

# uFlexPos

## JEDNOTKA PRO PRÁCI S INKREMENTÁLNÍMI ČIDLY NEBO RESOLVERY

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: 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	7
5.1. Metody	7
5.1.1. Init	7
5.1.2. GetIdent	7
5.1.3. ReadTime	7
5.1.4. ReadTimeClr	7
5.1.5. ReadPosition	8
5.1.6. ReadTimeClrPosition	8
5.1.7. ReadAbsolute	8
5.1.8. WriteOutL, WriteOutH	8
5.1.9. ReadInL, ReadInH	8
5.1.10. WriteCmd	8
5.1.11. ReadSts	9
5.1.12. RerouteInputs	9



## 1. O dokumentu

---

### 1.1. Revize dokumentu

---

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	Tu	16.05.2003	První vydání

### 1.2. Účel dokumentu

---

Tento dokument slouží jako popis jednotky pro práci s inkrementálními čidly nebo resolversy připojenými přes desku IOFLEX.

### 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. Úvod

---

Jednotka uFlexPos je určena pro odměřování vzdálenosti pomocí desky IOFLEX01, která obsahuje programovatelné pole Altera FLEX10K10 s 16 rychlými univerzálními vstupy a výstupy. Jednotka obsahuje funkce pro čtení absolutní polohy uvnitř jedné periody a relativní polohy vzhledem k bodu, ve kterém byla provedena kalibrace. Jednotka je navržena pro čtení polohy ve čtyřech nezávislých osách. Deska dále obsahuje časovač, který je možné číst resp. nulovat. Díky tomu je možné použít desku IOFLEX01 nejen na zjišťování polohy, ale také na měření rychlosti. Časovač běží s periodou 1,2 $\mu$ s.

Jednotka dále obsahuje funkce pro zápis na výstupy desky a čtení vstupů desky. Kalibrace, tj. určení polohy nuly souřadnicového systému jednotlivých os je umožněno pomocí funkcí **WriteCmd** a **ReadSts**.

Podrobné technické informace o desce IOFLEX01 jsou uvedeny v hardwarové dokumentaci této desky, v tomto dokumentu se budeme zabývat pouze popisem jednotky.

### 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
  tIOFlexCmd = (cNop, cRes, cResNm, cResNmSig);
```

Typ tIOFlexCmd definuje následující příkazy pro desku IOFLEX01:

- **cNop** – nebude se provádět kalibrace.
- **cRes** – okamžité vynulování registrů polohy.
- **cResNm** – vynulování registrů polohy při prvním průchodu nulovou pozicí resolveru.
- **cResNmSig** – vynulování registrů polohy při prvním průchodu nulovou pozicí resolveru, kterému předcházela náběžná hrana signálu od kalibračního snímače (tj. signálu přivedeného na vstupy Cal1 – Cal4 desky IOFLEX01).

•

```
tIOFlexSts = (sNotCal, sCal);
```

Typ tIOFlexSts definuje stav kanálu desky IOFLEX01. Může nabývat dvou stavů:

- **sNotCal** – byl zadán požadavek na kalibraci, ale ještě nebyla provedena.
- **sCal** – kalibrace už byla provedena, resp. byl zadán příkaz cNop.

---

## 5. Popis objektu

---

```
pIOFlexPos = ^tIOFlexPos;  
tIOFlexPos = object(tObject)
```

Objekt **tIOFlexPos** slouží k obsluze jedné desky IOFLEX01. Při vytvoření instance objektu se volá konstruktor předka **tIOFlex**, který nastavuje básovou adresu desky a ukazatel na začátek konfiguračního souboru pro obvod Altera FLEX10K10. Konfigurační soubor, který určuje funkci tohoto obvodu, je obsažen v souboru *FlexPos.inc*.

---

### 5.1. Metody

---

#### 5.1.1. Init

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

Konstruktor volá předka **tIOFlex**, který nastavuje básovou adresu **Addr** desky v I/O prostoru procesoru a konfiguruje programovatelné pole Flex10K10. Ukazatel FlexSW ukazuje na počátek konfiguračních dat v paměti. Výsledek konfigurace je uložen do proměnné **Result**, která může nabývat těchto hodnot:

Flex_Ok	úspěšná inicializace desky
Flex_ErrCfg	chyba během konfigurace FLEX10K10
Flex_ErrDevice	chyba desky IOFLEX01 nebo neplatná básová adresa

Po úspěšné konfiguraci se ještě zkontroluje zda je správná verze projektu ve FLEX10k10. Pokud ne, uloží se do proměnné **Result** hodnota Flex\_ErrProject.

#### 5.1.2. GetIdent

```
function GetIdent: Byte;
```

Funkce vrací číslo verze projektu ve FLEX10k10.

#### 5.1.3. ReadTime

