

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

(পেছ পাতা)

এবাব সকল কার্ড ব্যবহার করে বিষয় প্রোগ্রামের উদাহরণ দেয়া যাব। যথাক্রমে MDA, CGA, HGC, EGA, VGA এবং উপরে তিনি করে মেট পার্টিশনে প্রোগ্রাম গুর পর দেয়া হলো। এতে পুর বিশেষ প্রোগ্রাম নাম, মূলত প্রস্তুত কার্ডের ব্যবহার করা হিসেবে খুব সাধারণ প্রোগ্রাম করা হয়েছে। Display Option মিনে অভিযানে আরো আলাদান প্রোগ্রাম করলো। Assembly Language দ্বাৰা এবন্স প্রোগ্রামগুলো করা হলো। তাই এই প্রযোজনে কালো ধৰণী ধৰণী না ব্যবহার করে আবাসন প্রোগ্রাম কোথাৱে পাবেন।

: 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 di, 0

mov al, "

mov ah, 07h

mov cx, 2000

cld

rep stosw

mov al, 0

mov ah, 0

mov di, 160

again : mov es:[di], al

add di, 4

add ah, 2

cmp ah, 80

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

START endp

end START

: This program demonstrates drawing dots in

CGA

.MODEL small

.STACK 100h

.DATA

color db 10101010B

	masks	db1	1000000B	and ah, al
		db	00110000B	or ah, dh
		db	00001100B	mov es:[di]
		db	00000011B	ret
count	dw	24		draw endp
coordinates	dw	103, 154, 103, 155, 103, 156, 103, 157, 103, 158, 103, 159, 103, 160, 103, 161, 103, 162, 103, 163, 103, 164, 103, 165		end start
		dw	103, 166	: This program set the card asa CGA card (APA mode)
		dw	102, 155, 102, 165	.MODEL small
		dw	101, 156, 10, 164	.STACK 100h
		dw	100, 157, 100, 163	.DATA
		dw	99, 158, 99, 162	cga_argu db 38h, 28h, 2dh, 0ah, 7fh, 06h,
		dw	98, 159, 98, 161	64h, 70h, 2, 1, 6, 7, 0, 0, 0, 0
		dw	97, 160	.CODE
eighty	db	80		main proc far
		main		push ds
		start :		mov ax, 0
		proc	far	push ax,
		push	ds	mov ax, 40h
		sub	ax, ax	mov ds: [10h] ax
		push	ax	and ax, 00 cffh
		mov	ax, @data	or ax, 20h
		mov	ds, ax	mov dx: [10h] ax
		mov	ah, 0	mov al, 4
		mov	al, 4	mov ah, 0
		int	10h	int 10h
		mov	dx, 3d9h	mov ax, @DATA
		mov	al, 0fh	mov ds, ax
		out	dx, al	mov al, 3
		mov	ax, 0b80h	mov dx, 3b8h
		mov	es, ax	out dx, al
		mov	cx, count	call ega_init
		mov	bx, offset coordinates	ret
dloop :	mov	ax, [bx]		main ENDP
		inc	bx	cga_init PROC
		inc	bx	push es
		mov	dx, [bx]	push ds
		inc	bx	mov al, 2
		inc	bx	mov dx, 3b8h
		inc	bx	out dx, al
		call	draw	mov ax, ds
		loop	dloop	mov ex, ax
		ret		lea si, cga_argu
		endp		mov dx, 3b4h
		draw	near	mov cx, 16
		proc		xor ah, ah
		shr	si, 1	mov al, ah
		jc	odd	out dx, al
		mov	di, 0	inc dx
		jmp	short, common	lodsb
		mov	di, 2000h	out dx, al
odd:				dec ah
		common :		inc dx
		mul	eighty	- init_6845
		add	di, ax	init_6845
		mov	si, dx	
		shr	dx, 1	
		shr	dx, 1	
		add	di, dx	
		and	si, 03h	
		mov	al, masks [si]	
		mov	dh, color	
		and	dh, al	
		not	al	
		mov	ah, es:[di]	
				rep

```

mov al, 8ah
mov dx, 3b8h
out dx, al
pop ex
pop ds
ret
cga_init ENDP
end main

; An example of setting EGA palette in
; alphanumeric mode
.MODEL small
.STACK 100h
.DATA
cry_I equ 1
crs db 16, 15, 14, 13, 12, 11,
10, 9, 8, 7, 6, 5, 4, 2, 63, 1, 0
overscan db 0
flags db 0
.CODE
main PROC far
push ds
xor ax, ax
push ax
mov ax, @DATA
mov ds, ax
mov es, ax
mov ah, 2
mov ch, 05h
mov d1, 0ah
mov bh, 0
int 10h
mov cx, 16
mov bx, cx
dec bx
mov al, crs [bx]
call show_new
mov ah, 0
int 16h
loop d_lp
mov cx, 10h
push cx
mov cx, 16
or flags, crg_f
lp1:
mov bx, cx
dec bx
mov al, crs [bx]
test flags, not cry_f
jz no_c
inc ax
and flags, cry_f
cmp al, 64
jb no_cry
sub al, al
or flags, cry_f
no_cry:
call show_new
mov crs [bx], al
no_c:
loop lp1
mov ah, 0
int 16h
loop lp2
ret
main ENDP
show_new PROC near
push ax
push bx
push cx
mov bh, al
mov ah, 10h
mov al, 2

