

# Documentação do Módulo VGA\_Interface

Maurício dos Santos, Gabriel Andrade, Yago Martins, Vinícius

October 25, 2023

## 1 Introdução

Esta documentação descreve o módulo **VGA\_Interface**, que é projetado para gerar sinais de vídeo VGA e controlar a exibição de cores e posição na tela. Este módulo é útil para sistemas que desejam exibir gráficos em um monitor VGA. Ele inclui entradas e saídas para controle de cores, sincronização e posição na tela.

## 2 Entradas e Saídas

### 2.1 Entradas

- **R\_in, G\_in, B\_in** (Entradas de Cores RGB):
  - Descrição: Entradas de 8 bits representando as componentes de cores vermelha, verde e azul, respectivamente.
- **clk** (Clock):
  - Descrição: Sinal de clock para sincronizar operações no módulo.
- **rst** (Reset):
  - Descrição: Sinal de reset para redefinir o estado interno do módulo.

### 2.2 Saídas

- **R, G, B** (Saídas de Cores RGB):
  - Descrição: Saídas de 8 bits representando as intensidades das cores vermelha, verde e azul dos pixels a serem exibidos na tela.
- **HS** (Sincronismo Horizontal):
  - Descrição: Sinal de sincronismo horizontal que indica o início de cada linha na tela.
- **VS** (Sincronismo Vertical):
  - Descrição: Sinal de sincronismo vertical que indica o início de cada quadro ou campo na tela.

- **BLANK** (Blanking):
  - Descrição: Sinal que indica se a tela deve ser deixada em branco (0) ou se o conteúdo de vídeo deve ser exibido (1).
- **VGA\_SYNC** (Sincronismo do Sinal VGA):
  - Descrição: Sinal de sincronismo composto que combina os sinais de sincronismo horizontal e vertical.
- **VGA\_CLK** (Clock do Sinal VGA):
  - Descrição: Sinal de clock específico para o sinal VGA, usado para sincronizar a taxa de pixel com o dispositivo de exibição.
- **h\_pos** (Posição Horizontal):
  - Descrição: Saída que indica a posição horizontal atual na tela, representada como um número binário de 10 bits.
- **v\_pos** (Posição Vertical):
  - Descrição: Saída que indica a posição vertical atual na tela, representada como um número binário de 10 bits.
- **oAddress** (Endereço de Memória):
  - Descrição: Saída que representa o endereço de memória onde os dados de vídeo para o pixel atual estão armazenados.

### 3 Funcionamento

O módulo **VGA\_Interface** é responsável por gerar sinais de vídeo VGA e controlar a exibição de cores e posição na tela. Ele utiliza as entradas de cores RGB para determinar as cores dos pixels e gera sinais de sincronismo horizontal e vertical para controlar o início de cada linha e quadro na tela. O sinal **BLANK** indica se a tela deve ser deixada em branco ou se o conteúdo de vídeo deve ser exibido.

As saídas **h\_pos** e **v\_pos** representam as posições horizontal e vertical atuais na tela, enquanto **oAddress** representa o endereço de memória onde os dados de vídeo para o pixel atual estão armazenados.

### 4 Lógica de Controle Principal

O módulo **VGA\_Interface** incorpora uma lógica de controle principal que é fundamental para o funcionamento do sistema de exibição de vídeo VGA. Essa lógica é acionada pelo sinal de borda de subida (**posedge clk**), que é o clock do sistema, e é responsável por controlar a posição horizontal e vertical da imagem na tela.

A lógica de controle é sensível a dois sinais essenciais: `clk` e `rst`. O sinal `clk` é o sinal de clock que sincroniza todas as operações no sistema. O sinal `rst`, por sua vez, é o sinal de reset que permite reiniciar o estado interno do módulo.

