

ChnTypes

DEFINICE ZÁKLADNÍCH KONSTANT A
TYPŮ PRO KOMUNIKAČNÍ KNIHOVNY

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: 28.07.2004

Datum posledního uložení dokumentu: 28.07.2004

(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	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
4.1.	Konstanty sériového obvodu i8250	6
4.1.1.	Implicitní adresy sériových portů	6
4.1.2.	Konstanty offsetu registrů sériových portů	6
4.1.3.	Masky pro nastavení IER (Interrupt Enable) registru	7
4.1.4.	Masky pro nastavení MCR (Modem Control) registru	8
4.1.5.	Masky pro testování IIR (Interrupt Identification) registru	8
4.1.6.	Masky pro testování LSR (Line Status) registru	8
4.1.7.	Masky pro testování MSR (Modem Status) registru	9
4.2.	Konstanty sériového obvodu i8251	9
4.2.1.	Adresy registrů obvodu 8251	9
4.2.2.	Masky pro testování SST (Serial Status) registru	10
4.2.3.	Masky pro nastavování SCM (Serial Command) registru	10
4.2.4.	Masky pro nastavení SIMK (Serial Interrupt Mask) registru	10
4.3.	Konstanty společné pro oba obvody 8250 i 8251	11
5.	Typy	11

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	Wi		První vydání.
1.10	1.XX	Tu	22.05.2003	Úprava dokumentu dle ISO9000.
1.20	1.XX	Wil	28.07.2004	Konstanty pro obsluhu řadiče přerušení přesunuty do HWsyst. Indexy hardwarového přerušení (IRQ) pro sériové porty přesunuty do HWsyst.

1.2. Účel dokumentu

Tento dokument slouží jako popis jednotky definující základní konstanty a typy pro komunikační knihovny.

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.

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

Knihovna ChnTypes obsahuje standardní definice typů a konstant, které používají komunikační knihovny ChnVirt, ChnCom, ChnV40, ChnPrt a další.

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.

4.1. Konstanty sériového obvodu i8250

4.1.1. Implicitní adresy sériových portů

```
ACom1   = $3f8;
          Konstanta Acom1 definuje adresu sériového portu COM 1.
ACom2   = $2f8;
          Konstanta Acom2 definuje adresu sériového portu COM 2.
ACom3   = $3e8;
          Konstanta Acom3 definuje adresu sériového portu COM 3.
ACom4   = $3e0;
          Konstanta Acom4 definuje adresu sériového portu COM 4.
ACom5   = $2f0;
          Konstanta Acom5 definuje adresu sériového portu COM 5.
ACom6   = $2e8;
          Konstanta Acom6 definuje adresu sériového portu COM 6.
ACom7   = $2e0;
          Konstanta Acom7 definuje adresu sériového portu COM 7.
ACom8   = $260;
          Konstanta Acom8 definuje adresu sériového portu COM 8.

ACom    : array [1..8] of Word =
          ( Acom1, Acom2, Acom3, Acom4, Acom5, Acom6, Acom7, Acom8 );
          Pole ACom definuje pole konstant adres COM portů.
```

4.1.2. Konstanty offsetu registrů sériových portů

Následující konstanty definují offset (tzv. posunutí) od základní adresy (viz „4.1.1 Implicitní adresy sériových portů“) pro jednotlivé registry obvodu i8250.

```
ComTHR   = $00;
          Konstanta definuje offset pro Transmit Holding Register (THR) - Write.
ComRBR   = $00;
          Konstanta definuje offset pro Receiver Buffer Register (RBR) - Read.
ComIER   = $01;
          Konstanta definuje offset pro Interrupt Enable Register (IER) – Read and Write.
ComIIR   = $02;
          Konstanta definuje offset pro Interrupt Identification Register (IIR) - Read.
```

ComFCR = \$02;
 Konstanta definuje offset pro FIFO Control Register (FCR) – Write.

ComLCR = \$03;
 Konstanta definuje offset pro Line Control Register (LCR) – Read and Write.

ComMCR = \$04;
 Konstanta definuje offset pro Modem Control Register (MCR) - Read and Write.

ComLSR = \$05;
 Konstanta definuje offset pro Line Status Register (LSR) – Read only.

ComMSR = \$06;
 Konstanta definuje offset pro Modem Status Register (MSR) – Read only.

4.1.3. Masky pro nastavení IER (Interrupt Enable) registru

Masky pro nastavování registru IER ale i dalších registrů (viz další kapitoly) se dělí na dva typy: "And" masky a "Or" masky. Rozdíl mezi nimi je v použití. "Or" masky se používají pro nastavování jednotlivých bitů příslušných registrů a používá se k nim logická spojka OR. "And" masky se používají buď k nulování jednotlivých bitů příslušných registrů nebo k jejich testování a používá se k nim logická spojka AND. "And" masky jsou v podstatě negované "Or" masky.

