

# uSimT03

## JEDNOTKA IMPLEMENTUJÍCÍ OBJEKTY PRO SIMULACI TERMINÁLU TERM03 NA PC

Příručka uživatele a programátora



**SofCon<sup>®</sup> spol. s r.o.**  
Střešovická 49  
162 00 Praha 6  
tel/fax: +420 220 180 454  
E-mail: [sofcon@sofcon.cz](mailto: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: 09.06.2003

Datum posledního uložení dokumentu: 09.06.2003

(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	4
1.1. Revize dokumentu	4
1.2. Účel dokumentu	4
1.3. Rozsah platnosti	4
1.4. Související dokumenty	4
2.Termíny a definice	4
3.Úvod	5
4.Popis konstant a typů	5
5.Popis objektu tSimDispT03	5
5.1. Proměnné	5
5.2. Metody	6
5.2.1. Init	6
5.2.2. Done	6
5.2.3. InitHwProc	7
5.2.4. DoneHwProc	7
5.2.5. MoveVideoRWM_ToHw	7
5.2.6. mDispContr	7
5.2.7. mFILight	7
5.2.8. mPutRTS, mPutDTR, mGetDCD, mGetDSR, mGetCTS, mGetRI	7
5.2.9. mWriteEEPROM	7
5.2.10. mReadEEPROM	7
6.Popis objektu tSimKeybT03	8
6.1. Proměnné	8
6.2. Metody	8
6.2.1. Init	8
6.2.2. HwInit	8
6.2.3. KTick	8
7.funkce Cr_SimTermT03	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	09.06.2003	Úprava dokumentu dle ISO9000. Přidán popis fce Cr_SimTermT03 a metod pro nastavení / čtení modemových signálů.

### 1.2. Účel dokumentu

---

Tento dokument slouží jako popis jednotky implementující objekty pro simulaci terminálu Term03 na PC.

### 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 uCharBuf, uATerm, uKeybT03, uDispT03 a uTermT03.

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 implementuje objekty pro simulaci klávesnice a displeje terminálu TERM03 na PC **tSimKeybT03** a **tSimDispT03**. Objekty jsou potomky objektových typů **tKeybT03** a **tDispT03**. Pro simulaci terminálu na PC stačí vytvořit jejich instance a předat je objektu terminálu **tTermT03** místo instancí objektů **tKeybT03** a **tDispT03**.

Klávesnice terminálu je nahrazena klávesnicí počítače PC. Jednotlivá tlačítka terminálu jsou nahrazena příslušnými tlačítky klávesnice PC až na tyto výjimky:

terminál	klávesnice PC
START	Alt - A
Shift - Enter	Ctrl - Enter

Dále je zavedena kombinace tlačítek Alt - X, která je určena k ukončení programu.

Displej terminálu je nahrazen okénkem na monitoru PC. V okénku nad displejem je zobrazen nastavený kontrast displeje.

Zděděné metody jsou popsány v dokumentacích k jednotkám **uATerm**, **uDispT03** a **uKeybT03**.

### 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.

```
zAltXPc    =#$2d;
zAltAPc    =#$1E;
zAltSPc    =#$1F;
zCtrlEnterPc =#$0A;
zDelPc     =#$53;
```

Tyto konstanty definují kódy některých kláves na klávesnici PC. Jsou používány interně.

### 5. Popis objektu tSimDispT03

```
type
  pSimDispT03=^tSimDispT03;
  tSimDispT03=object(tDispT03);
```

Objektový typ **tSimDispT03** je potomkem objektového typu displeje terminálu TERM03 **tDispT03**. Slouží k simulaci displeje na monitoru PC.

#### 5.1. Proměnné

```
vSimT03GraphRect:tGraphRect;
```

Proměnná **vSimT03GraphRect** obsahuje rámeček okénka displeje na monitoru PC.

```
vOfsX:Word;
```

`vOfsY:Word;`

Proměnné **vOfsX** a **vOfsY** obsahují posunutí zobrazení displeje od počátku souřadného systému obrazovky v pixelech.

`vDoubleSize:Boolean;`

Proměnná **vDoubleSize** obsahuje příznak zobrazení displeje v dvojnásobné velikosti.

`SimTERM03Win:pGraphWin;`

Proměnná **SimTerm03Win** obsahuje odkaz na okénko, ve kterém je displej zobrazen.

`VisibleContr:Byte;`

Proměnná **VisibleContr** obsahuje zobrazený kontrast displeje.

`EEPROMFile:File;`

Proměnná **EEPROMFile** představuje soubor, který nahrazuje sériovou paměť EEPROM terminálu TERM03.

## 5.2. Metody

---

### 5.2.1. Init

```
constructor Init(TermOwner:pATerm;  
                 CharColls,CharRows:Byte;  
                 NAdr:Word;EnIniHwProc:Boolean;  
                 OfsX,OfsY:Word;  
                 OrgX,OrgY,OrgW,OrgH:Integer;  
                 QDoubleSize:Boolean);
```

Konstruktor **Init** inicializuje objekt, nastavuje jeho proměnné a otevře soubor EEPROM.BIN nahrazující sériovou paměť EEPROM terminálu TERM03. Pokud soubor neexistuje, vytvoří ho. Parametr **TermOwner** obsahuje odkaz na vlastníka objektu, objekt terminálu, parametry **CharColls** a **CharRows** rozměry znakového rastru displeje, parametr **Adr** obsahuje adresu displeje v I/O prostoru a je zaveden pro zpětnou kompatibilitu, parametr **EnIniHwProc** definuje, zda se má volat metoda **InitHwProc**. Jeho nastavení na false umožňuje toto volání potlačit a metodu **InitHwProc** volat dodatečně např. po inicializaci dalších proměnných v konstruktoru potomka tohoto objektového typu. Parametry **OfsX** a **OfsY** udávají posunutí zobrazení displeje vůči počátku souřadného systému monitoru (levý horní roh) v pixelech. Parametry **OrgX**, **OrgY**, **OrgW**, **OrgH** definují umístění a rozměry okénka pro zobrazení displeje v pixelech a parametr **QDoubleSize** definuje požadavek na zobrazení displeje v dvojnásobné velikosti.