O comportamento da lógica de controle é descrito da seguinte maneira:

- Quando o sinal de reset (`rst`) é igual a 1 (`1'b1`), o módulo é reiniciado. Nesse caso, as posições horizontal (`h_pos`) e vertical (`v_pos`) são redefinidas para seus valores iniciais, que são ambos zeros (10 bits de zeros).
- Quando o sinal de reset não está ativo (ou seja, `rst` é igual a 0), a lógica começa a atualizar a posição horizontal (`h_pos`) e vertical (`v_pos`). Primeiro, verifica se a posição horizontal está dentro dos limites da tela (comparando-a com `HORIZONTAL_TOTAL - 1`). Se estiver dentro dos limites, a posição horizontal é incrementada em 1 (`h_pos <= h_pos + 1'b1`).
- Se a posição horizontal atingir o limite, ela é reiniciada para zero (`h_pos <= 1'b0`), e a posição vertical passa pelo mesmo processo de verificação e atualização, mantendo-a dentro dos limites da tela (`v_pos <= v_pos + 1'b1`).
- Esse processo de atualização contínua de `h_pos` e `v_pos` permite que a imagem seja exibida na tela, movendo-se horizontal e verticalmente enquanto é varrida linha por linha e quadro por quadro.

Essa lógica desempenha um papel crítico na criação e atualização da imagem na tela VGA, garantindo que ela seja exibida corretamente e que as posições horizontal e vertical sejam mantidas dentro dos limites da tela.

## 5 Sincronismo Horizontal, Sincronismo Vertical e Geração de Blank

Este módulo compreende as lógicas de sincronismo horizontal (`HS`), sincronismo vertical (`VS`), geração de blank (`BLANK`), atribuições e geração de endereços na memória. Cada uma dessas partes desempenha um papel crucial no controle e na exibição da imagem na tela VGA.

### 5.1 Sincronismo Horizontal (`HS`)

A lógica de sincronismo horizontal é acionada pelo sinal de borda de subida (`posedge clk`). Quando o sinal de reset (`rst`) está ativo (1), `HS` é definido como 1, indicando o início de uma nova linha na tela. Durante a exibição da imagem, o sinal de sincronismo horizontal é desativado (0) quando a posição horizontal (`h_pos`) está dentro das regiões de *back porch*, *display* e *front porch*. Fora dessas regiões, `HS` é definido como 1.

### 5.2 Sincronismo Vertical (`VS`)

A lógica de sincronismo vertical é semelhante à de sincronismo horizontal. Quando o sinal de reset (`rst`) está ativo (1), `VS` é definido como 1, indicando o início de um novo quadro ou campo na tela. Durante a exibição da imagem, o sinal de sincronismo vertical é desativado (0) quando a posição vertical (`v_pos`) está dentro das regiões de *back porch*, *display* e *front porch* vertical. Fora dessas regiões, `VS` é definido como 1.

### 5.3 Blank (BLANK)

O sinal de blank (BLANK) é responsável por controlar se a tela deve ser mantida em branco (0) ou se o conteúdo de vídeo deve ser exibido (1). Durante a exibição da imagem, BLANK é desativado (0) apenas quando a posição horizontal e vertical está dentro dos limites do *display* ativo. Fora desses limites, a tela permanece em branco.

### 5.4 Atribuições e Atualização de Cores

As próximas partes deste módulo incluem atribuições de sinais de VGA\_CLK e VGA\_SYNC, que estão relacionadas ao sinal VGA. Além disso, há uma atualização contínua das cores vermelha (R), verde (G) e azul (B) com base nas entradas R\_in, G\_in e B\_in. Isso permite a exibição de cores na tela.

### 5.5 Geração de Endereços na Memória

A última parte deste módulo é responsável pela geração de endereços na memória. Quando o sinal de reset está ativo (1), o endereço é redefinido para zero. Durante a parte visível do vídeo, o endereço é incrementado continuamente, representando a posição atual na memória de vídeo. Quando o último pixel é atingido, o endereço retorna ao início. Na parte de sincronismo do vídeo, o endereço permanece inalterado.

Essas várias partes trabalham em conjunto para controlar a exibição de vídeo VGA e garantir que as cores e a posição da imagem sejam atualizadas conforme necessário para uma visualização adequada na tela.

## 6 Objetivos Alcançados

Nesta seção, descreveremos os objetivos alcançados com o módulo `VGA_Interface`, destacando suas funcionalidades e capacidades.

