

T11MDrv

OVLADAČE TERMINÁLU TOUCH11 MONO PRO VIZUALIZAČNÍ KNIHOVNY PRO JEDNOTKU KIT

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: 23.01.2004

Datum posledního uložení dokumentu: 23.01.2004

(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
3.1. Účel knihovny T11MDrv	5
3.2. Terminál Touch11 Mono	5
3.2.1. Ovladač displeje	5
3.2.2. Použití ovladačů terminálu Touch11 Mono	5
3.3. Nastavení terminálu Touch11 Mono	6
4.Reference	7
4.1. Třídy	7
4.1.1. Třída TT11MonoDispDriver	7
4.1.1.1. Položka TT11MonoDispDriver.ioBase	7
4.1.1.2. Položka TT11MonoDispDriver.Cache	7
4.1.1.3. Položka TT11MonoDispDriver.CacheSize	8
4.1.1.4. Položka TT11MonoDispDriver.TempState	8
4.1.1.5. Položka TT11MonoDispDriver.TempTimer	8
4.1.1.6. Položka TT11MonoDispDriver.ContrastOfs	8
4.1.1.7. Položka TT11MonoDispDriver.Contrast	8
4.1.1.8. Položka TT11MonoDispDriver.Brightness	8
4.1.1.9. Konstruktor TT11MonoDispDriver.Init	8
4.1.1.10. Destruktor TT11MonoDispDriver.Done	9
4.1.1.11. Metoda TT11MonoDispDriver.Initialize	9
4.1.1.12. Metoda TT11MonoDispDriver.Finalize	10
4.1.1.13. Metoda TT11MonoDispDriver.Tick	10
4.1.1.14. Metoda TT11MonoDispDriver.Refresh	10
4.1.1.15. Metoda TT11MonoDispDriver.GetContrast	11
4.1.1.16. Metoda TT11MonoDispDriver.SetContrast	11
4.1.1.17. Metoda TT11MonoDispDriver.GetBrightness	12
4.1.1.18. Metoda TT11MonoDispDriver.SetBrightness	12

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.00	Cr	21.01.2004	První vydání

1.2. Účel dokumentu

Tento dokument slouží jako popis knihovny T11MDrv, která je součástí balíku vizualizačních knihoven pro jednotku KIT.

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 IoDrv a Controls.
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

3.1. Účel knihovny T11MDrv

Knihovna T11MDrv obsahuje ovladače displeje terminálu Touch11 Mono fy SofCon pro použití s vizualizačními knihovnami pro jednotku KIT.

3.2. Terminál Touch11 Mono

Terminál Touch11 Mono je terminál s monochromatickým grafickým displejem s rozlišením 320x240 pixelů s 1-bitovou hloubkou barev. Terminál je dále vybaven dotykovým panelem. Ovladač dotykového panelu je implementován v knihovně TPDrv.

3.2.1. Ovladač displeje

Knihovna T11MDrv obsahuje třídu **TT11MonoDispDriver**, která implementuje ovladač grafického displeje terminálu. Ovladač vychází ze třídy **TCachedDisplayDriver** implementované v knihovně IoDrv.

Ovladač **TT11MonoDispDriver** umožňuje řízení kontrastu a jasu displeje. Ovladač měří v nastavených periodách teplotu displeje a podle naměřených vzorků automaticky upravuje kontrast, tak aby nedocházelo k jeho kolísání.

3.2.2. Použití ovladačů terminálu Touch11 Mono

Následující příklad ukazuje, jak vytvořit základní komponentu aplikace **TApplication** pro terminál Touch11 Mono.

```
const
  ioTouch11 = $2300; { Bázová adresa terminálu v I/O prostoru }
var
  App : PApplication;

App :=
  New( PApplication, Init (
    New( PInputDriver, Init(
      nil,
      New( PTouchPanelDriver, Init( ioTouch11 )
    )),
    New( PT11MonoDispDriver, Init( ioTouch11 ) ),
    @g_AppSettings
  ));
```

Pozn.:

- Ovladač **TTouchPanelDriver** je definován v knihovně TPDrv.
- Proměnná **g_AppSettings** obsahuje nastavení ovladače displeje a klávesnice terminálu. Tato proměnná musí být umístěna v zálohované paměti RAM.