```
function ReadTime: Longint;
```

Funkce vrátí čas přečtený z desky IOFLEX01. Čas je udáváný v [1,2μs].

#### 5.1.4. ReadTimeClr

```
function ReadTimeClr: Longint;
```

Funkce vrátí čas přečtený z desky IOFLEX01 a zároveň vynuluje čítač času. Čas je udáváný v [1,2μs].

### 5.1.5. ReadPosition

```
function ReadPosition0: Longint;  
function ReadPosition1: Longint;  
function ReadPosition2: Longint;  
function ReadPosition3: Longint;
```

Funkce vrátí polohu pro příslušnou osu v jednotkách odpovídajících použitému resolveru nebo inkrementálnímu čidlu. Poloha je počítána od bodu kalibrace přes libovolný počet period. Poloha je uchovávána v 24b formátu, čímž je zároveň dána maximální měřitelná vzdálenost. Například pro lineární snímače polohy s délkou periody 2mm a 4096 dílků na periodu je maximální měřitelná vzdálenost asi  $\pm 2\text{m}$ .

### 5.1.6. ReadTimeClrPosition

```
procedure ReadTimeClrPosition(var Time,  
                               Pos0, Pos1, Pos2, Pos3: Longint);
```

Procedura vrací čas a polohu přečtenou z desky IOFLEX01 pro všechny čtyři osy. Při čtení času se zároveň vynuluje čítač času. Přečtená poloha je stejná jako v případě funkcí **ReadPosition**, ale procedura **ReadTimeClrPosition** je méně časově náročná než čtyřnásobné volání funkce **ReadPosition**.

### 5.1.7. ReadAbsolute

```
procedure ReadAbsolute(var AbsPos0, AbsPos1, AbsPos2, AbsPos3: integer);
```

Funkce vrátí absolutní polohu uvnitř jedné periody pro všechny osy. Například pro výše zmíněná lineární snímače a 12b A-D převodník (2mm perioda a 4096 dílků na periodu) se bude **AbsPos** pohybovat v rozmezí 0-4095, což v tomto příkladu odpovídá rozmezí 0-1,9995 mm. Tato funkce má smysl pouze při použití resolverů!

### 5.1.8. WriteOutL, WriteOutH

```
procedure WriteOutL(Data: Byte);  
procedure WriteOutH(Data: Byte);
```

Procedura provede zápis jednoho byte na výstupy spodních resp. horních 8b desky IOFLEX01.

### 5.1.9. ReadInL, ReadInH

```
function ReadInL: Byte;  
function ReadInH: Byte;
```

Funkce načte spodních resp. horních 8b ze vstupu desky IOFLEX01.

### 5.1.10. WriteCmd

```
procedure WriteCmd(Cmd0, Cmd1, Cmd2, Cmd3: tIOFlexCmd);
```

Procedura zapíše pro příslušné kanály dané **Cmd**. Takto se dá provést okamžité vynulování registru polohy daného kanálu (tj. kalibrace), nebo vynulování registru polohy vyvolané průchodem nulou resolveru apod. viz. popis typu **tIOFlexCmd**.



### 5.1.11. ReadSts

```
procedure ReadSts(var Sts0, Sts1, Sts2, Sts3: tIOFlexSts);
```

Procedura přečte status jednotlivých kanálů desky IOFLEX01. Po resetu, desky, po kalibraci a po zadání příkazu **cNop** vrací procedura pro příslušnou osu hodnotu **sCal**. Pokud byl zadán požadavek na kalibraci a ještě nebyla provedena kalibrace, vrací procedura hodnotu **sNotCal**.

### 5.1.12. RerouteInputs

```
procedure RerouteInputs(i: byte);
```

Tato procedura může být použita například při měření korekční křivky lineárních snímačů. V tomto případě se na vstupy **Cal3** a **Cal4** připojí výstupy etalonu **A** a **B**. Procedurou **RerouteInputs** se přesměrují vstupy **Ai**, **Bi** na **Cal3** a **Cal4**, kde **i** může nabývat hodnot 0 – 4. Například pro  $i=2$  se přestanou číst vstupy z **A2**, **B2** a místo nich se budou číst vstupy etalonu připojeného na **Cal3**, **Cal4**. Všechny funkce, vracející polohu, budou načítat polohu z čidla1, etalonu, čidla3 a čidla4. Pokud bude etalon umístěn například ve stejné ose jako čidlo1, je možné získat korekční křivku porovnáním hodnot získaných funkcí **ReadPosition0** a **ReadPosition1**. Pro  $i=0$  budou všechny vstupy přesměrovány na původní umístění, tj. **A1**, **B1** – **A4**, **B4**.