

## Hardware para Computação Gráfica

GBC204 – Computação Gráfica  
Prof. Dr. rer. nat. Daniel Duarte Abdala

## Introdução

- ▶ Para entendermos os requisitos de hardware impostos por aplicações de computação gráfica devemos primeiro entender a natureza dos dados gráficos;
- ▶ Dados gráficos podem ser divididos em dois tipos:
  - ▶ Vetoriais
  - ▶ matriciais

## Dados Vetoriais

- ▶ Dados vetoriais especificam coordenadas e regras de desenho;
- ▶ São muito mais compactos que sua contra-parte, os dados matriciais;
- ▶ São adequados para aplicações geométricas no entanto inadequados para fotorealismo;
- ▶ Em geral, cada entrada de um modelo vetorial apresenta a seguinte forma:

```
<Primitiva>
<Lista de Pontos>
<Lista de Parâmetros>
```

## Dados Vetoriais

- ▶ Ex:
 

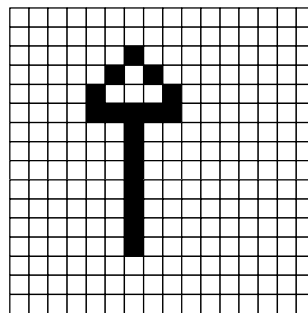
```
<Linha>
<{(0,0),(50,100)}>
```
- ▶ Ex:
 

```
<Circulo>
<{(10,10)}>
<50>
```
- ▶ No primeiro exemplo uma linha é especificada entre os pontos (0,0) e (50,100);
- ▶ No segundo exemplo um círculo é especificado com centro no ponto (10,10) e raio 50.
- ▶ A unidade dos dados do modelo é arbitrária, podendo representar centímetros, milímetros, pixels, etc.
- ▶ Os dados do modelo não estão limitados a coordenadas Inteiras.

## Dados Matriciais

- ▶ Outra forma de se representar informação gráfica se dá por meio de matrizes de elementos gráficos (pixel elements, pixels or pels);
- ▶ Forma adequada para aplicações de fotorealismo;
- ▶ Os dados em forma matricial possuem números de coluna e linhas fixos e finitos;
- ▶ São muito mais volumosos se comparados aos dados na forma vetorial;
- ▶ No entanto o volume de dados apresentado não importa, a carga é constante, diferentemente do modelo vetorial;

## Exemplo de Dado Matricial



1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

## Demanda em Memória

- ▶ Em sua forma mais restrita, dados matriciais utilizam apenas duas cores, preto e branco;
- ▶ Neste caso, apenas um bit é necessário para representar cada célula da matriz;
- ▶ Ex: imagem de 100x100 pixels  
Requer  $100 \times 100 \times 1 = 10.000$  bits ou 1250 bytes
- ▶ Em um caso mais atual, imagine uma imagem contendo 1024x768 pixels. Imagine ainda que são utilizados 24 bits para representar a cor individual de cada pixel. Neste caso serão necessários:  
 $1024 \times 768 \times 24 = 18.874.368$  bits ou 2.359.296 bytes

## Um Breve Histórico

- ▶ Desde sua concepção, computadores requereram uma forma de produzir a saída dos resultados de seus cálculos;
- ▶ Inicialmente a saída se deu por meio de luzes, depois cartões perfurados, folhas impressas e finalmente por meio de monitores;
- ▶ Os primeiros monitores eram na verdade adaptações de visores de radar;
- ▶ Eram criados a partir de um tubo de raios catódicos e uma tela de células de fósforo;
- ▶ O feixe de elétrons podia ser desviado para incidir sobre uma célula de fósforo específica por meio de indução eletromagnética;
- ▶ Tais dispositivos eram naturalmente adequados para modelos vetoriais pois para se desenhar uma primitiva qualquer bastava que se calculasse o nível de defração;

## Histórico

- ▶ No entanto a popularização de monitores como dispositivo primário de saída de informação se deu nos computadores da IBM (em especial a série 360) e os minicomputadores da NEC, tal como a série PDP;
- ▶ Embora houvesse interesse em gráficos, a preocupação primária se dava com saída de textos – devido a natureza das aplicações primárias na época e as limitações óbvias de memória frente as demandas previamente vistas;
- ▶ Outro ponto a ser considerado era o custo associado aos displays gráficos. Um modelo de display muito mais popularizado na época era a televisão, que com alguns ajustes poderia ser utilizada;

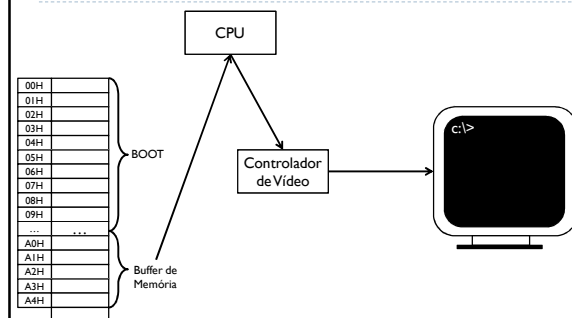
## Histórico

- ▶ No entanto o modelo de tubo de raios catódicos direcionáveis usando modelos vetoriais persistiu por um considerável tempo, principalmente para aplicações de engenharia;
- ▶ Gradativamente no entanto, o modelo matricial foi ganhando espaço e predominância:
  - ▶ Em condições controladas, apresentar requisitos moderados de memória;
  - ▶ Mais adequado para apresentação de textos que o modelo vetorial;
- ▶ Com o surgimento dos computadores pessoais em meados da década de 70 o interesse por gráficos ganhou novo ímpeto;

