

UDK 681.325

Matjaž Debevč
Metka Zorič
Rajko Svečko
Dali Đonlagič
Tehniška fakulteta Maribor

POVZETEK

Na področju računalništva prihaja vedno bolj do izraza področje računalniške grafike. Osebnih računalnikov postajajo tudi vedno bolj pristopni širokim množicam uporabnikov. Trenutno najbolj aktualna možnost grafičnega prikazovanja nam podaja EGA grafična kartica za IBM in kompatibilne osebne računalnike. Delo opisuje zgradbo, osnovo in programiranje te kartice.

ABSTRACT

Computer graphics is becoming more and more distinctive in the field of computers. Personal computers are becoming accessible to a wide number of users. At the moment the EGA graphics card presents the best possibility of graphic display for IBM and a compatible personal computers. The present thesis describes the structure and programming of this card.

1. UVOD

Računalniška grafika je ena najbolj spektakularnih možnosti, ki nam jih dajejo računalniki. Računalniška grafika je poseben medij, ki omogoča najlažjo in najhitrejšo komunikacijo med človekom in strojem. Človeško oko lahko dosti hitreje razbere informacijo z grafične slike, kakor pa s tabele, polne številke.

Dolga leta je bilo to področje izredno drago in težko dostopno. Zadnjih 30 let je čutili enakomeren padec svetovnih cen na področju računalništva. Ta padec se giblje za okoli 15% letno. To je vzrok, da postajajo računalniki z velikimi zmogljivostmi vedno bolj dostopni. Povečanje zmogljivosti in padec cen se med drugim tudi čuti na področju osebnih računalnikov IBM in njemu kompatibilnih. Nekdaj niti pomisliti ni bilo mogoče na visoko zmogljivo grafiko. Slika, ki je bila nekoč narisana v nekaj minutah z majhno resolucijo zaslona, se danes nariše v trenutku na visoko resolucijskem ekranu. Ni daleč čas, ko bo trodimenzionalna grafika čisto nekaj vsakdanjega.

2. IBM - Enhanced Graphics Adapter

Na osebnih računalnikih IBM in njemu kompatibilnih je danes najbolj aktualna grafična kartica EGA, ki je našla pot v marsikateri osebni računalnik in je največkrat obravnavana kot standard za ostale grafične kartice. EGA kartico imenujejo tudi IBM-HR (High Resolution) kartica. To je kartica, ki postavlja nove možnosti na področju grafike na osebnih računalnikih.

Že več kot dve leti je ta kartica na trgu in že obstaja cela vrsta tako imenovanih kompatibilnih EGA kartic. V grafičnem načinu je EGA kartica z resolucijo 640 * 350 točk za 150 grafičnih vrstic boljše od stare barvne grafične kartice CGA (Colour Graphic Adapter).

Takšna resolucija nam omogoča ukinitvev utripajoče slike, ne glede na to ali uporabljamo monohromatski zaslon ali novi IBM "Enhanced Color Display". Seveda omogoča EGA kartica tudi emuliranje starega zaslona s resolucijo 320 * 200 ali 640 * 200 (glej Načini EGA kartice).

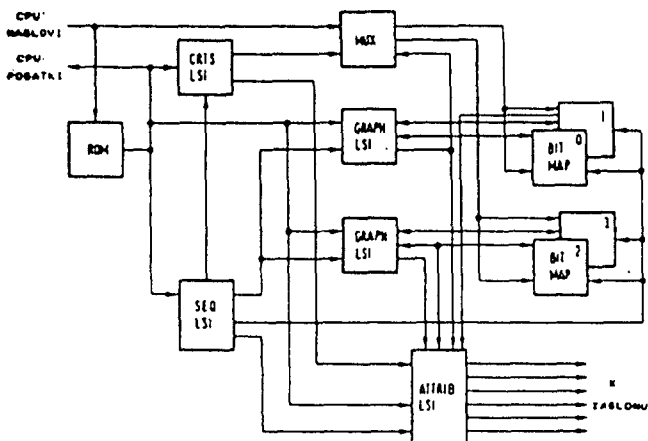
Za individualno uporabo EGA kartice sedaj ne obstaja veliko uporabniških programskih paketov. Družba Borland International

predstavlja s svojim TURBO PASCAL-om verzija 4.0 zelo dober in tudi poceni programski paket, ki med drugim podpira tudi EGA kartico. Vendar običajno tako velikega programskega paketa niti ne potrebujemo. Poleg tega mora biti program napisan v pascalu.

