

Exemplo de Aplicação de Buzzer

Categoria: EXEMPLOS COM CPLD Publicado em QUINTA, 19 SETEMBRO 2013 16:31 Escrito por ERNANI RODRIGUES DE STHIAGO Acessos: 1851



Introdução

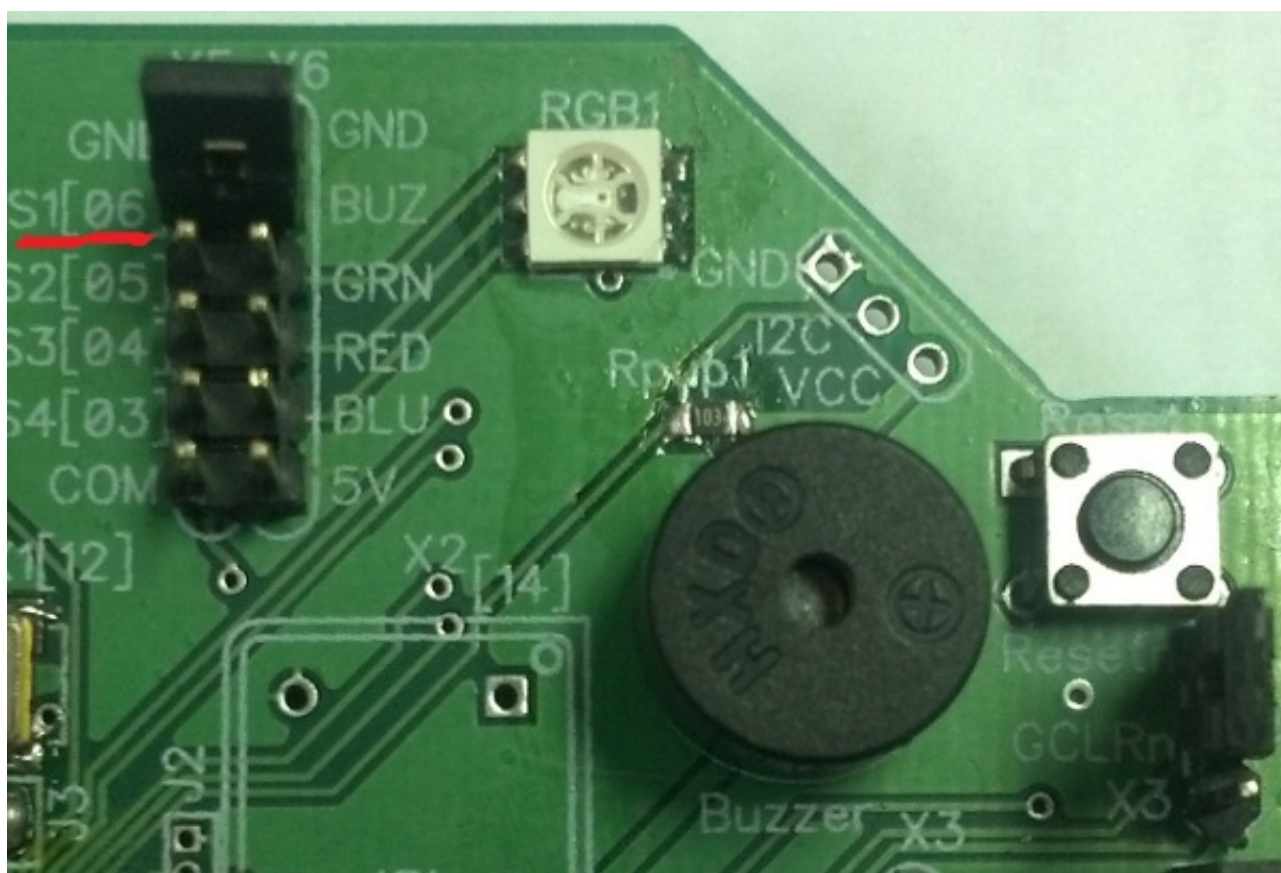


Figura 1

Buzzer é um dispositivo de áudio que pode ser mecânico, eletrônico ou piezoelétrico ([Wikipedia](#)). Entre diversas aplicações, *buzzers* são principalmente utilizados como *beeps* de teclado, por exemplo: Quando o usuário entra com um número no teclado de um sistema de alarme, o *buzzer* vibra para gerar um som e anunciar que a tecla foi “lida”. Existem diversos tipos de *buzzers* e suas diferenças não são triviais, uma das características que deve-se preocupar ao utilizar um *buzzer* é a presença ou não de um oscilador interno. O objetivo deste artigo é produzir um exemplo típico de aplicação de *buzzer* (em VHDL) e utilizá-lo como componente simples para outros projetos.

Componente utilizado: **Kit Max V** com *buzzer* de oscilador interno.

Desenvolvimento e Funcionamento

Teste do Buzzer no Kit MAXV



0:53 / 0:53



Neste projeto, apenas se desenvolveu-se um sistema composto por um botão de função *reset*, outro de função *enable* e um *prescaler*. Basicamente, utilizamos o *prescaler* para dividir o sinal de origem do kit (de 24MHz) para obter um sinal de frequência próximo de 440Hz (equivalente a nota musical “La”). Duas, importantes premissas do projeto são: O *buzzer* deverá produzir som quando “*enable* = '0'”, e a função do botão *reset* não deve ser visível ao usuário (a não ser que este esteja pressionando os botões de *reset* e *enable* ao mesmo tempo), pois sua única função é alterar os valores do contador da saída para zero.