### 6.1 Tamanho da Imagem de Até 9 Quadros

Um dos principais objetivos alcançados é a capacidade de suportar imagens de até 9 quadros na tela. Embora não haja uma declaração explícita sobre o tamanho da imagem no código, o módulo pode acomodar tamanhos diferentes de imagem. Isso é possível ajustando as configurações dos parâmetros `H_DISPLAY` (horizontal) e `V_DISPLAY` (vertical) de acordo com as necessidades do projeto. Desde que a resolução total não exceda os limites da especificação VGA, o módulo é flexível o suficiente para acomodar diferentes tamanhos de imagem.

### 6.2 Possibilidade de Posicionar a Imagem em Qualquer Coluna/Linha

Outro objetivo alcançado é a capacidade de posicionar a imagem em qualquer coluna ou linha na tela. O código controla a posição dos pixels na tela por meio das variáveis `h_pos` (posição horizontal) e `v_pos` (posição vertical). Essas variáveis são atualizadas e sincronizadas com os sinais de sincronismo horizontal (HS) e vertical (VS). Ao ajustar os valores iniciais dessas variáveis, é possível controlar a posição precisa da imagem, permitindo um posicionamento flexível.

### 6.3 Sem Extrapolar os Limites de Exibição

O módulo também alcançou o objetivo de garantir que não ocorra extrapolação dos limites de exibição da tela VGA. Para isso, o código inclui verificações que ajustam corretamente os sinais de sincronismo horizontal e vertical. Essas verificações asseguram que a imagem seja exibida dentro dos limites especificados em `H_DISPLAY` e `V_DISPLAY`, evitando que qualquer parte da imagem se estenda além dos limites da tela.

Em resumo, o módulo `VGA_Interface` demonstrou sucesso na capacidade de acomodar diferentes tamanhos de imagem, permitir o posicionamento flexível em qualquer coluna e linha, e garantir que a imagem seja exibida dentro dos limites de exibição da tela VGA, atingindo assim os objetivos estabelecidos.

## 7 Explicação do Testbench do Módulo `VGA_Interface`

Aqui, vamos explicar o código do Testbench que foi criado para testar o módulo `VGA_Interface`. O objetivo deste Testbench é verificar se o módulo funciona conforme o esperado e identificar possíveis erros ou problemas.

### 7.1 Declaração de Variáveis

No início do Testbench, são declaradas algumas variáveis que serão usadas para testar o módulo. Isso inclui variáveis para controlar a posição horizontal e vertical, bem como os sinais de cores R, G e B. Além disso, também são declaradas variáveis para os sinais de sincronismo, o sinal de blank e outros sinais relacionados à interface VGA.

### 7.2 Instanciação do Módulo `VGA_Interface`

O módulo `VGA_Interface` é instanciado dentro do Testbench, com seus sinais de entrada e saída conectados às variáveis e sinais declarados anteriormente. Isso permite que o Testbench se comunique com o módulo para testá-lo.

### 7.3 Parâmetros de Especificações VGA

São definidos parâmetros que especificam as características da interface VGA, como os totais horizontais e verticais. Esses valores podem variar dependendo das especificações do seu sistema.

### 7.4 Geração de Sinal de Clock

O sinal de clock é gerado no Testbench com um período de 5 unidades de tempo, alternando entre 0 e 1. Isso simula o comportamento de um sinal de clock na simulação.

### 7.5 Início da Simulação

O Testbench começa definindo um sinal de reset (`rst`) e aguarda alguns ciclos de clock antes de desativá-lo. Isso permite uma inicialização controlada do módulo.

Em seguida, são simulados sinais de cores diferentes (`R_in`, `G_in` e `B_in`) para verificar se o módulo responde corretamente a diferentes entradas.

São realizadas verificações de sinais de sincronismo horizontal e vertical, bem como verificações de limites de tela. Se algum erro for detectado, uma mensagem é impressa na saída.

Um teste adicional é executado com valores de posição (`h_pos_test` e `v_pos_test`) que ultrapassam os limites da tela, e uma mensagem de erro é impressa se isso ocorrer.

Você pode adicionar mais testes e verificações conforme necessário para garantir que o módulo funcione corretamente em diferentes situações.

A simulação é finalizada após algum tempo com o comando `$finish`.