Za pisanje samo nujno potrebnih, kratkih funkcij za uporabo EGA grafične kartice je potrebno poznati zgradbo in delovanje grafične kartice.

2.1. ZGRADBA EGA KARTICE

Programiranje za EGA kartico je težko že zaradi množice registrov (70), kakor tudi zaradi množice funkcij, ki jih ta kartica omogoča. Že ko pogledamo v "Technical Reference Manual" in blokovni diagram za EGA kartico (slika 1), nam postane jasno, da je potrebno natančno razumevanje funkcij in registrov, če želimo, da bomo imeli pravilno delujoč programski paket. Kot osnova nam služi knjiga "IBM - Personal Computer, IBM - Enhanced Graphics Adapter".



slika 1 : Blokovni diagram EGA kartice

Glede na blokovni diagram delimo EGA kartico na:

2.1.1. CRT Controller

CRTC (Cathode Ray Tube Controller) je po funkciji razširjeni 6845. Generira sinhronizirane signale za horizontalne in vertikalne premike. Odgovarjajoči parametri se programirajo v 19 CRT registrih. Vertikalni registri so dolgi 9 bitov, kar nam da resolucijo 512 slikovnih točk. Če je drugi bit v "Mode Control Register" enak 1, se lahko resolucija podvoji na 1024 vrstic na sliko, kar pa pomeni, da potrebujemo poseben monitor. Nadalje skrbi CRTC za obnovev slike shranjene v dinamičnem RAM-u. Naslov za obnovev slike je dolg dva byta, kar pomeni, da CRTC dovoljuje naslavljanje do 64 Kbytov.

CRT kontroler dosežemo z vhodno/izhodnim naslovom 3?4 in 3?5, kjer

"?" pomeni B ali D.B uporabimo za Monochrome način, D pa za barvni način. 3?4 je naslovni register, medtem ko je 3?5 bralno pisalni podatkovni register. Vsebina registrov:

1. Address Register
2. Horizontal Total
3. Horizontal Display End
4. Start Horizontal Blank
5. End Horizontal Blank
6. Start Horizontal Retrace
7. End Horizontal Retrace
8. Vertical Total
9. Overflow
10. Preset Row Scan
11. Max Scan Line
12. Cursor Start
13. Cursor End
14. Start Address High
15. Start Address Low
16. Cursor Location High
17. Cursor Location Low
18. Vertical Retrace Start
19. Light Pen High
20. Vertical Retrace End
21. Light Pen Low
22. Vertical Display End
23. Offset
24. Underline Location
25. Start Vertical Blank
26. End Vertical Blank
27. Mode Control
28. Line Compare

2.1.2. Zaslonski pomnilnik

Zaslonski pomnilnik se sestoji iz štirih pomnilniških prostorov (Bit Planes). V osnovni verziji je vsak prostor velik 16 Kbytov in se nahaja na EGA kartici. V dveh korakih se lahko kapaciteta dvakrat poveča. Potem govorimo o 64, 128 ali 256 Kbytni EGA kartici. Slika z resolucijo 640 * 350 ima 80 * 350 bytov. Organizacija slikovne točke v zaslonskem pomnilniku je v grafičnem načinu naslednja:

Prvih osem slikovnih točk zaslona (zgornji levi kot !) se nahaja na naslovu 000H slikovnega pomnilnika, naslednjih osem na naslovu 001H itd. Prva slikovna točka znotraj enega byta je bit 7, naslednja je bit 6.

2.1.3. Grafični in tekstovni način

Štirje pomnilniški prostori zaslonskega pomnilnika reprezentirajo v grafičnem načinu tudi naslov barvnih atributov ene slikovne točke (16 barv). Ta naslov je označen v Attribute-Controller-ju kot indeks v barvnem registru. Več o tem je napisano v

poglavju o Attribute-Controller-ju. Če ima EGA kartica 64 Kbytov na voljo, lahko uporabimo samo 4 barve.

