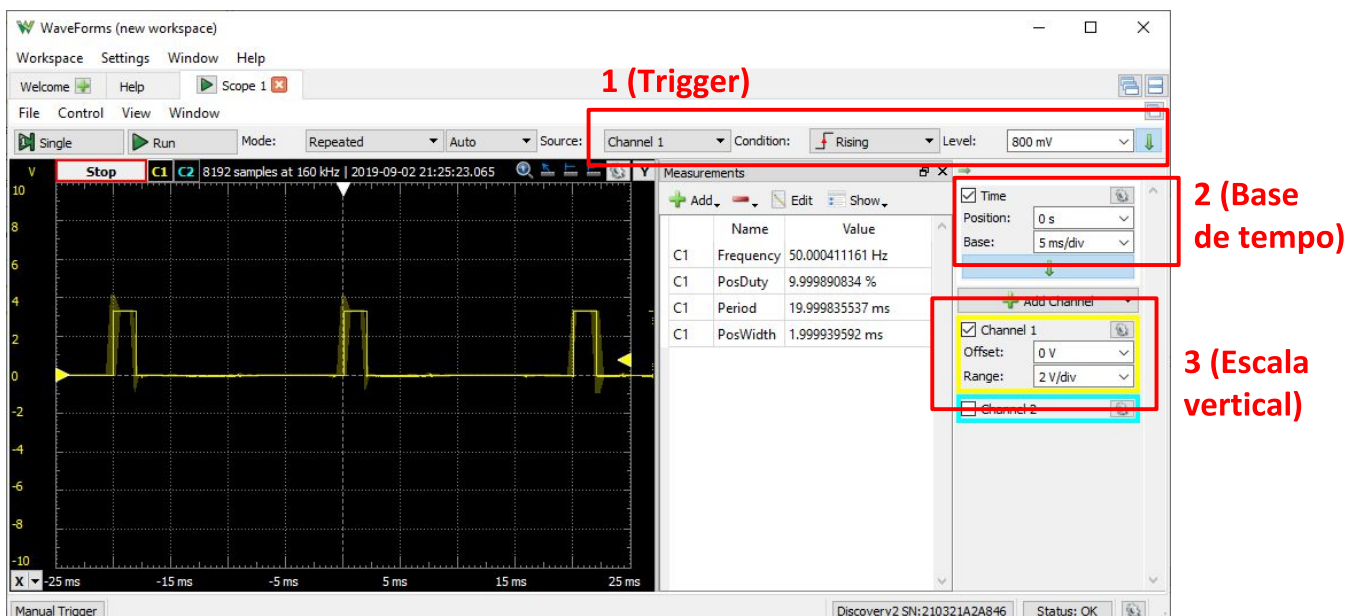
- Atividade 1 – Familiarização



- Descrição baseada no código fornecido
- Síntese na DE0-CV
(designação da saída pwm para a entrada CH1+)
- Formas de onda no Analog Discovery

Experiência 1

- Atividade 1 – Familiarização
 - Formas de onda no Analog Discovery



Experiência 1

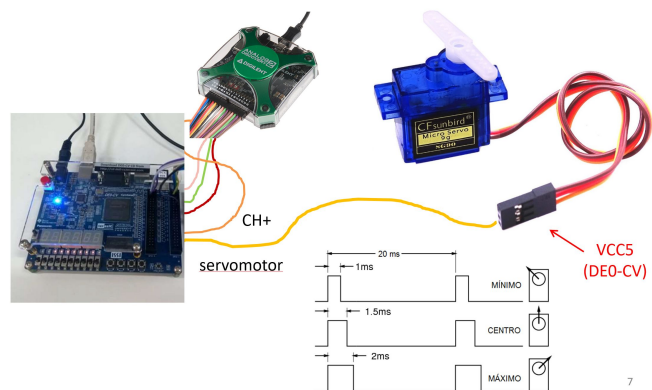
- Atividade 2 – Verificação do Funcionamento

- Ligação da saída `pwm` no servomotor
- Controle do circuito com Analog Discovery
- Verificar posicionamento do servo motor em função da entrada `posição []`

– **Cuidados:** pode travar ou queimar o servo se as formas de onda estiverem fora da especificação.