```

```

mov bx, offset.cls
int 10h
push bx
mov ah, 2
mov bh, 0
mov dh, 0
mov dl, bl
shl dx, 1
int 10h
pop bx
mov al, bh
call bin 2asc
push ax
xchg ah, al
mov ah, 9
mov bl, bh
mov bh, 0
push bx
mov cx, 1
int 10h
mov ah, 2
inc dx
int 10h
pop bx
pop ax
mov ah, 9
int 10h
pop cx
pop bx
pop ax
ret
show_new ENDP
bin 2asc PROC
aam
add ax, 3030h
rel
bin 2asc ENDP
end main

; Get the VGA or EGA configuration information
.MODEL small
.STACK 100h
.DATA
no_support db 'EGA or VGA is not active' 0dh, 0ah
crg_ega db 'You have an active color EGA', 0dh, 0ah
crg_vga db 'You have an active VGA', 0dh, 0ah
mono_ega db 'You have an active Mono EGA', 0dh, 0ah
mono_vga db 'You have an active Mono VGA', 0dh, 0ah
.CODE
main PROC far
push ds
xor ax, ax
push ax
mov ax, @ DATA
mov ds, ax
mov ax, 1a00h
int 10h
cmp al, 1ah
cmp bl, 7
je mono_v
cmp bf, 8
je color_v
cmp al, 4
je color_e
cmp al, 5

```

```

je mono_e
mov ah, 12h
b1, 10h
int 10h
cmp b1, 10h
je invalid
push ds
mov ax, 40h
mov bx, ax
mov bx, ds: [87h]
pop ds
test bl, 8
jz valid
invalid:
mov bx, offset.no_support
jmp finish
valid:
mov cmp bh, 1
je mono_e
jmp color_e
mono_v:
mov bx, offset.mono_vga
jmp finish
color_v:
mov bx, offset.clr_vga
jmp finish
color_e:
mov bx, offset.clr_ega
jmp finish
mono_e:
mov bx, offset.mono_ega
jmp finish
finish:
mov dl, [bx]
mov ah, 2
int 21h
inc bx
cmp dt, 10
jne rel
rel
main ENDP
end main

টার্নিভিত পার্শ্বিক প্রোগ্রামে কোনো ফুল-স্মিথ ধারকে
পার্টিশন নিরের ঠিকানায় সরাসরি যোগাযোগ করতে
পারেন।
A.S.M. ASHRAFUL HAQ [RIPON]
NORTHERN JIANGTONG UNIVERSITY
LIU XUE SHENG LOU
ROOM NO - 307
BEIJING - 10004, CHINA.
```

বিশেষ সুযোগ !

মাসিক কম্পিউটার জগৎ-এর প্রাথমিক হওয়ার জন্য বিশেষ সুযোগ দেওয়া হচ্ছে। এখন থেকে একজন দুই বছরের জন্য অথবা দুইজন একত্রে (বিভিন্ন ঠিকানায়) এক বছরের জন্য প্রাথক হতে হলে মাত্র ৩০০/- (তিনশত) টাকা সংগ্রহ/প্রেতজর্জির/মালি অর্ডারের মাধ্যমে পার্টিলেই চলবে। ঢাকা শহরের প্রাথক বাতীত চেক অঙ্গনযোগ্য নয়। এছাড়া ৬ মাসের জন্য প্রাথক ফী ১১০/- টাকা এবং এক বছরের জন্য ২০০/- (মুইশত) টাকা মাত্র। প্রাথক টাঙ্গা পার্টিতে হবে 'কম্পিউটার জগৎ'-এই নামে।

ঠিকানা ১৪৬/১ আজিবপুর রোড, ঢাকা-১২০৫।