V tekstovnem načinu je pomnilniški prostor znakovno kodni pomnilnik, pomnilniški prostor 1 je pomnilnik za attribute, pomnilniški prostor 2 pa je znakovni generator.

Če se selektira tekstovni način, prenese BIOS enega od dveh znakovnih stavkov v pomnilniški prostor 2.

2.1.4. Sekvenčni generator in multiplexer

Naslovni multiplexer (MUX) krmili naslove zaslonskega pomnilnika, ki pridejo enkrat direktno od CPU-ja (naložitev zaslonskega pomnilnika) in drugič od CRTC (zaslonska osvežitev). Sekvenčni generator (SEQ) generira krmilne signale za zaslonski pomnilnik in takt za zaslonsko ponavljanje. Map-Mask-Register omogoča individualno vpisovanje na zaslonski pomnilnik. Zraven lahko tudi izberemo določeno barvo. Za tekstovni način (A/N) se nahaja v Character-Map-Select registru izbira štirih znakovnih stavkov.

Sekvenčni register dosežemo z naslovi 3C4 (select naslova) in 3C5 (podatkovni vhodno - izhodni register).

Vsebina registra :

1. Naslov
2. Reset
3. Clocking Mode
4. Map Mask
5. Character Map Select
6. Memory Mode

2.1.5. Grafični kontroler

Grafični kontroler lahko dela v dveh načinih; v grafičnem ali v tekstovnem. V grafičnem načinu se prenesejo podatki serijsko preko tako imenovanih Bit-Plane vodil C0, C1, C2, C3 k Attribute - Controller-ju. Ta štiri vodila, vsako za zaslonski pomnilniški prostor (Bit Plane Map) tvorijo 16 možnih barv (glej Attribute Controller).

Prvi grafični kontroler je za zaslonski pomnilniški prostor 0 in 1 (2 byta), drugi grafični kontroler pa za ostala dva (2 in 3). Oba grafična kontrolerja imata vsak po 16 bitni podatkovni register, kjer se lahko vpiše ali bere 32 bitov slikovnih podatkov v enem pomnilniškem ciklusu. Nadalje se nahaja v obeh grafičnih kontrolerjih še krmiljenje za izbiro slikovne točke in njegove barve, kakor tudi štiri logične funkcije (zamenjava, AND, OR, XOR).

Color-Compare register omogoča

hkratno primerjanje z eno določeno barvo preko osmih slikovnih točk enega byta. V tekstovnem načinu se podatki transportirajo direktno v Attribute-Controller.

Register dosežemo z naslovi 3CE (select naslova) in 3CF (podatkovni vhodno izhodni naslov).

Vsebina registra:

1. Graphics 1 Position
2. Graphics 2 Position
3. Graphics 1 & 2 naslov
4. Set/Reset
5. Enable Set/Reset
6. Color Compare
7. Data Rotate
8. Read Map Select
9. Mode Register
10. Miscellaneous
11. Color Don't Care
12. Bit Mask

2.1.6. Attribute Controller

Ta kontroler je zadnja postaja slikovnih podatkov na poti k zaslonu. Ima 21 registrov. Prvih 16 naslovov so tako imenovani barvno paletni registri. Biti 0 do bita 5 teh 16 registrov omogočajo dinamično izbiro 64 možnih barv.

Barvno paletni register se naslavlja preko štirih vodil C0 - C3, ki pridejo od grafičnega kontrolerja. To se dogaja serijsko, torej točka za točko. Preden enega od teh 16 registrov naslavljam, gredo podatki tudi preko Color-Plane-Enable registra. Ponavadi se naslov pusti tukaj na miru, da imamo lahko na voljo 16 barv. Z Color-Plane-Enable registrom pa lahko omogočamo prikaz enega od štirih slik, ali pa kombinacijo teh štirih slik.

V tekstovnem načinu se tekstovni podatki prenašajo paralelno iz slikovnega pomnilnika v Attribute Controller.

