

GO240128 a GO128x64

JEDNOTKY IMPLEMENTUJÍCÍ OBJEKTY
PRO PŘÍMÉ KRESLENÍ DO
VIDEOPAMĚTI

Příručka uživatele a programátora



SofCon[®] spol. s r.o.
Střešovická 49
162 00 Praha 6
tel/fax: +420 220 180 454
E-mail: sofcon@sofcon.cz
www: <http://www.sofcon.cz>

Informace v tomto dokumentu byly pečlivě zkontrolovány a SofCon věří, že jsou spolehlivé, přesto SofCon nenese odpovědnost za případné nepřesnosti nebo nesprávnosti zde uvedených informací.

SofCon negarantuje bezchybnost tohoto dokumentu ani programového vybavení, které je v tomto dokumentu popsáno. Uživatel přebírá informace z tohoto dokumentu a odpovídající programové vybavení ve stavu, jak byly vytvořeny a sám je povinen provést validaci bezchybnosti produktu, který s použitím zde popsaného programového vybavení vytvořil.

SofCon si vyhrazuje právo změny obsahu tohoto dokumentu bez předchozího oznámení a nenese žádnou odpovědnost za důsledky, které z toho mohou vyplynout pro uživatele.

Datum vydání: 16.05.2003

Datum posledního uložení dokumentu: 16.05.2003

(Datum vydání a posledního uložení dokumentu musí být stejné)

Upozornění:

V dokumentu použité názvy výrobků, firem apod. mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

Obsah :

1.O dokumentu	5
1.1. Revize dokumentu	5
1.2. Účel dokumentu	5
1.3. Rozsah platnosti	5
1.4. Související dokumenty	5
2.Termíny a definice	5
3.Úvod	6
4.Popis konstant a typů	6
5.Popis objektu tGraphicArea	6
5.1. Proměnné	6
5.2. Metody	6
5.2.1. Init	6
5.2.2. ClearImage	6
5.2.3. LoadBkBmp	7
6.Popis objektu tGrRecorder	7
6.1. Proměnné	7
6.2. Metody	7
6.2.1. Init	7
6.2.2. DefRecorderRect	7
6.2.3. ClearRecorderRect	7
6.2.4. ShiftRecorderRectLeft	8
6.2.5. WrDifXToRecorder	8
6.2.6. WrToRecorder	8

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	Če		První vydání
1.10	2.XX	Tu	16.05.2003	Úprava dokumentu dle ISO9000 Opravená jména proměnných a funkcí

1.2. Účel dokumentu

Tento dokument slouží jako popis knihoven **GO240128** a **GO128x64**, jednotek implementujících objekty pro přímé kreslení do videopaměti.

1.3. Rozsah platnosti

Určen pro programátory a uživatele programového vybavení SofCon.

1.4. Související dokumenty

Pro čtení tohoto dokumentu je potřeba seznámit se s manuálem uBitMap a uTermGr.

Popis formátu verze knihovny a souvisejících funkcí je popsán v manuálu LibVer.

2. Termíny a definice

Používané termíny a definice jsou popsány v samostatném dokumentu Termíny a definice.

3. Úvod

Jednotka **GO240128** a **GO128x64** implementují dva objekty pro přímé kreslení do videopaměti. Implementace obou jednotek je shodná, liší se pouze v rozměrech videopaměti. Jednotka **GO240128** je určena pro videopaměť o rozměru 240 × 128 pixelů (rozměr displeje terminálu TERM10) a jednotka **GO128x64** pro videopaměť o rozměrech 128 × 64 pixelů (rozměr displeje terminálu TERM03).

Jednotky nahrazují starší jednotku **uGrObj**.

4. Popis konstant a typů

```
cVerNo = např. $0251; { BCD formát }  
cVer   = např. '02.51,07.08.2003';
```

Číslo verze jednotky v BCD tvaru a v textové podobě včetně datumu změny.

5. Popis objektu tGraphicArea

```
pGraphicArea=^tGraphicArea;  
tGraphicArea=object
```

Objektový typ **tGraphicArea** implementuje bitmapu umístěnou v paměti RWM. Je-li tato bitmapa instalována do pole uživatelských bitmap, může být použita jako pozadí některé menu-obrazovky. Její umístění do paměti RWM dává uživateli možnost měnit její obsah za běhu programu.

5.1. Proměnné

```
Hed:tBmpHeader;
```

Proměnná **Hed** obsahuje hlavičku bitmapy. V hlavičce je uložena její vlastní délka, způsob zakódování bitmapy a rozměry bitmapy.

```
Img:tGraphicArea;
```

Proměnná **Img** představuje videopaměť pro uložení bitmapy.

