

```
+-----+
| ASSEMBLY XXVI |
+-----+
```

Vistos os três primeiros modos de escrita da placa VGA, nos resta apenas o modo 2. Esse modo é muito útil para escrita de bitmaps nos modos de vídeo de 16 cores... Ele trabalha basicamente como o registro Set/Reset, sem que tenhamos que manusear esse registro explicitamente.

| O modo de escrita 2

Uma vez setado, o modo de escrita 2 habilita todos os quatro bits de "Enable Set/Reset", da mesma forma que o modo de escrita 3. No entanto, diferente do modo de escrita 3, o registro Set/Reset não precisa ser ajustado com a "cor" desejada. Neste modo o registro Set/Reset é setado com os quatro bits menos significativos enviados pela CPU à memória do sistema. Precisaremos mascarar os bits não desejados em BitMask, bem como ajustar os planos de bits desejados em MapMask.

Repare na força deste modo de vídeo... poderemos atualizar pixels com a "cor" que quisermos sem usarmos Set/Reset diretamente, e sem termos que setar os bits de "Enable Set/Reset". Mas, teremos que ajustar BitMask para não setarmos todos os oito pixels no byte que estamos escrevendo dos planos de bits... Eis um exemplo do modo de escrita 2:

```
+-----+
|
| ideal
| model tiny
| locals
| jumps
|
| include "vga.inc"
|
| LINE_LENGTH      equ      80
|
| codeseg
| org 100h
| start:
|     mov     ax,12h  ; Ajusta modo de vídeo 640x480x16
|     int     10h
|
|     writeMode 2      ; modo de escrita 2
|     MapMask   1111b ; todos os planos de bits
|
|     mov     ax,0A000h
|     mov     es,ax   ; ES = segmento de vídeo
|
|     sub     di,di   ; DI = offset
|     sub     bl,bl   ; usaremos BL p/ contar as linhas.
|
|     mov     ah,10000000b ; ah = bitmask inicial
|     mov     cl,1000b    ; CL = cor inicial
|
| @@1:
|     BitMask ah
|     mov     al,[es:di] ; carrega latches
|     mov     [es:di],cl ; escreve nos planos
|     ror     ah,1      ; rotaciona bitmask
|     inc     cl        ; próxima cor
|     cmp     cl,1000b  ; ops... ultrapassou?!
|
+-----+
```

```

    jb      @@1      ; não... então permanece no loop.
    mov     cl,1000b ; ajusta p/ cor inicial.
    add     di,LINE_LENGTH ; próxima linha
    inc     b1       ; incrementa contador de linhas
    cmp     b1,8     ; chegou na linha 8?
    jb     @@1      ; não... continua no loop.

    sub     ah,ah    ; espera tecla, senão não tem graça!
    int     16h

    mov     ax,3     ; volta ao modo texto...
    int     10h

    int     20h     ; fim do programa.
end start

```

Esse modo parece mais fácil que os demais, não?! Aparentemente é... mas tenha em mente que os outros modos de escrita também têm suas vantagens.

! E os modos de leitura?!

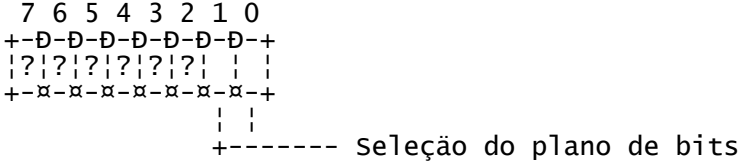
Na grande maioria das vezes não é vantajoso lermos os dados que estão nos planos de bits... Isso porque a memória de vídeo é mais lenta que a memória do sistema (mesmo a memória do sistema associada à placa VGA é mais lenta que o resto da memória do seu PC... por causa dos WAIT STATES que a placa VGA adiciona para não se perder - a velocidade da CPU é maior que a do circuito de vídeo!).

Para encerrarmos os modos de 16 cores é interessante vermos alguma coisa sobre o modo de leitura 0, que é o modo default da placa VGA.

No modo de leitura 0 devemos ler um plano de bits por vez... não é possível ler mais que um plano ao mesmo tempo... e ainda, MapMask não é responsável pela habilitação dos planos de bits. Nesse caso a leitura é feita através de uma ramificação do circuito de vídeo... a escrita é feita por outra. O registrador BitMask também não tem nenhum efeito na leitura. Por isso a seleção dos bits fica por sua conta (através de instruções AND).

A seleção do plano de bits que será lido é feito pelo registrador ReadMap que é descrito abaixo:

» Registrador READMAP



ReadMap também faz parte do circuito GC... Então é acessível via endereços de I/O 3CEh e 3CFh, da mesma forma que BitMask e o registro de MODE, só que seu índice é 4.

Uma nota importante é a de que, embora a leitura seja feita por uma ramificação diferente (por isso a existência de ReadMap), quando fazemos uma leitura dos planos de bits, os latches são automaticamente carregados... e os latches pertencem à ramificação do circuito de escrita (somente os latches dos planos selecionados

por MapMask são carregados, lembra?!).

E zé fini... pelo menos até o próximo texto! :)