Osvežilni naslov generiran od CRTC se naslavlja najprej na slikovni pomnilniški prostor 0 in 1.

Byte naslova, shranjen v pomnilniškem prostoru 1 se prenese v Attribute Controller, kjer ima za vsakokratno prikazano črko svojo veljavnost. Prebrana znakovna koda iz pomnilniškega prostora 0, na primer 41H za črko "A", se naslavlja v povezavi z Row-Scan-Count registrom zaslonskega pomnilniškega prostora 2, kjer se nahaja ta v ROM-u naložen znak.

Prebrani biti se pošljejo skozi grafični kontroler k Attribute Controller-ju, skupaj z barvnimi vrednostmi in so nato vodeni serijsko dalje k zaslonu. Row-Scan-Count register nastavi vertikalno višino znaka na zaslonu, to pomeni število horizontalnih vrstic iz katerih je znak sestavljen. Znak je lahko velik maksimalno

32 znakov in se nastavi v CRTC registru 9.

Vsebina:

1. Address Register
2. Palette Register
3. Mode Control Register
4. Overscan Color Register
5. Color Plane Enable Register
6. Horizontal Pel Panning Register

3. PROGRAMIRANJE EGA KARTICE

Prejšnje poglavje je bilo namenjeno spoznavanju zgradbe in strukture EGA kartice. To poglavje pa je namenjeno programiranju EGA kartice.

Namesto da bi pisali velik programski grafični paket, včasih potrebujemo samo osnovne ukaze za risanje na ekran, kakor so na primer vklop, izklop grafike, risanje črte in brisanje zaslona. V resnici obstaja veliko programskih paketov, ki podpira grafiko na EGA kartici za IBM-PC in kompatibilne, vendar programerji, ki želijo imeti hitre in učinkovite programe raje sežejo po osnovnih funkcijah, ki jih lahko sami napišejo.

Za programiranje EGA kartice potrebujemo EGA-BIOS funkcije. V ROM-u kartice se nahajajo poleg teh funkcij še oba znakovna stavka in ena samostojna funkcija.

EGA-BIOS vsebuje interrupt (prekinitvene) vektorje 05H in 10H. (H pomeni heksadecimalno). Interrupt 05H uporabimo za izpis na zaslon, medtem ko uporabljamo interrupt 10H za 19 krmilnih funkcij EGA kartice.

Ker se na PC-ju največkrat uporabljata prevajalnika TURBO-PASCAL in TURBO-C od firme BORLAND, so tudi podprogrami napisani v teh dveh jezikih. Prenos na prevajalnike ostalih firm ni problematičen, samo potrebno je vedeti, kako kličemo BIOS funkcije. Programa vključujeta ukaze za vklop, izklop grafike, risanje točke (pixla) in brisanje ekrana.

3.1. Nastavitev načina EGA kartice

Kadar želimo risati z EGA kartico, moramo najprej vklopiti EGA v ustrezni način.

Delovni načini EGA kartice:

EGA način	resolucija	število barv
C 0	320 X 200	16
E 0	320 X 350	16/64
C 1	320 X 200	16
E 1	320 X 350	16/64
C 2	640 X 200	16
E 2	640 X 350	16/64
C 3	640 X 200	16
E 3	640 X 350	16/64
C 4	320 X 200	4
C 5	320 X 200	4
C 6	640 X 200	2
M 7	720 X 350	4
C D	320 X 200	16

M E	640 X 200	16
M F	640 X 350	4
M 10	640 X 350	4/16

C - IBM Color zaslon, E - IBM Enhanced Color
M - IBM Monochrom zaslon

3.1.1. Za vklop in izklop grafičnega zaslona uporabljamo naslednje BIOS funkcije:

AH = 0; izbira BIOS načina
AL: EGA način po zgornji tabeli

AH = 0FH; prikaz trenut. zaslonskega statusa
parametri, ki vračajo podatke so:
AL: EGA - način
AH: število znakov/vrstico
BH: število trenutnih zaslonskih strani

Podprogrami:

Ce uporabljamo verzijo 4.0 TURBO-PASCALA, potem v glavi programa vstavimo :

```
uses DOS;
```

tip pa je definiran z Registers.