### 5.2.2. Done

`destructor Done;virtual;`

Destruktor **Done** uzavře soubor nahrazující sériovou paměť EEPROM a zruší objekt.

### 5.2.3. InitHwProc

```
procedure InitHwProc;virtual;
```

Metoda **InitHwProc** slouží k inicializaci hardware displeje. V tomto objektu vytvoří a vykreslí grafické okénko pro zobrazení displeje na monitoru PC.

### 5.2.4. DoneHwProc

```
procedure DoneHwProc;virtual;
```

Metoda **DoneHwProc** slouží k ukončení činnosti hardware displeje. V tomto objektu zruší grafické okénko pro zobrazení displeje na monitoru PC.

### 5.2.5. MoveVideoRWM\_ToHw

```
procedure MoveVideoRWM_ToHw;virtual;
```

Metoda **MoveVideoRWM\_ToHw** slouží k přesunu dat pro zobrazení na hardware displeje. V tomto objektu vykreslí data na monitor PC.

### 5.2.6. mDispContr

```
procedure mDispContr(B:byte); virtual;
```

Metoda **mDispContr** nastavuje kontrast displeje. Nastavený kontrast je vypsán nad zobrazením displeje na monitoru. Hodnota zobrazeného kontrastu je omezena na rozsah 0 až 99.

### 5.2.7. mFlLight

```
procedure mFlLight(b:byte);virtual;
```

Metoda **mFlLight** slouží k nastavení jasu displeje.

### 5.2.8. mPutRTS, mPutDTR, mGetDCD, mGetDSR, mGetCTS, mGetRI

Metody pro nastavování a čtení modemových signálů nemají v simulátoru žádný význam. Procedury nastavující modemové signály nevykonávají žádnou činnost, funkce pro čtení modemových signálů vrací vždy hodnotu FALSE.

### 5.2.9. mWriteEEPROM

```
procedure mWriteEEPROM(A:Word;D:byte);virtual;
```

Metoda **mWriteEEPROM** zapisuje byte do souboru nahrazujícího sériovou paměť EEPROM. Parametrem **A** je předána adresa a parametrem **D** zapisovaný byte.

### 5.2.10. mReadEEPROM

```
function mReadEEPROM(A:Word):byte;virtual;
```

Metoda **mReadEEPROM** přečte byte ze souboru nahrazujícího sériovou paměť EEPROM. Parametrem **A** je předána adresa. Přečtený byte je vrácen jako návratová hodnota.

## 6. Popis objektu tSimKeybT03

---

type

```
pSimKeybT03=^tSimKeybT03;  
tSimKeybT03=object(tKeybT03);
```

Objektový typ **tSimKeybT03** je potomkem objektového typu klávesnice TERM03 **tKeybT03**. Slouží k simulaci klávesnice terminálu TERM03 klávesnicí PC.

### 6.1. Proměnné

---

FlNul:Boolean;

Proměnná **FlNul** je využívána interně pro dekódování kláves stisknutých na klávesnici PC.

FlgEndPtr:^Boolean;

Proměnná **FlgEndPtr** je ukazatel na proměnnou pro ukončení programu.

### 6.2. Metody

---

#### 6.2.1. Init

```
constructor Init(TermOwner:pATerm;Len:Word; var AFlgEnd:Boolean);
```

Konstruktor **Init** inicializuje objekt a nastavuje jeho proměnné. Parametr **TermOwner** obsahuje odkaz na vlastníka objektu, objekt terminálu, parametr **Len** délku alokovaného vyrovnávacího bufferu klávesnice a **AFlgEnd** je proměnná pro ukončení programu. Tato proměnná bude nastavena na true po stisku kombinace tlačítek Alt - X.

#### 6.2.2. HwInit

```
procedure HwInit;virtual;
```

Metoda **HwInit** je prázdná. Překrývá metodu předka **tKeybT03.HwInit**, která instaluje vlastní obsluhu přerušení INT 08h.

#### 6.2.3. KTick

```
procedure KTick;virtual;
```

Metoda **KTick** zabezpečuje cyklické čtení stisknutých kláves z hardware klávesnice. V tomto objektu čte kódy stisknutých kláves na klávesnici PC.

## 7. funkce Cr\_SimTermT03

---

```
function Cr_SimTermT03(Colls, Rows:integer; NAdr:Word;
```



var AFlgEnd:Boolean):PTermT03;

Funkce **Cr\_SimTermT03** vytvoří instance objektů terminálu klávesnice a displeje pro simulátor terminálu TERM03. Jako parametry jsou předávány rozměry znakového rastru displeje (**Cols** - sloupce, **Rows** - řádky) s adresou terminálu v IO prostoru **NAdr**. **AFlgEnd** je příznak ukončení programu po stisku **Alt - X**. Funkce vrací ukazatel na vytvořenou instanci objektu terminálu.