5.2. Metody

5.2.1. Init

```
constructor Init(W,H:integer);
```

Konstruktor **Init** inicializuje objekt a nastaví hlavičku bitmapy. V parametrech **W**, **H** je předána šířka a výška bitmapy v pixelech.

5.2.2. ClearImage

```
procedure ClearImage;
```

Metoda **ClearImage** smaže bitmapu.

5.2.3. LoadBkBmp

```
procedure LoadBkBmp (BmpPtr: Pointer; var Lng: Word);
```

Metoda **LoadBkBmp** naplní bitmapu obsahem jiné bitmapy. V parametru **BmpPtr** je předán ukazatel na zdrojovou bitmapu, jejíž obsah má být přesunut do bitmapy. V parametru **Lng** se předává délka zaváděné bitmapy.

6. Popis objektu tGrRecorder

Objektový typ **tGrRecorder** slouží ke kreslení pohyblivých grafů ve stylu souřadnicového zapisovače. Objekt vlastní odkaz na bitmapu (instanci objektu **tGraphicArea**), do které je zapisovač kreslen.

6.1. Proměnné

```
RefGrArea : pGraphicArea;
```

Proměnná **RefGrArea** uchovává odkaz na bitmapu, do které je zapisovač kreslen.

```
Mx1, My1, Mx2, My2: integer;
```

Proměnné **Mx1**, **My1**, **Mx2**, **My2** uchovávají souřadnice okénka v bitmapě, do kterého je zapisovač kreslen.

```
MBx1, MBx2, LastX, LastY: integer;
```

Proměnné **MBx1**, **MBy1**, **LastX**, **LastY** jsou pomocné proměnné určené pro vnitřní potřebu objektu.

6.2. Metody

6.2.1. Init

```
constructor Init (var GrArea: tGraphicArea);
```

Konstruktor **Init** inicializuje objekt. Jako parametr je mu předávána instance objektu **tGraphicArea** jako bitmapa, do které bude zapisovač kreslen. Okénko pro kreslení zapisovače je implicitně nastaveno na celý rozměr bitmapy.

6.2.2. DefRecorderRect

```
procedure DefRecorderRect (X1, Y1, X2, Y2);
```

Metoda **DefRecorderRect** slouží k nastavení okénka v bitmapě, do kterého bude zapisovač kreslen. Z důvodu implementace je nutné aby parametry **X1** a **X2** měli hodnotu dělitelnou osmi.

6.2.3. ClearRecorderRect

```
procedure ClearRecorderRect;
```

Metoda **ClearRecorderRect** smaže okénko pro kreslení zapisovače.

6.2.4. ShiftRecorderRectLeft

```
procedure ShiftRecorderRectLeft (ShiftX: integer);
```

Metoda **ShiftRecorderLeft** posune obsah okénka o zadaný počet pixelů vlevo. Pixely, které se dostanou na levé straně mimo okénko jsou ztraceny a na pravé straně jsou doplněn příslušný počet prázdných pixelů. K posunu metoda využívá rychlou a efektivní rutinu v implementovanou assemblerem. Metoda **ShiftRecorderLeft** je využívána metodami **WrDifXToRecorder** a **WrToRecorder**.

6.2.5. WrDifXToRecorder

```
procedure WrDifXToRecorder (DifX, Y: integer);
```

Metoda **WrDifXToRecorder** je určena k vykreslení kroku souřadnicového zapisovače. Parametr **DifX** udává posunutí souřadnicového zapisovače a parametr **Y** vertikální souřadnici pisátka. Metoda posune voláním **ShifRecorderLeft** obsah okénka o **DifX** pixelů vlevo a vykreslí úsečku od polohy pisátka při posledním volání **WrDifXToRecorder** nebo **WrToRecorder** do nové polohy pisátka udané parametrem **Y**. Hodnota parametru **Y** je vztažena k souřadnému systému bitmapy a je na zodpovědnosti programátora, aby ležela uvnitř okénka zapisovače.

6.2.6. WrToRecorder

```
procedure WrToRecorder (X, Y: integer);
```

Metoda **WrToRecorder** je určena k vykreslení kroku souřadnicového zapisovače. Parametr **X** udává horizontální souřadnici pisátka a parametr **Y** vertikální souřadnici pisátka. Metoda posune voláním **ShifRecorderLeft** obsah okénka vlevo tak, aby nová souřadnice **X** ležela na pravém okraji okénka a vykreslí úsečku od polohy pisátka při posledním volání **WrDifXToRecorder** nebo **WrToRecorder** do nové polohy pisátka udané parametrem **Y**. Hodnota parametru **Y** je vztažena k souřadnému systému bitmapy a je na zodpovědnosti programátora, aby ležela uvnitř okénka zapisovače.