3.3. Nastavení terminálu Touch11 Mono

Terminál Touch11 Mono jako celek umožňuje nastavovat následující parametry.

- Úroveň kontrastu displeje
- Úroveň jasu displeje (úroveň podsvícení)
- Čas pro automatické snížení jasu při nečinnosti uživatele
- Úroveň jasu při nečinnosti uživatele
- Kalibrační konstanty dotykového panelu
- Parametry řadiče dotykového panelu

Všechny tyto parametry lze nastavit pomocí k tomu určených metod ovladače displeje a dotykového panelu.

Pro jednoduchou nastavení terminálu Touch11 byla vytvořena knihovna T11Setup, obsahující tři speciální stránky, na kterých je možné tyto parametry terminálu modifikovat viz. následující obrázek:

Stránka pro nastavení parametrů displeje:

The screenshot shows a menu titled "Nastavení displeje terminálu Touch 11". It contains four settings, each with a slider and a numerical value:

Parametr	Úroveň
Kontrast displeje:	(Slider)
Jas výbojky:	1
Automaticke vypnutí:	56
Jas ve vypnutém stavu:	1

At the bottom left, there are two status indicators: "Verze: N/A" and "Teplota: N/A". At the bottom right, there is a button labeled "Uložit nastavení". At the very bottom, there is a navigation bar with three buttons: "Zpět", "Panel", and "Kalibrace".

4. Reference

4.1. Třídy

4.1.1. Třída TT11MonoDispDriver

Třída **TT11MonoDispDriver** implementuje ovladač displeje terminálu Touch11 Mono. Tato třída vychází z základní třídy pro implementaci ovladačů displejů s vyrovnávací pamětí **TCachedDisplayDriver** (viz. dokumentace ke knihovně IoDrv)

```
TT11MonoDispDriver = ^TT11MonoDispDriver;
TT11MonoDispDriver = object( TCachedDisplayDriver )
public
    IoBase      : Word;
    Cache       : Pointer;
    CacheSize   : Word;

    TempTimer   : TTimerEx;
    TempState   : Byte;

    ContrastOfs : Integer;
    Contrast    : Integer;
    Brightness  : Byte;

    constructor Init( AIoBase: Word );
    destructor Done; virtual;

    function Initialize: Boolean; virtual;
    procedure Finalize; virtual;
    procedure Refresh( const R: TRect ); virtual;
    procedure Tick; virtual;

    function GetContrast: Integer; virtual;
    procedure SetContrast( AValue: Integer ); virtual;
    function GetBrightness: Integer; virtual;
    procedure SetBrightness( AValue: Integer ); virtual;
end;
```

4.1.1.1. Položka TT11MonoDispDriver.IoBase

Položka **IoBase** obsahuje základní adresu řadiče terminálu v I/O prostoru. Položka je inicializovaná konstruktorem a je určena pouze pro čtení.

```
IoBase      : Word;
```

4.1.1.2. Položka TT11MonoDispDriver.Cache

Položka **Cache** obsahuje ukazatel na vyrovnávací paměť pro videodata zapisovaná do řadiče displeje alokovanou na hromadě. Tato položka je inicializovaná konstruktorem instance a je určena pouze pro čtení.

```
Cache       : Pointer;
```

4.1.1.3. Položka TT11MonoDispDriver.CacheSize

Položka **CacheSize** obsahuje velikost vyrovnávací paměti pro videodata zapisovaná do řadiče displeje, tj. velikost bloku, na který se odkazuje položka Cache (viz. kapitola 4.1.1.2). Položka je inicializovaná konstruktorem instance a je určena pouze pro čtení.

```
CacheSize : Word;
```

4.1.1.4. Položka TT11MonoDispDriver.TempState

Položka **TempState** obsahuje stav automatu pro měření teploty displeje. Položka je interní a je určena pouze pro čtení.

```
TempState : Byte;
```