Experiência 1

- Atividade 2 – Verificação do Funcionamento



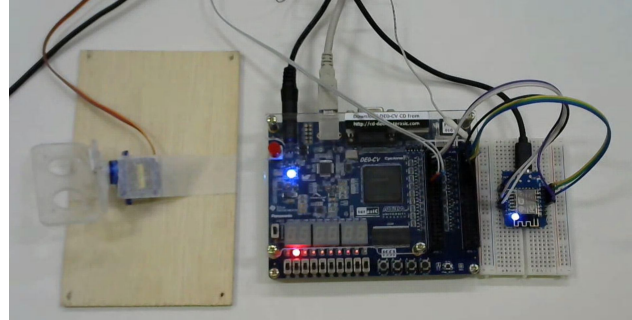
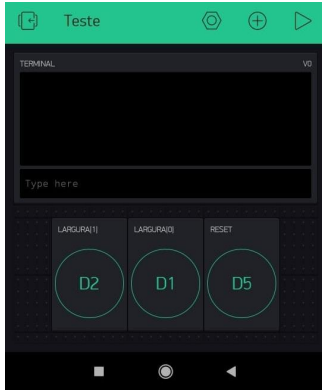
Usar designação abaixo:

Sinal	Ligação na placa FPGA	Pino na FPGA	Analog Discovery
clock	CLK_50	PIN_M9	-
reset	GPIO_0_D27	PIN_P18	DIO0
largura[0]	GPIO_0_D29	PIN_R17	DIO1
largura[1]	GPIO_0_D31	PIN_T20	DIO2
pwm	GPIO_1_D3	PIN_B12	-

saída no servomotor

Experiência 1

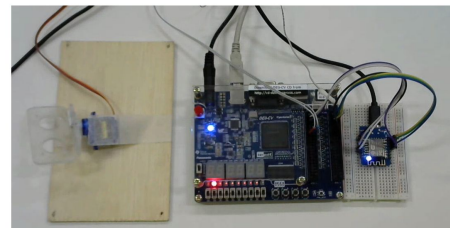
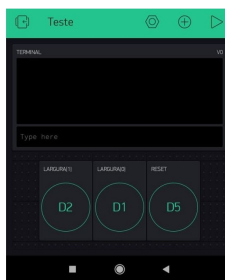
- Atividade 3 – Controle com Blynk



- Nova designação dos sinais RESET e LARGURA.
- Webcam: acompanhar posição do servomotor.

Experiência 1

- Atividade 3 – Controle com Blynk



Usar designação abaixo:

Sinal	Ligação na placa FPGA	Pino na FPGA	ESP8266 (Blynk)
clock	CLK_50	PIN_M9	-
reset	GPIO_0_D2	PIN_M16	D5
posicao[0]	GPIO_0_D0	PIN_N16	D1
posicao[1]	GPIO_0_D1	PIN_B16	D2
pwm	GPIO_1_D3	PIN_B12	-

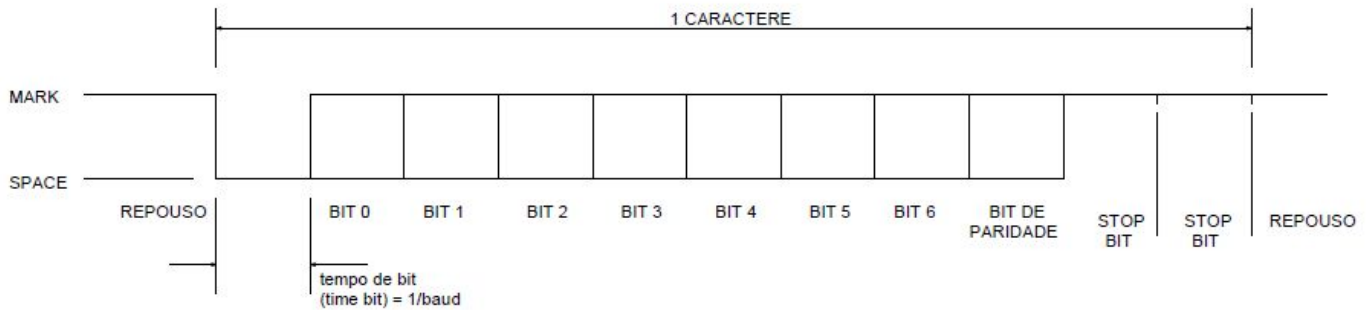
Experiência 1

- Atividade 4 – Desafio
 - Modificação proposta pelo professor.