Gerando o *clock* de 440Hz

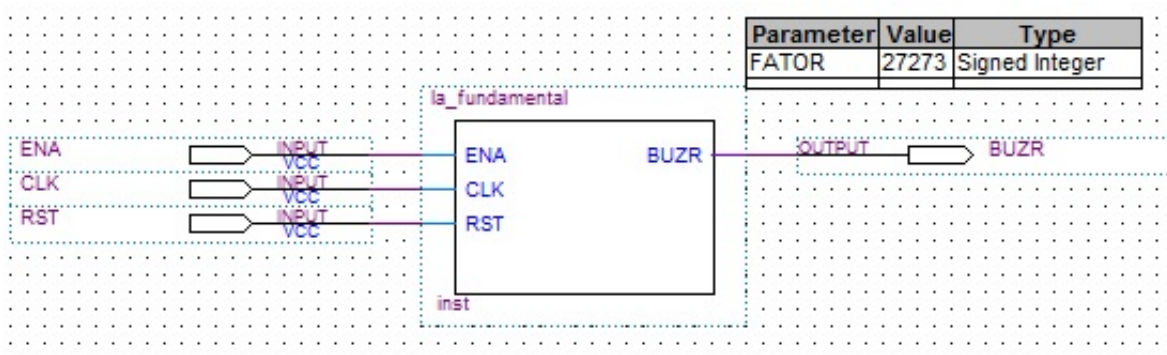


Figura 2

Para gerar o sinal de clock de 440 Hz, um circuito conta os pulsos de clock, até atingir um valor fixo denominado FATOR, e então a saída é invertida. Em dois ciclos de contagem, se gera um período do sinal de saída. Assim, para gerar um período do sinal de 440 Hz na saída, deve-se satisfazer à equação:

$$1/440 \text{ Hz} = 2 * FATOR * (1 / F_{\text{clock}}) \Rightarrow FATOR = \sim 27272,2727272727 \text{ (aproximadamente)} = \sim 27273$$

A implementação do *clock* de 440HZ, foi feita da seguinte:

(sendo *CLK* o *clock* do kit e *cnt* uma variável contadora)

```

01. if rising_edge(CLK) then
02.   cnt := cnt + 1;
03.
04.   if cnt = FATOR then
05.     cnt := 0;
06.     CLK440 := not CLK440;
07.   end if;
08.
09. end if;

```

Dessa forma, atribuímos a *CLK440* a frequência da nota La baseado no *clock* do kit.

Implementando a funcionalidade *enable*

Na Introdução, definimos que *CLK440* só seria gerado se “*ENA* = '0'” (*ENA* = *enable*), portanto a implementação do *enable* se dá apenas uma verificação adicional do botão a que atribuímos a função de *enable*:

```

01. if ENA = '0' then
02.   if rising_edge(CLK) then
03.     cnt := cnt + 1;
04.     if cnt = FATOR then
05.       cnt := 0;
06.       CLK440 := not CLK440;
07.     end if;
08.   else
09.     CLK440 := '0';
10.   end if;

```

Observe que quando *CLK440* é zero não há som.

Implementado a funcionalidade de *reset*

Agora, apenas acrescenta-se outra verificação para implementar a funcionalidade de *reset*:

```

01. if RST = '0' then
02.   CLK440 := '0';
03.   cnt := 0;
04.
05. elsif ENA = '0' then
06.   --gera La--
07. else
08.   CLK440 := '0';
09. end if;

```

Circuito de Saída

A frequência da saída *CLK440*, define como será o som emitido pelo *buzzer*, lembrando que *enable* deve estar ativo e *reset* não ativo para *CLK440* ser diferente de zero.

Para finalizar a implementação do VHDL, deve-se levar o valor de *CLK440* a saída *buzzer*:

```
1. BUZR <= CLK440;
```

A pinagem do projeto possui apenas dois botões e um *buzzer* de saída, para o **kit MAX V do projeto FPGA para Todos** pinos selecionados foram os seguintes (baseado na posição do *jumper*):

Node Name	Direction	Location	I/O Bank	Fitter Location	I/O Standard	Reserved	Current Strength	Differential Pa
BUZR	Output	PIN_6	1	PIN_6	3.3-V LV...default)		16mA (default)	
CLK	Input	PIN_12	1	PIN_12	3.3-V LV...default)		16mA (default)	
ENA	Input	PIN_27	1	PIN_27	3.3-V LV...default)		16mA (default)	
RST	Input	PIN_44	1	PIN_44	3.3-V LV...default)		16mA (default)	
<<new node>>								

Figura 3

Perceba que deve-se colocar o *jumper* entre os pinos 06 e BUZ (como mostrado na Figura 1) para receber o sinal de saída.

Conclusão

Este artigo foi produzido com o intuito de demonstrar um exemplo de uso do *buzzer* do **kit de CPLD MAXV do Projeto FPGA Para Todos**. O componente utilizado ocupa 6% do espaço disponível no MAXV (36 elementos lógicos) e trata-se apenas de um contador digital cujo conceito pode ser aplicável em outras situações. Pela imagem disponível no início deste artigo, é possível perceber um *jumper* no pino 06 (sublinhado em vermelho), onde é possível fazer a análise do sinal em um circuito externo. Vale ressaltar que, para este teste, utilizamos um *buzzer* que já possui oscilador interno e por isso se pusermos o sinal em VCC, este gerará ruído. Porém, o componente desenvolvido neste projeto apenas simula uma onda de 440Hz portanto, funcionaria em outro *buzzer* sem oscilador interno.

Arquivo do projeto para Download

Projeto para o kit de CPLD MAXV

Arquivo de Gravação

[< Ant](#) [Próximo >](#)