4.1.1.5. Položka TT11MonoDispDriver.TempTimer

Položka **TempTimer** obsahuje strukturu časovače určenou pro účely měření teploty displeje. Položka je interní a je určena pouze pro čtení.

```
TempTimer : TTimerEx;
```

4.1.1.6. Položka TT11MonoDispDriver.ContrastOfs

Položka **ContrastOfs** obsahuje optimální hodnotu kontrastu displeje, vypočítanou podle změřené teploty displeje. Hodnota položky je aktualizována v pravidelných intervalech automatem řadiče displeje a je určena pouze pro čtení.

```
ContrastOfs : Integer;
```

4.1.1.7. Položka TT11MonoDispDriver.Contrast

Položka **Contrast** obsahuje aktuální hodnotu jasu displeje. Položka je interní. Pro čtení a nastavení kontrastu displeje slouží metody **GetContrast** a **SetContrast** (viz. kapitoly 4.1.1.15 a 4.1.1.16).

```
Contrast : Integer;
```

4.1.1.8. Položka TT11MonoDispDriver.Brightness

Položka **Brightness** obsahuje aktuální hodnotu jasu displeje. Položka je interní. Pro čtení a nastavení jasu displeje slouží metody **GetBrightness** a **SetBrightness** (viz. kapitoly 4.1.1.17 a 4.1.1.18).

```
Brightness : Byte;
```

4.1.1.9. Konstruktor TT11MonoDispDriver.Init

Konstruktor **Init** provádí inicializaci instance třídy.

```
constructor Init( AIOBase: Word );
```


Parametry:

AIoBase Bázová adresa řadiče terminálu v I/O prostoru.

Návratové hodnoty:

Konstruktor nevrací žádnou hodnotu.

Poznámky:

Konstruktor vytvoří instanci textového kurzoru a kreslicího povrchu (viz. dokumentace ke knihovně IoDrv). Nastaví implicitní jas a kontrast displeje.

4.1.1.10. Destruktor TT11MonoDispDriver.Done

Destruktor **Done** provádí uvolnění prostředků alokovaných konstruktorem.

```
destructor Done; virtual;
```

Parametry:

Destruktor nemá žádné parametry.

Návratové hodnoty:

Destruktor nevrací žádnou hodnotu.

Poznámky:

Destruktor uvolní vytvořené instance textového kurzoru a kreslicího povrchu.

4.1.1.11. Metoda TT11MonoDispDriver.Initialize

Metoda **Initialize** provádí inicializaci hardware řadiče displeje terminálu Touch11 Mono.

```
function Initialize: Boolean; virtual;
```

Parametry:

Metoda nemá žádné parametry.

Návratové hodnoty:

Metoda vrací hodnotu True v případě úspěšné inicializace řadiče displeje.

Poznámky:

Metoda **Initializace** předefinovává metodu **Initialize** bázové třídy **TDisplayDriver** (viz. dokumentace ke knihovně IoDrv).

Metoda provede nastavení řadiče displeje do správného pracovního režimu a nastaví implicitní hodnoty jasu a kontrastu.

4.1.1.12. Metoda TT11MonoDispDriver.Finalize

Metoda **Finalize** provádí deinitializaci hardware řadiče displeje terminálu Touch11 Mono.

```
procedure Finalize; virtual;
```

Parametry:

Metoda nemá žádné parametry.

Návratové hodnoty:

Metoda nevrací žádnou hodnotu

Poznámky:

Metoda **Finalize** předefinovává metodu **Finalize** báze třídy **TDisplayDriver** (viz. dokumentace ke knihovně IoDrv).

Metoda provede zhasnutí výbojky terminálu.

4.1.1.13. Metoda TT11MonoDispDriver.Tick

Metoda **Tick** provádí jeden krok automatu ovladače displeje.

```
procedure Tick; virtual;
```

Parametry:

Metoda nemá žádné parametry.

Návratové hodnoty:

Metoda nevrací žádnou hodnotu.