Experiência da Próxima Semana

Experiência 2

- Transmissão serial com RS232C



Especificação: 7E2

Comunicação Serial Assíncrona

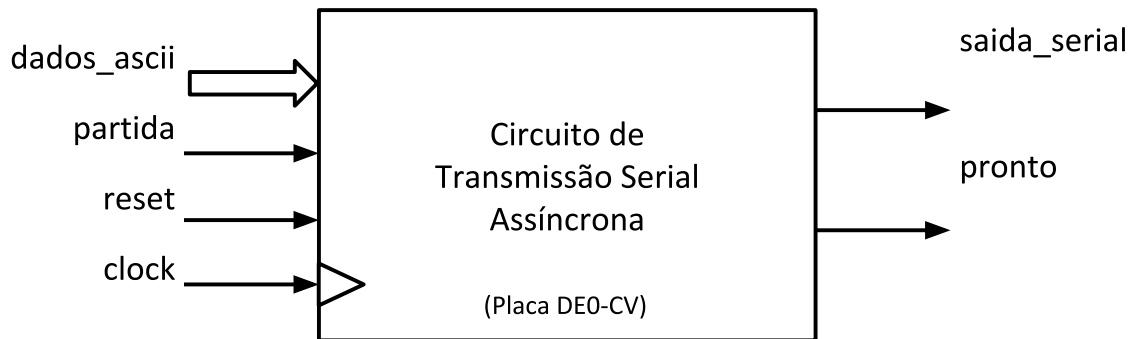
- Tabela ASCII

5 □ $0110101_2 = 35H$

Bits				b ₆	0	0	0	0	1	1	1	1
				b ₅	0	0	1	1	0	0	1	1
				b ₄	0	1	0	1	0	1	0	1
b ₃	b ₂	b ₁	b ₀									
0	0	0	0	NULL	DLE	SP	0	@	P	'	p	
0	0	0	1	SOH	DC1	!	1	A	Q	a	q	
0	0	1	0	STX	DC2	"	2	B	R	b	r	
0	0	1	1	ETX	DC3	#	3	C	S	c	s	
0	1	0	0	EDT	DC4	\$	4	D	T	d	t	
0	1	0	1	ENQ	NAK	%	5	E	U	e	u	
0	1	1	0	ACK	SYN	ε	6	F	V	f	v	
0	1	1	1	BEL	ETE	`	7	G	W	g	w	
1	0	0	0	BS	CAN	(8	H	X	h	x	
1	0	0	1	HT	EM)	9	I	Y	i	y	
1	0	1	0	LF	SUB	*	:	J	Z	j	z	
1	0	1	1	VT	ESC	+	;	K	[k	{	
1	1	0	0	FF	FS	,	<	L	\	l	:	
1	1	0	1	CR	GS	-	=	M]	m	}	
1	1	1	0	S0	RS	.	>	N	^	n	~	
1	1	1	1	S1	US	/	?	O	_	o	DEL	

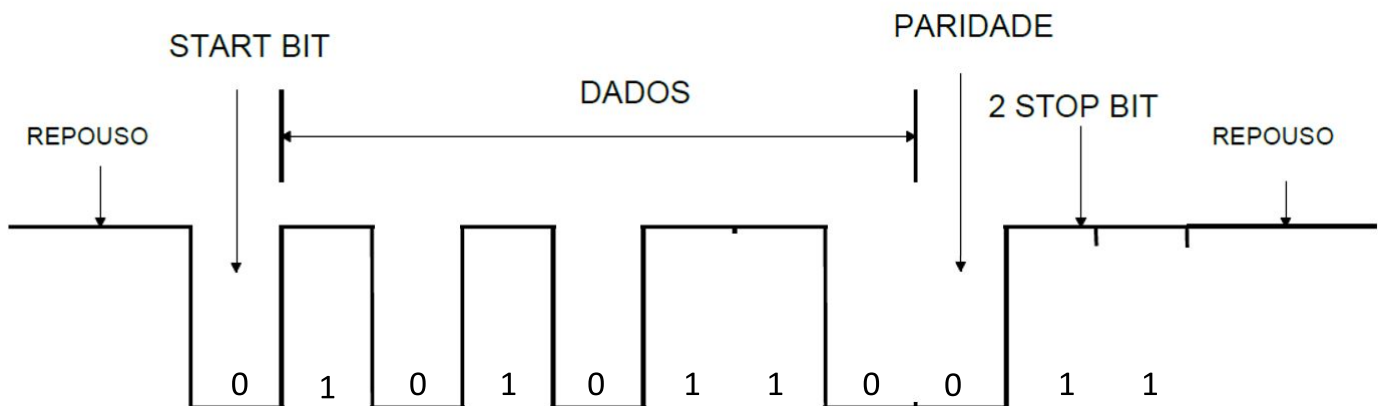
Experiência 2

- Especificação (Transmissão Serial Assíncrona)