Kadar pa uporabljamo verzijo 3.0 pa je potrebno dodati :

```
type Registers =
  record
    case integer of
      1: (ax,bx,cx,dx,bp,si,di,ds,
         es,flgs: integer);
      2: (al,ah,bl,bh,cl,c,dl,dh: byte);
    end;

procedure Grafika(vklop: boolean);
var
  reg: Registers;
  nacin: integer;

begin {Grafika}
  if vklop then
    begin
      { branje aktualne nastavitve zaslona }
      reg.ax:= $0F00;
      Intr($10,reg);
      { shranitev načina }
      nacin:= reg.al;

      { EGA način za Enhanced Color Display }
      reg.ax:= $0010;
      Intr($10,reg);

    end
  else
    begin
      reg.ah:= 0; { izklop grafičnega načina }
      reg.ax:= nacin;
      Intr($10,reg);
    end;
end; {Grafika}
```

Jezik C:

```
# include <dos.h>

union REGS reg;
int nacin;

Grafika(vklop)
int vklop;

{
  if (vklop)
  {
    reg.x.ax = 0x0F00;
    int86(0x10,&reg,&reg);

    reg.x.ax = 0x0010;
    int86(0x10,&reg,&reg);
  }
  else
  {
    reg.h.ah = 0;
    reg.x.ax = nacin;
    int86(0x10,&reg,&reg);
  }
}
```

3.1.2. Za risanje točke uporabljamo naslednje BIOS funkcije:

AH = 0CH; nariši grafično točko
 BH: tekoča stran
 DX: Y pozicija (0 - 349)
 CX: X pozicija (0 - 639)
 AL: barva (0 - 63)

AH = 0DH; beri grafično točko
 BH: tekoča stran
 DX: Y pozicija (0 - 349)
 CX: X pozicija (0 - 639)

Vrtnjen parameter:
 AL: barva željene vrednosti

Podprogram za risanje točke v pascalu:

```
procedure Tocka(x,y,barva);
begin {Tocka}
  reg.ah:= $0C; { riši grafično točko }
  reg.cx:= x;
  reg.dx:= y;
  reg.al:= barva;
  Intr($10,reg);
end; {Tocka}
```

Program v C jeziku:

```
Tocka(x,y,barva)
int x,y,barva
{
  reg.x.ah = 0x0C;
  reg.x.cx = x;
  reg.x.dx = y;
  reg.h.al = barva;
  int86(0x10, &reg, &reg);
}
```

3.1.3. Za brisanje ekrana uporabimo naslednjo BIOS funkcijo:

AH = 07H; pomik trenutne strani navzdol
 AL: vstavitev okna
 CH: spodnji levi rob (vrstica)
 CL: spodnji levi rob (kolona)
 DH: zgornji desni rob (vrstica)
 DL: zgornji desni rob (kolona)
 BH: barva ozadja;

Podprogram za brisanje ekrana v pascalu:

```
procedure brisiZaslou;
begin {brisiZaslou}
  reg.ah:= $07; { pomik strani navzdol }
  reg.al:= 0; { vstavitev okna }
  reg.ch:= 0; { spodnji levi rob }
  reg.cl:= 0;
  reg.dh:= 24; { zgornji desni rob }
  reg.dl:= 79;
  reg.bh:= 0; { barva ozadja }
  Intr($10,reg);
end; {brisiZaslou}
```

V C jeziku :

```
brisiZaslou()
{
  reg.h.ah = 0x07;
  reg.h.al = 0;
  reg.h.ch = 0;
  reg.h.cl = 0;
  reg.h.dh = 24;
  reg.h.dl = 79;
  reg.h.bh = 0;
  int86(0x10, &reg, &reg);
}
```

LITERATURA

Johnson Nelson:
 ADVANCED GRAPHICS IN C
 Programming and Techniques,

EGA Users' Manual

IBM Personal Computer:
 IBM Enhanced Graphics Adapter,

IBM Personal Computer:
 TURBO PASCAL V4.0.