## Arquitetura Raster

- ▶ A arquitetura de varredura, ou raster se popularizou pelas razões descritas anteriormente;
- ▶ Ela funciona da seguinte forma:
  - ▶ Uma parcela da memória principal é reservada para espelhar a saída de dados. Caso um programa deseje apresentar informação no monitor raster, ele precisa simplesmente escrever de maneira apropriada nesta região de memória;
  - ▶ A região reservada de memória é lida por uma subrotina do SO e enviada ao controlador de vídeo;
  - ▶ O controlador traduz os bits enviados e assim ajusta de um a três feixes de elétrons que varrem continuamente o monitor;
  - ▶ A varredura do monitor é feita da esquerda para direita, e de cima para baixo em frequências de 50 ou 60 Hertz;

## Arquitetura Raster



### Buffer de Memória

- ▶ Originalmente, o buffer de memória era apenas uma região reservada da memória principal (RAM);
- ▶ Em sistemas antigos, sem hardware dedicado para computação gráfica a memória era lida por uma subrotina do SO (um rudimento de device driver) e enviada pelo barramento de dados para o controlador de vídeo;
- ▶ O montante de memória reservada varia; Por exemplo para apresentação de texto, considerando o padrão de 80 colunas e 60 linhas faz-se necessário um buffer de memória de 4,8k bytes.
- ▶ Para uma tela gráfica de de 360x480 usando uma paleta de 256 cores, são necessários 307,2k bytes.
- ▶ Problemas com o modelo:
  - ▶ Barramento ocupado durante a transmissão de dados para o controlador;
  - ▶ A CPU fica ocupada executando a subrotina de atualização de tela;

### CPU e SO

- ▶ A CPU é alocada uma vez a cada 1/60 segundos para transferir o montante reservado pelo buffer de vídeo da memória para o controlador de vídeo;
- ▶ Note que o trabalho da subrotina de atualização de tela é consideravelmente simples, copia da memória e escreve no buffer de dados;
- ▶ Esta requisição foi aliviada com a invenção da técnica de DMA (Direct Memory Access).

### Controlador de Vídeo

- ▶ O controlador de vídeo inicialmente era um chipset dedicado presente na placa mãe;
- ▶ Sua função era traduzir os bytes do buffer de memória para sinais que controlar o tubo de raios catódicos;
- ▶ O aspecto fundamental refere-se ao *timing!* no controle;
- ▶ Em um monitor monocromático, o bit correspondente à célula de fósforo sob a qual o feixe de electrons incide no momento é consultada. Caso o valor seja "0" o feixe é atenuado não excitando assim os átomos de fósforo. Caso o valor seja "1" o feixe é mantido em força excitando assim os átomos de fósforo e acendendo o pixel;

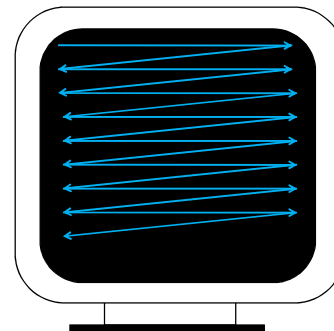
### Controlador de Vídeo

- ▶ No caso de monitores coloridos o processo é praticamente o mesmo;
- ▶ Normalmente ao invés de um único feixe de electrons, três são utilizados;
- ▶ As células de fósforo também são divididas em uma latice de três células;
- ▶ Para se controlar a cor que se deseja apresentar, a intensidade dos feixes individuais é ajustada.

### Monitor Raster (Originalmente CRT)

- ▶ Num monitor raster, o feixe de electrons fica "varrendo" a tela da esquerda para direita e de cima para baixo, 50 ou 60 vezes por segundo;
- ▶ O esquema de movimento do feixe não muda, a única parte controlável do sistema é a intensidade do feixe de electrons;

### Esquema de Varredura



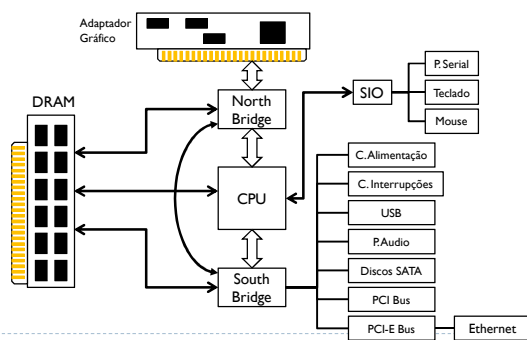
### Setup de Hardware Atual

- ▶ Em termos de dispositivos de saída pouco mudou!
- ▶ Monitores CRT raramente são utilizados. Foram substituídos por Displays LED, OLED, Plasma, etc.
- ▶ A tecnologia construtiva é diferente, no entanto eles continuam a funcionar utilizando o esquema de varredura, de 50 a 60 vezes por segundo;
- ▶ Há algumas novidades, tal como displays translucidos

### Setup de Hardware Atual

- ▶ Há alterações na arquitetura do sistema computacional para acomodar os requisitos de gráficos;
- ▶ A primeira alteração relevante, refere-se a um barramento dedicado de dados apenas para a transferência de dados de vídeo entre a memória e o controlador de vídeo (mais provavelmente entre a memória primária e a placa de vídeo);
- ▶ A CPU é muito pouco utilizada pelo processo de refresh, visto que a um chipset dedicado ao processo;

### Arquitetura Simplificada do Barramento



### Setup de Hardware Atual

- ▶ Na memória principal é mantido atualmente apenas o modelo;
- ▶ A renderização final é feita na memória de vídeo que geralmente fica localizada na placa de vídeo;
- ▶ A placa de vídeo é essencialmente um sistema computacional por si só, com um processador (GPU – Graphics Processing Unit) dedicado a rasterizar o modelo;