Experiência 2

- Transmissão serial com RS232C
 - Dado ASCII 35H = 0110101_2



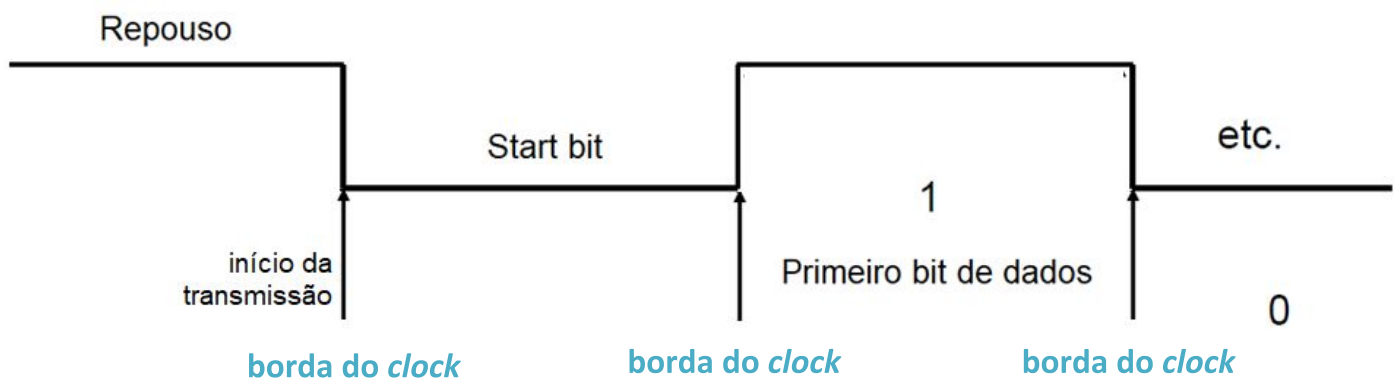
Especificação: 7E2

Experiência

- Atividades
 - Ativ. 1 - Estudo do projeto base de tx serial
 - Simulações para entender o funcionamento
 - Ativ. 2 – Adaptação para usar clock de 50MHZ
 - Projeto - uso da técnica de superamostragem (*oversampling*) para um *clock* de 50MHZ
 - Ativ.3 – Síntese e Testes
 - Controles via Analog Discovery
 - Controles via Blynk

Transmissão Serial

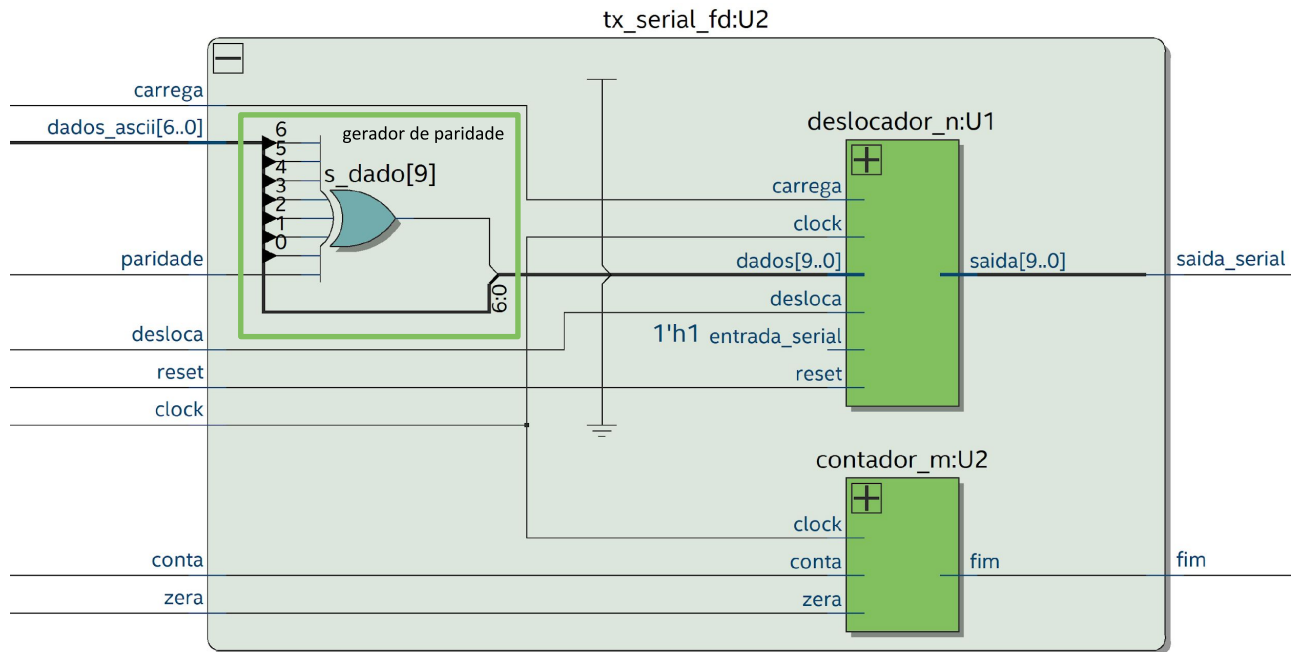
- Forma de onda na Transmissão Serial



Clock = taxa de transmissão
(p.ex. 9600 *bauds*)