Příklad:

```
port[ACom1+ComIER] := port[ACom1+ComIER] or RecOn and TransOff;
```

Nastavení povolení přerušení od přijímače a zakázání přerušení od vysílače pro COM1. Nastavení ostatních bitů je ponecháno.

```
if port[ACom1+ComLSR] and MErrPar <> 0 then {chyba parity}
```

Dekódování chyby parity v LSR registru na COM1.

RecOn = \$01;
 Konstanta definuje "or" masku pro povolení přerušení od přijímače.

RecOff = \$fe;
 Konstanta definuje "and" masku pro zakázání přerušení od přijímače.

TransOn = \$02;
 Konstanta definuje "or" masku pro povolení přerušení od vysílače.

TransOff = \$fd;
 Konstanta definuje "and" masku pro zakázání přerušení od vysílače.

LSROn = \$04;
 Konstanta definuje "or" masku pro povolení přerušení od statusu.

LSROff = \$fb;
 Konstanta definuje "and" masku pro zakázání přerušení od statusu.

MSROn = \$08;
 Konstanta definuje "or" masku pro povolení přerušení od modemových signálů.

MSROff = \$f7;
 Konstanta definuje "and" masku pro zakázání přerušení od modemových signálů.

NoInterrupt = \$00;
 Konstanta definuje hodnotu registru pro potlačení všech přerušení.

4.1.4. Masky pro nastavení MCR (Modem Control) registru

SetDTR = \$01;
Konstanta definuje "or" masku pro nastavení signálu DTR/ na výstupu 8250 na logickou 0.

SetRTS = \$02;
Konstanta definuje "or" masku pro nastavení signálu RTS/ na výstupu 8250 na logickou 0.

SetOut2 = \$08;
Konstanta definuje "or" masku pro nastavení signálu OUT2/ na výstupu 8250 na logickou 0 (povoluje přerušování od 8250).

ResDTR = \$fe;
Konstanta definuje "and" masku pro nastavení signálu DTR/ na výstupu 8250 na logickou 1.

ResRTS = \$fd;
Konstanta definuje "and" masku pro nastavení signálu RTS/ na výstupu 8250 na logickou 1.

ResOut2 = \$f7;
Konstanta definuje "and" masku pro nastavení signálu OUT2/ na výstupu 8250 na logickou 1 (zakazuje přerušování od 8250).

4.1.5. Masky pro testování IIR (Interrupt Identification) registru

MIP = \$01;
Konstanta definuje "and" masku pro identifikaci přerušování.

MTransReady = \$02;
Konstanta definuje "and" masku pro identifikaci, zda o přerušování žádá vysílač.

MRecReady = \$04;
Konstanta definuje "and" masku pro identifikaci, zda o přerušování žádá přijímač.

MErrSTS = \$06;
Konstanta definuje "and" masku pro identifikaci, zda přerušování je způsobeno chybou.

MModSTS = \$00;
Konstanta definuje "and" masku pro identifikaci, zda přerušování je způsobeno změnou modemových signálů.

4.1.6. Masky pro testování LSR (Line Status) registru

MErrLSR = \$1e;
Konstanta definuje "and" masku pro odstranění přebytečných bitů z LSR registru pro dekódování chyby.

MokLSR = \$00;
Konstanta definuje správnou hodnotu pro LSR registr po nastavení daného sériového portu.

MErrOvr = \$02;
Konstanta definuje "and" masku pro dekódování chyby OverRun.

`MErrPar` = `$04;`
Konstanta definuje "and" masku pro dekódování chyby Parity.

`MErrFrm` = `$08;`
Konstanta definuje "and" masku pro dekódování chyby Framing.

`MBrkInt` = `$10;`
Konstanta definuje "and" masku pro dekódování Break Interruptu.

4.1.7. Masky pro testování MSR (Modem Status) registru

`MsDCTS` = `$01;`
Konstanta definuje "and" masku pro dekódování modemového signálu DCTS (Delta Clear to Send).

`MsDDSR` = `$02;`
Konstanta definuje "and" masku pro dekódování modemového signálu DDSR (Delta Data Set Ready).

`MsTERI` = `$04;`
Konstanta definuje "and" masku pro dekódování modemového signálu TERI (Trailing Edge Ring Indicator).

`MsDRLSD` = `$08;`
Konstanta definuje "and" masku pro dekódování modemového signálu DRLSD (Delta Receive Line Signal Detect).

`MsCTS` = `$10;`
Konstanta definuje "and" masku pro dekódování modemového signálu CTS (Clear to Send).

`MsDSR` = `$20;`
Konstanta definuje "and" masku pro dekódování modemového signálu DSR (Data Set Ready).

