

WATCHDOG

JEDNOTKA PRO UŽIVATELSKÉ OBSLOUŽENÍ OBVODU WATCHDOG

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: 20.05.2003

Datum posledního uložení dokumentu: 20.05.2003

(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.Princip činnosti jednotky	6
4.Popis konstant a typů	6
5.Procedura InitUserWD	6
6.Příklad použití jednotky	6

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	We		První vydání.
1.10	1.XX	Tu	20.05.2003	Úprava dokumentu dle ISO9000.

1.2. Účel dokumentu

Tento dokument slouží jako popis jednotky pro uživatelské obslužení obvodu WatchDog.

1.3. Rozsah platnosti

Určen pro programátory a uživatele programového vybavení SofCon.

1.4. Související dokumenty

Pro čtení tohoto dokumentu není potřeba číst žádný další manuál, ale je potřeba orientovat se v používání programového vybavení SofCon.

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. Princip činnosti jednotky

Po startu počítače KitV40 i KIT386 bios zařadí obnovování monostabilního obvodu WatchDog do přerušeni časovače a tam také zůstává i v uživatelském programu, pokud to není změněno. Tím je sice zajištěno bezchybné a periodické obnovování stavu obvodu, ale není zajištěno hlídání správné funkce uživatelského programu. Je totiž známé, že při havárii programu procedury vyvolávané přerušením se zhroutí až jako poslední. Z toho důvodu byla vytvořena jednotka WatchDog, která umožňuje snadné navázání vyhodnocení chodu uživatelského programu do procedury obsluhy obvodu.

Jednotka ponechává původní obsluhu obvodu vyvolávanou přerušením časovače, ale před ní volá uživatelskou pascalskou funkci. Podle výsledku funkce zavolá původní obsluhu a ukončí přerušeni, nebo také zavolá ještě jednou obsluhu Watchdogu ale potom zakáže přerušeni a skončí RunError(255). Při zakázaném přerušeni není vyvolána další obsluha a WatchDog vyvolá přibližně po jedné vteřině hardwarový reset celého systému. Exit procedura vyvolaná RunErrorem má dost času stroj před resetem zastavit.

Uživatelská funkce je obyčejná pascalská funkce typu boolean. Musí být deklarovaná jako far, nebo být v interface sekci unitu. Při výsledku funkce true se normálně obcerstvuje Watchdog, při false se vyvolá reset.

4. Popis konstant a typů

```
const
  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.

```
type
  tWDFunction = function:boolean;
```

5. Procedura InitUserWD

```
procedure InitUserWD(UserFunction:tWDFunction);
```

tWDFunction je obyčejná pascalská funkce typu boolean. **InitUserWD** je inicializační funkce jednotky. Jako parametr se jí předává uživatelská funkce vyhodnocující stav programu.

6. Příklad použití jednotky

```
program TestWD;
uses
  Crt,
  WatchDog;
const
  cPocitadloWD = 10;           {pocet projiti MyWDFunction, nez to
                               shodi WatchDog}
  flProcesWD   : boolean = true; {flag, ktery proces nahazuje a
                               MyWDFunction shazuje}
  flEnd        : boolean = false;
```

```
function MyWDFunction:boolean;far;
const
  PocitadloWD : byte =cPocitadloWD;
begin
  if not flProcesWD
  then
    begin
      {$R-,Q-}
      dec(PocitadloWD);
      {$R+,Q+}
      if PocitadloWD=0
      then MyWDFunction:=false
      else MyWDFunction:=true;
    end
  else
    begin
      flProcesWD:=false;
      PocitadloWD:=cPocitadloWD;
      MyWDFunction:=true;
    end;
end;

begin
  InitUserWD(MyWDFunction);
  repeat
    flProcesWD:=true;
    delay(100);
    write('jeste ziju ');
    if KeyPressed then
      begin
        if #20=ReadKey then flEnd:=true; {po stisku mezery to ukonci
          tento cyklus}
        end;
      until flEnd;
      Writeln;
      Writeln('Konec obnovovani flProcesWD, ted zabere WatchDog. ');
      repeat
        until false;
      end.
end.
```