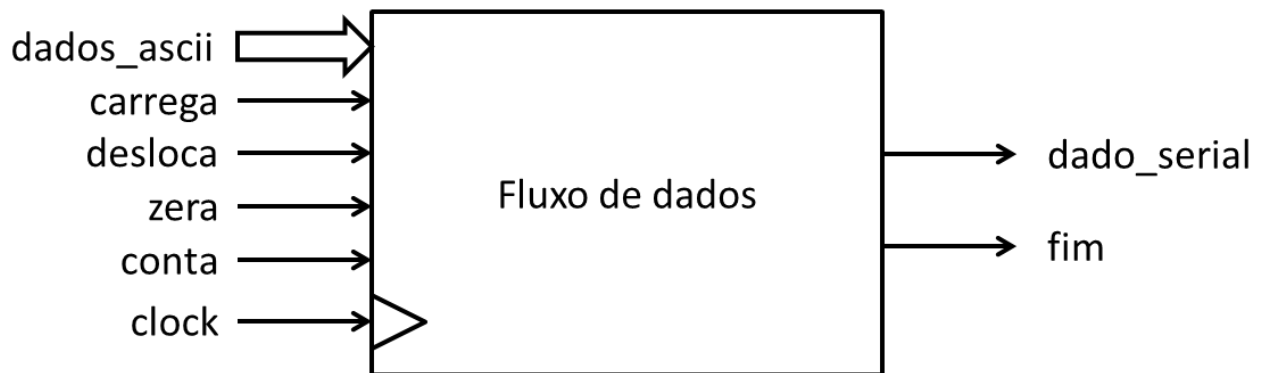
Comunicação Serial Assíncrona

- Fluxo de Dados (FD)



Comunicação Serial Assíncrona

- Especificação do circuito (FD)



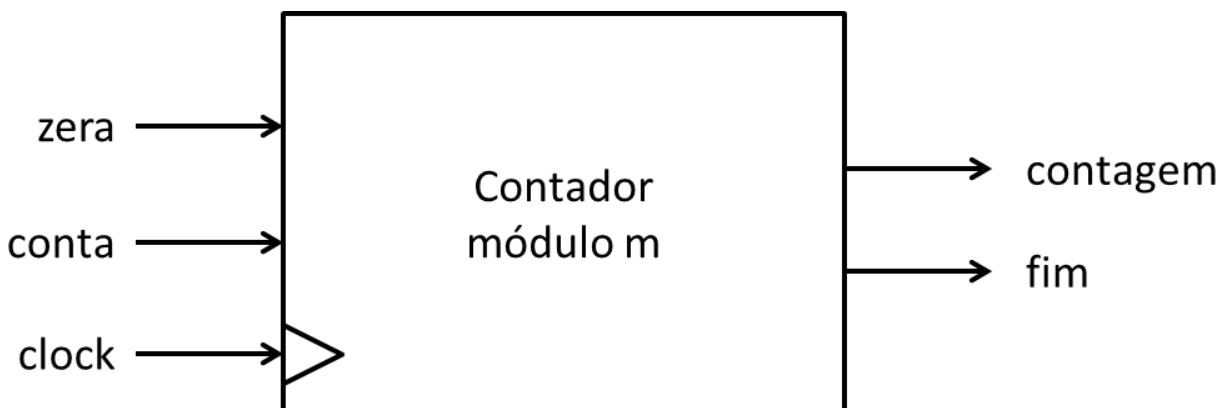
Comunicação Serial Assíncrona

- Especificação do circuito (FD)



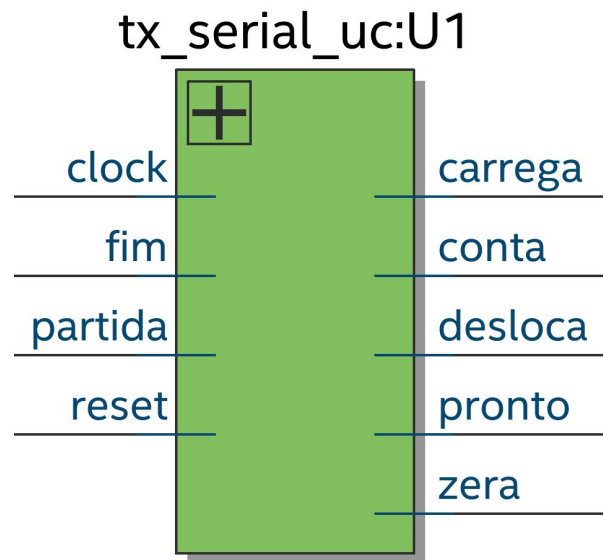
Comunicação Serial Assíncrona

- Especificação do circuito (FD)