`MsRI` = `$40;`
Konstanta definuje "and" masku pro dekódování modemového signálu RI (Ring Indicator).

`MsRLSD` = `$80;`
Konstanta definuje "and" masku pro dekódování modemového signálu RLSD (Receive Line Signal Detect).

`MsCD` = `MsRLSD;`
Konstanta definuje "and" masku pro dekódování modemového signálu CD (Carrier Detect).

4.2. Konstanty sériového obvodu i8251

4.2.1. Adresy registrů obvodu 8251

`ASCU` = `$00f0;`
Konstanta definuje základní adresu sériového komunikačního adaptéru i8251.

`ASCUSRB` = `ASCU+0;`
Konstanta definuje adresu SRB (Serial Receive Buffer) registru - Read.

`ASCUSTB` = `ASCU+0;`
Konstanta definuje adresu STB (Serial Transmit Buffer) registru - Write.

ASCUSST = ASCU+1;
Konstanta definuje adresu SST (Serial Status) registru - Read.

ASCUSCM = ASCU+1;
Konstanta definuje adresu SCM (Serial Command) registru - Write.

ASCUSMD = ASCU+2;
Konstanta definuje adresu SMD (Serial Mode) registru – Read and Write.

ASCUSIMK = ASCU+3;
Konstanta definuje adresu SIMK (Serial Interrupt Mask) registru – Read and Write.

4.2.2. Masky pro testování SST (Serial Status) registru

MSCUTReady = \$01;
Konstanta definuje "and" masku pro dekódování odvyšlání znaku.

MSCURReady = \$02;
Konstanta definuje "and" masku pro dekódování přijmutí znaku.

MSCUParErr = \$08;
Konstanta definuje "and" masku pro dekódování chyby Parity.

MSCUOvrErr = \$10;
Konstanta definuje "and" masku pro dekódování chyby OverRun.

MSCUFrmErr = \$20;
Konstanta definuje "and" masku pro dekódování chyby Framing.

MSCUBrkDet = \$40;
Konstanta definuje "and" masku pro dekódování Break Detect.

4.2.3. Masky pro nastavování SCM (Serial Command) registru

MSCUTransOn = \$21;
Konstanta definuje "or" masku pro povolení přerušení od vysílače.

MSCUTransOff= \$fe;
Konstanta definuje "and" masku pro zakázání přerušení od vysílače.

MSCURecOn = \$34;
Konstanta definuje "or" masku pro povolení přerušení od přijímače s vynulováním příznaku chyby.

MSCURecOff = \$fb;
Konstanta definuje "and" masku pro zakázání přerušení od přijímače.

MSCUBrkOn = \$28;
Konstanta definuje "or" masku pro nastavení Break Interruptu.

MSCUBrkOff = \$f7;
Konstanta definuje "and" masku pro zrušení Break Interruptu.

MSCUNoInt = \$20;
Konstanta definuje "or" masku pro zakázání všech přerušení.

4.2.4. Masky pro nastavení SIMK (Serial Interrupt Mask) registru

MSIMKNoInt = \$03;
Konstanta definuje hodnotu pro zakázání přijímače a vysílače.

```
MSIMKOnInt = $00;
```

Konstanta definuje hodnotu pro povolení přijímače a vysílače.

4.3. Konstanty společné pro oba obvody 8250 i 8251

```
ParNone = 0;
```

Konstanta **ParNone** znamená bez parity.

```
ParOdd = 1;
```

Konstanta **ParOdd** znamená lichá parita.

```
ParEven = 2;
```

Konstanta **ParEven** znamená sudá parita.

```
Stop1 = 1;
```

Konstanta **Stop1** znamená jeden stop-bit.

```
Stop2 = 2;
```

Konstanta **Stop2** znamená dva stop-bity.

```
Bits5 = 5;
```

Konstanta **Bits5** znamená 5 datových bitů.

```
Bits6 = 6;
```

Konstanta **Bits6** znamená 6 datových bitů.

```
Bits7 = 7;
```

Konstanta **Bits7** znamená 7 datových bitů.

```
Bits8 = 8;
```

Konstanta **Bits8** znamená 8 datových bitů.

5. Typy

```
tRate = 1..300000;
```

Typ **tRate** definuje intervalový typ pro předání baudové rychlosti.

```
tParity = ParNone..ParEven;
```

Typ **tParity** definuje intervalový typ pro předání typu parity přenosu.

```
tStop = Stop1..Stop2;
```

Typ **tStop** definuje intervalový typ pro předání počtu stopbitů.

```
tLength = Bits5..Bits8;
```

Typ **tLength** definuje intervalový typ pro předání délky slova (počet vysílaných datových bitů pro jeden byte).