Poznámky:

Metoda **Tick** volá metodu **Tick** svého předka, tj. třídy **TCachedDisplayDriver** (viz. dokumentace ke knihovně IoDrv). a navíc přidává automat pro pravidelné měření teploty displeje pro účely korekce kontrastu.

4.1.1.14. Metoda TT11MonoDispDriver.Refresh

Metoda **Refresh** slouží k překopírování části kopie videopaměti do paměti řadiče displeje.

```
procedure Refresh( const R: TRect ); virtual;
```

Parametry:

R Obdélník ohraničující oblast videopaměti, kterou je potřeba přenést do paměti řadiče displeje.

Návratové hodnoty:

Metoda nevrací žádnou hodnotu.

Poznámky:

Metoda **Refresh** předefinovává abstraktní metodu **Refresh** předka, tj. třídy **TCachedDisplayDriver** (viz. dokumentace ke knihovně IoDrv).

4.1.1.15. Metoda TT11MonoDispDriver.GetContrast

Metoda **GetContrast** vrací aktuální hodnotu kontrastu displeje.

```
function GetContrast: Integer; virtual;
```

Parametry:

Metoda nemá žádné parametry.

Návratové hodnoty:

Metoda vrací aktuální hodnotu kontrastu displeje. Hodnota může být v rozsahu -255 až 255. Hodnota -255 odpovídá minimálnímu kontrastu. Hodnota 255 odpovídá maximálnímu kontrastu. Hodnota 0 odpovídá optimálnímu kontrastu zjištěného měřením teploty displeje.

Poznámky:

Metoda **GetContrast** předefinovává metodu **GetContrast** báze třídy **TDisplayDriver** (viz. dokumentace ke knihovně IoDrv).

4.1.1.16. Metoda TT11MonoDispDriver.SetContrast

Metoda **SetContrast** slouží pro nastavení aktuální hodnoty kontrastu displeje.

```
procedure SetContrast( AValue: Integer ); virtual;
```

Parametry:

AValue Požadovaná hodnota kontrastu. Hodnota může být v rozsahu -255 až 255. Hodnota -255 odpovídá minimálnímu kontrastu. Hodnota 255 odpovídá maximálnímu kontrastu. Hodnota 0 odpovídá optimálnímu kontrastu zjištěného měřením teploty displeje.

Návratové hodnoty:

Metoda nevrací žádnou hodnotu.

Poznámky:

Metoda **SetContrast** předefinovává metodu **SetContrast** báze třídy **TDisplayDriver** (viz. dokumentace ke knihovně IoDrv).

4.1.1.17. Metoda TT11MonoDispDriver.GetBrightness

Metoda **GetBrightness** vrací aktuální hodnotu jasu (úroveň podsvícení) displeje.

```
function GetBrightness: Integer; virtual;
```

Parametry:

Metoda nemá žádné parametry.

Návratové hodnoty:

Metoda vrací aktuální hodnotu jasu displeje. Vracená hodnota je v rozsahu 0 až 16. Hodnota 0 odpovídá vypnuté výbojce. Hodnota 16 odpovídá maximálnímu jasu displeje.

Poznámky:

Metoda **GetBrightness** předefinovává metodu **GetBrightness** báze třídy **TDisplayDriver** (viz. dokumentace ke knihovně IoDrv).

4.1.1.18. Metoda TT11MonoDispDriver.SetBrightness

Metoda **SetBrightness** slouží pro nastavení aktuální hodnoty jasu (úroveň podsvícení) displeje.

```
procedure SetBrightness( AValue: Integer ); virtual;
```

Parametry:

AValue	Požadovaná hodnota jasu. Hodnota se může pohybovat v rozsahu 0 až 16. Hodnota je 0 odpovídá vypnuté výbojce. Hodnota 16 odpovídá maximálnímu jasu.
--------	--

Návratové hodnoty:

Metoda nevrací žádnou hodnotu.

Poznámky:

Metoda **SetBrightness** předefinovává metodu **SetBrightness** báze třídy

TDisplayDriver (viz. dokumentace ke knihovně IoDrv).