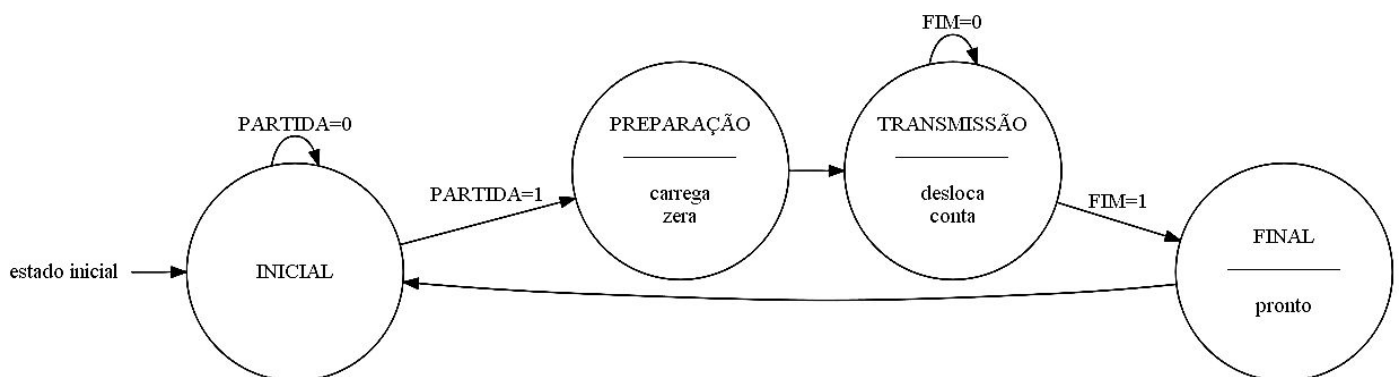
Comunicação Serial Assíncrona

- Unidade de Controle (UC)



Comunicação Serial Assíncrona

- Especificação do circuito (UC)



Comunicação Serial Assíncrona

- Comunicação com superamostragem

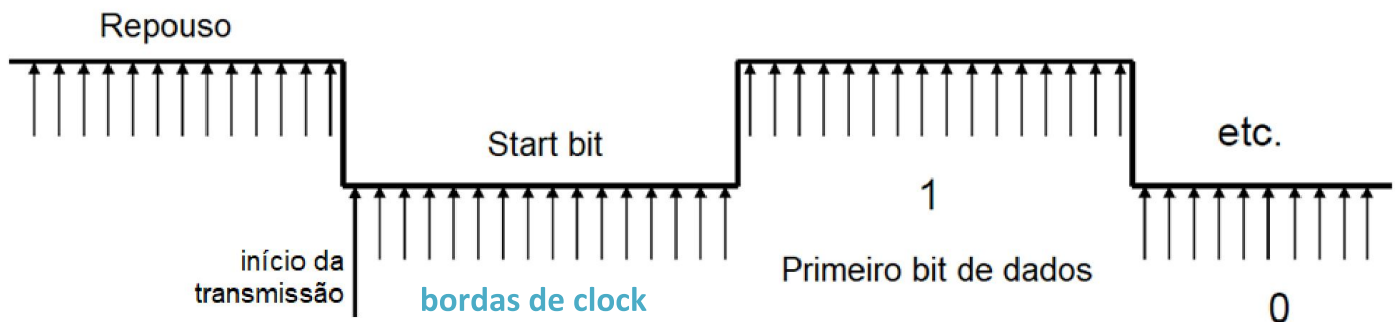
“Modificar o projeto para usar o *clock* interno de 50MHz da placa DE0-CV”

– Unidade de Controle

- Unidade de controle trabalha com clock de 50MHz;
- Transmissão deve ser realizada na taxa de comunicação do terminal serial (p.ex. 110 *bauds*)

Transmissão Serial

- Superamostragem na Transmissão Serial



Solução: criar um sinal de *tick* que indica o momento certo para transmitir um novo *bit*.

