

# MDA, CGA, HGC, EGA এবং VGA কার্ডের উপর ভিত্তি করে প্রোগ্রাম করা

(পঞ্চম পর্ব)

এবার সকল কার্ড ব্যবহার করে কিছু প্রোগ্রামের উদাহরণ দেয়া যাবে। যথাক্রমে MDA, CGA, HGC, EGA, VGA এর উপরে ভিত্তি করে মোট পঁচাত্তি প্রোগ্রাম পর পর দেয়া হলো। এগুলো কোন বিশেষ প্রোগ্রাম নয়, মূলতঃ গ্রীসকাল কার্ডের ব্যবহার করা হিসেবে বুঝে সাধারণ প্রোগ্রাম করা হয়েছে। Display Option দিয়ে ভবিষ্যতে আরো আলোচনা করবে। Assembly Language দ্বারা এসকল প্রোগ্রামগুলো করা হলো। তাই এই শ্যাসেমেজে ভুলো ধারণা না থাকলে প্রথমে ভাষাটি শিখে নেবেন, তারপর প্রোগ্রামে গঠন বেলাতে পারবেন।

; Program to display all characters on MDA

```

MODEL small
.STACK 100h
.Code
START
    proc far
    push ds
    mov ax, 0
    push ax

    mov ax, 0b000h
    mov es, ax
    mov cx, 0
    mov ah, 07h
    mov cx, 2000
    cld
    rep stosw

    mov al, 0
    mov ah, 0
    mov cx, 160
    mov es: [di], al
    add di, 4
    mov ah, 80
    cmp jb same_line
    add di, 160
    mov ah, 0
    same_line: cmp al, 255
    je finished
    inc al
    jmp again

finished:
    mov ah, 0
    int 16h
    mov di, 0
    mov al, "
    mov ah, 07h
    mov cx, 2000
    rep stosw
    ret
endp
end
START
; This program demonstrates drawing dots in
CGA
MODEL small
.STACK 100h
.DATA
color db 10101010B
    
```

এ, এস, এম, আশরাফুল হক (রিপন)

```

masks db 1 1000000B
       db 00110000B
       db 00001100B
       db 00000011B
count  dw 24
coordinates dw 103, 154, 103, 155,
            103, 156, 103, 157, 103, 158, 103, 159,
            dw 103, 160, 103, 161,
            103, 162, 103, 163, 103, 164, 103, 165
            dw 103, 166
            dw 102, 155, 102, 165
            dw 101, 156, 10, 164
            dw 100, 157, 100, 163
            dw 99, 158, 99, 162
            dw 98, 159, 98, 161
            dw 97, 160
            db 80

eighty .CODE
main   proc far
start:  proc push ds
        sub ax, ax
        push ax

        mov ax, @data
        mov ds, ax
        mov ah, 0
        mov al, 4
        int 10h
        mov dx, 3d9h
        mov al, 0fh
        mov out dx, al
        mov ax, cb800h
        mov es, ax
        mov cx, count
        mov bx, offset coordinates
        mov ax, [bx]
        loop:
            inc bx
            inc bx
            mov dx, [bx]
            inc bx
            inc bx
            call draw
            loop loop

main   endp
draw  proc near
        al, 1
        odd:
            mov di, 0
            jmp short common
        odd:
            mov di, 2000h
        common:
            mul eighty
            add di, ax
            mov si, dx
            shr dx, 1
            shr dx, 1
            add di, dx
            and si, 03h
            mov al, masks [si]
            mov ch, color
            mov dh, al
            not al
            mov ah, es: [di]
    
```

```

and ah, al
or ah, dh
mov es: [di]
ret
draw   endp
end start
; This program set the card as a CGA card (APA
mode)
.MODEL small
.STACK 100h
.DATA
cga_argu db 38h, 28h, 2dh, 0ah, 7h, 06h,
          64h, 70h, 2, 1, 6, 7, 0, 0, 0, 0
.CODE
main   proc far
        proc push ds
        mov ax, 0
        push ax
        mov ax, 40h
        mov ds, ax
        mov ax, ds: [10h]
        and ax, 00 ch
        or ax, 20h
        ds: [10h] ax
        mov al, 4
        mov ah, 0
        int 10h
        mov ax, @DATA
        mov ds, ax
        mov al, 3
        mov dx, 3bth
        out dx, al
        call ega_init
        ret
main   endp
cga_init PROC
        push es
        push ds
        mov al, 2
        mov dx, 3b8h
        out dx, al
        mov ax, ds
        mov ex, ax
        lea si, cga_argu
        mov dx, 3b4h
        mov cx, 16
        xor ah, ah
        mov al, ah
        out dx, al
        inc dx
        lodsb
        out dx, al
        inc ah
        dec dx
        loop loop
        cld
        mov cx, 2000h
        mov ax, 0b800h
        mov es, ax
        xor di, di
        mov ax, 0
        rep stosw
    
```