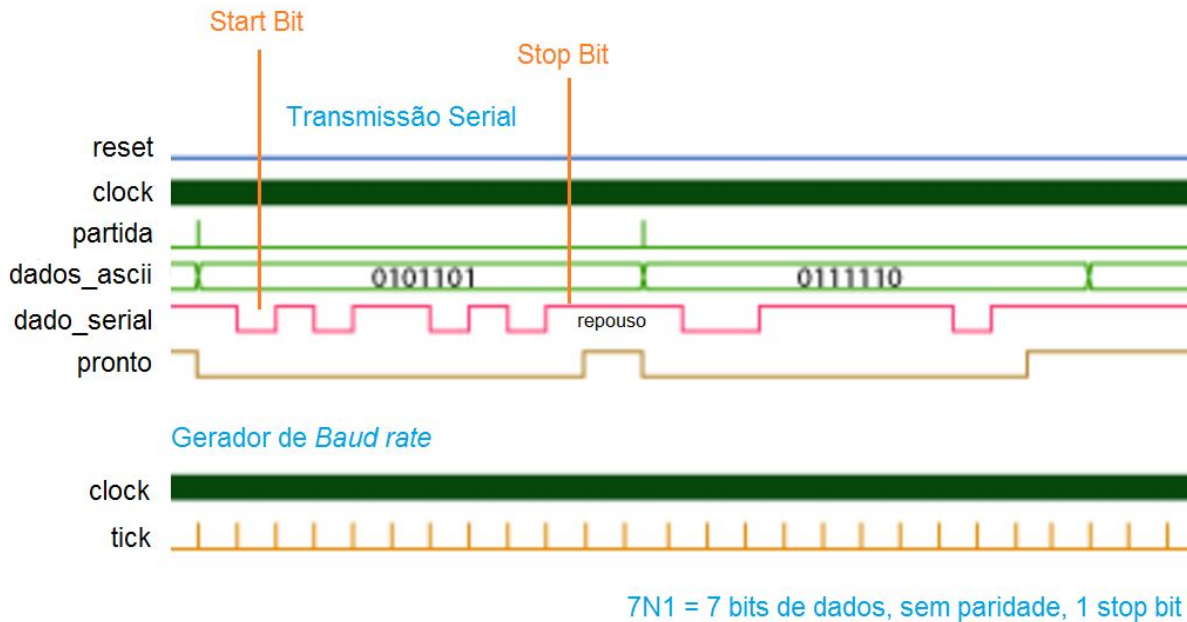
$\left\{ \begin{array}{l} \text{Clock} = 50\text{MHz} \\ \text{Baudrate} = 9600 \text{bauds} \end{array} \right.$



1 tick a cada 5.208 bordas de clock

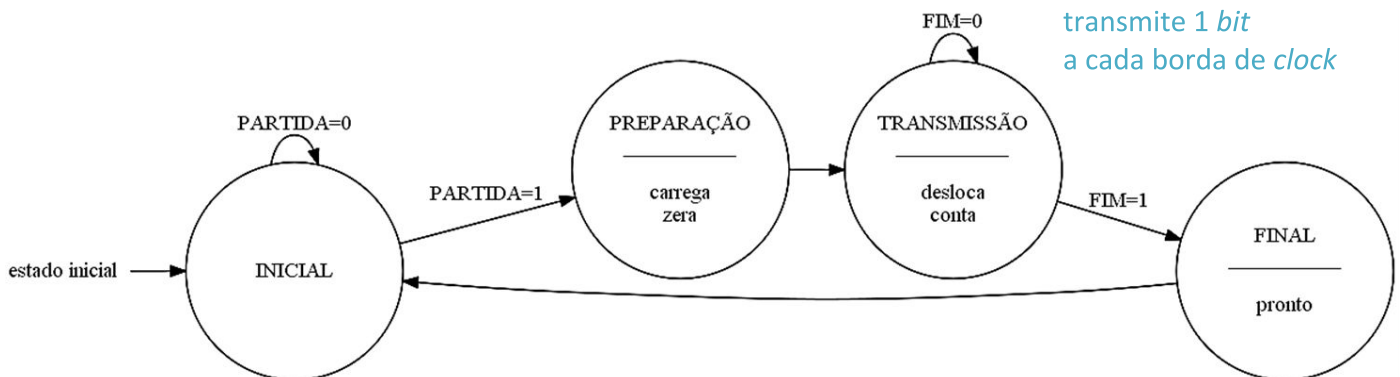
Comunicação Serial Assíncrona

- Superamostragem na Transmissão Serial
 - Formas de onda (relação *clock* e *tick*)



Comunicação Serial Assíncrona

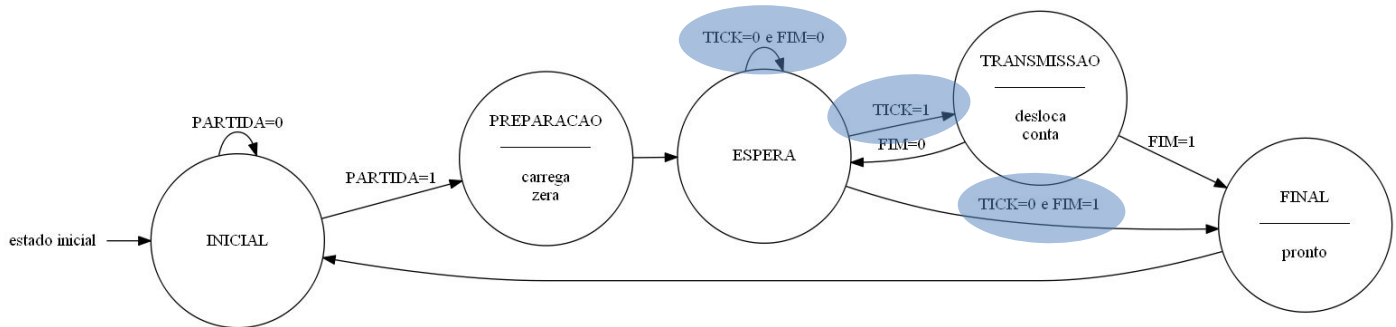
- Modificação na UC
 - FSM da Unidade de Controle original



Como adaptar a UC para transmitir os bits a 110 *bauds*, operando com *clock* de 50MHz?

Comunicação Serial Assíncrona

- Modificação na UC
 - FSM da Unidade de Controle

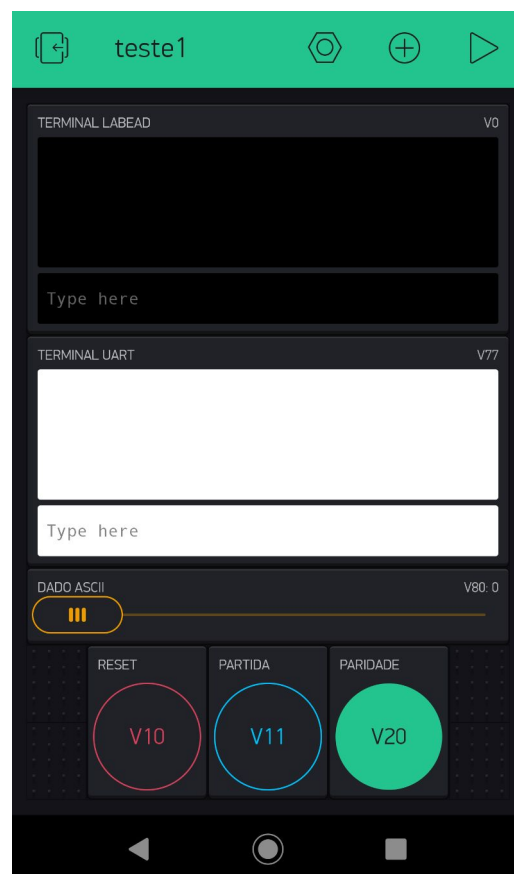


Como o sinal de *tick* é reiniciado a cada transmissão?

Sinal TICK é acionado na frequência de comunicação serial
⇒ 1 *tick* a cada $\frac{1}{110}$ seg.

Comunicação Serial Assíncrona

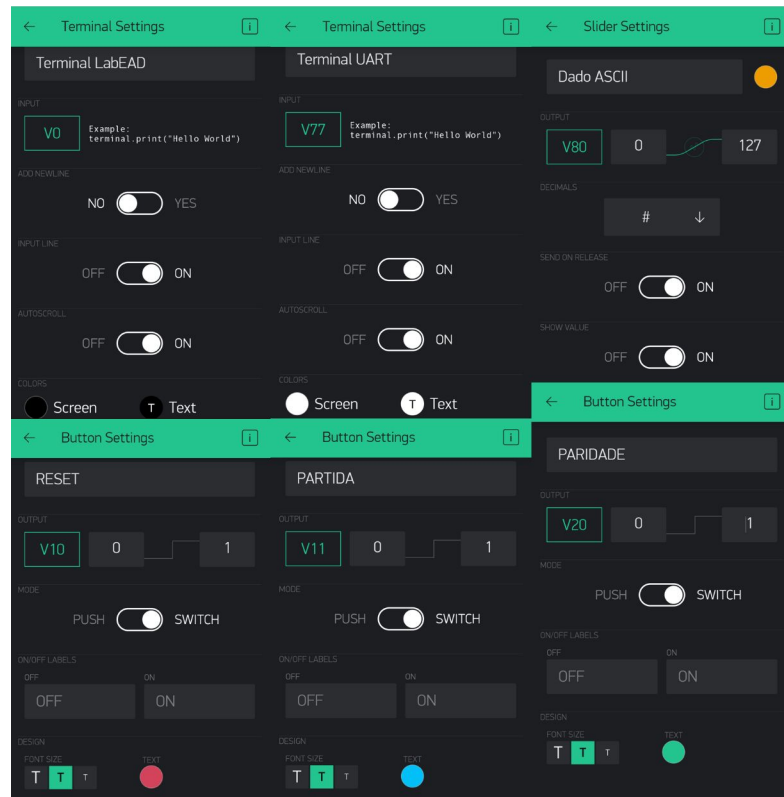
- Projeto do Blynk
 - Tela principal (*widgets*)
 - Reusar projeto da experiência
 - Preparar antes da aula



Comunicação Serial Assíncrona

- Projeto do Blynk (configurações dos *widgets*)

Atenção na
configuração
do projeto



PCS3645 - ETM / 2020

35

Planejamento

- Prazos para submissão no Moodle USP
 - Turmas de 3ª feira – 14/09 (segunda) às 12h
 - Turmas de 4ª feira – 15/09 (terça) às 12h

PCS3645 - ETM / 2020

36