

Primjeri s ostvarenjem podsustava za upravljanje vremenom

Ostvariti podsustav za upravljanje vremenom korištenjem 24-bitovnog brojila. Brojilo ima slijedeća svojstva: nalazi se na adresi BROJILO, odbrojava od upisane vrijednosti do 0 frekvencijom od $F = 5 \text{ MHz}$, kad dođe do nule generira prekid te ponovno odbrojava od zadnje upisane vrijednosti.

Rješenja:

pomoćne vrijednosti:

- $F = 5000000$ //frekvencija brojila
- maksimalna vrijednost koja stane u brojilo
 $B_{\text{MAX}} = 2^{24} - 1 = 0\text{ffffff} = 16777215$
- trajanje jednog otkucaja
 $T_1 = 1/F = 1/(5 \text{ MHz}) = 0,2 \cdot 10^{-6} = 2 \cdot 10^{-7} = 200 \cdot 10^{-9} = 200 \text{ ns}$
- najveći ciklus traje:
 $T_{\text{MAX}} = B_{\text{MAX}} * T_1 = 16777215 * 200 * 10^{-9} = 3,355443000 \text{ s}$

Upravljanje vremenom

1. samo sat (kao realna vrijednost)

varijable:

```
sat //sat, realna vr. npr. 2.771
UČITANO = 15000000 //3 sekunde!
//može se staviti i B_MAX ili
//nešto drugo;
//bolje što veća vrijednost
```

```
init() {
    sat = 0
    BROJILO = UČITANO
}
prekid_brojila {
    sat += UČITANO/F //realna vr.
}
```

```
postavi_sat(t) {
    sat = t
    BROJILO = UČITANO
}
dohvati_sat() {
    proslo = (UČITANO - BROJILO) / F
    vrati sat + proslo
}
ažuriraj_i_dohvati_sat2() {
    sat += (UČITANO - BROJILO) / F
    BROJILO = UČITANO
    vrati sat
}
```

2. samo sat (vrijeme u mikrosekundama)

Pretpostavka je da nema prekoračenja u varijablama!

varijable:

```
sat //u mikrosekundama
UČITANO = 15000000
```

```
init() {
    sat = 0
    BROJILO = UČITANO
}

otk_u_us(otkucaja) {
    //račun: otkucaja * T_1 * 1000000
    //kad se uvrsti i sredi:
    vrati otkucaja/5
}
```

```
prekid_brojila {
    sat += otk_u_us(UČITANO)
}

postavi_sat(t) {
    sat = t
    BROJILO = UČITANO
}

dohvati_sat() {
    proslo = otk_u_us(UČITANO - BROJILO)
    vrati sat + proslo
}
```

3. samo sat (vrijeme u sekundama i nanosekundama) + oprezno s velikim brojevima

varijable:

sat = {.s, .ns}

UČITANO = 15000000

F = 5000000 //frekvencija brojila

T1NS = 200 //T1*1000000000

zbroji_sat_i_otk(t, otk) {

 x = t *cjelobrojno dijeljenje*

 x.s += otk / F //otk * T1

 x.ns += (otk MOD F) * T1NS

 ako je x.ns > 1000000000 onda

 x.s += 1

 x.ns -= 1000000000

 }

 vrati x

}

init() {

 sat.s = sat.ns = 0

 BROJILO = UČITANO

}

prekid_brojila {

 sat = zbroji_sat_i_otk(sat, UČITANO)

}

postavi_sat(t) {

 sat.s = t.s

 sat.ns = t.ns

 BROJILO = UČITANO

}

dohvati_sat() {

 vrati zbroji_sat_i_otk(sat, UČITANO - BROJILO)

}

drukčije:

ns += otk * T1NS / F

x.s += ns / 1000 000 000

x.ns = ns MOD 1000000000

ali potencijalni problem s velikim br.

4a. sat i jedan alarm (vrijeme u mikrosekundama, apsolutna/relativna odgoda)

pretvorba otkucaji => mikrosekunde

t = (otk / F) * 1000000 - ali problem otk / F = gubitak preciznosti
 ^^^^^^- u sekundama

t = otk * 1000000 / F - ali problem overflow

kada je F s puno nula to se da prokratiti

F = 5000000 => t = otk * 1000000 / 5000000 = otk / 5

pretvorba mikrosekunde u otkucaji

otk = (tus / 1000000) * F = ... = tus * 5

^^^^^^^^^^^^^^- u sekundama

$T_{MAX} = B_{MAX} * T_1 * 1000000$ //najveći period, u mikrosekundama (oprezno računati)
 sat //u mikrosekundama
 kada //kada treba aktivirati alarm, u „apsolutnom vremenu“
 za_koliko //za koliko mikrosekundi treba aktivirati alarm
 akcija //funkcija koju treba pozvati
 UČITANO //zadnja vrijednost učitana u BR.

1. Rješenje s apsolutnim vremenima

```

otk_u_us(otkucaja) {
    vrati otkucaja / 5
}
us_u_otk(vrijeme_us) {
    vrati vrijeme_us * 5
}
init() {
    sat = 0
    UČITANO = BMAX
    BROJLO = UČITANO
    kada = 0
    akcija = NULL
}
postavi_sat(t) {
    init() //briše se alarm!!!
    sat = t
}
dohvati_sat() {
    proslo = otk_u_us(UČITANO-BROJLO)
    vrati sat + proslo
}
postavi_alarm(za_koliko2, obrada) {
    sat += otk_u_us(UČITANO - BROJLO)
    kada = sat + za_koliko2
    akcije = obrada
    UČITANO = us_u_otk(min(kada-sat, TMAX))
    BROJLO = UČITANO
}
prekid_brojila {
    sat += otk_u_us(UČITANO) //ažuriraj sat

    ako je (kada > 0) onda { //ima alarm?
        ako je (sat >= kada) onda { //akt. alarm?
            kada = 0
            BROJLO = UČITANO = BMAX
            akcija()
        }
        inače ako je (kada - sat < TMAX) onda {
            //mijenjati BROJLO samo ako je neophodno!
            BROJLO = UČITANO = us_u_otk(kada - sat)
        }
        //inače već je u UČITANO vrijednost BMAX
    }
}
  
```

2. Rješenje s relativnim vremenima

```

otk_u_us(otkucaja) {
    vrati otkucaja / 5
}
us_u_otk(vrijeme_us) {
    vrati vrijeme_us * 5
}
init() {
    sat = 0
    UČITANO = BMAX
    BROJLO = UČITANO
    za_koliko = 0
    akcija = NULL
}
postavi_sat(t) {
    init() //briše se alarm!!!
    sat = t
}
dohvati_sat() {
    proslo = otk_u_us(UČITANO-BROJLO)
    vrati sat + proslo
}
postavi_alarm(za_koliko2, obrada) {
    sat += otk_u_us(UČITANO - BROJLO)
    za_koliko = za_koliko2
    akcije = obrada
    UČITANO = us_u_otk(min(za_koliko, TMAX))
    BROJLO = UČITANO
}
prekid_brojila {
    sat += otk_u_us(UČITANO) //ažuriraj sat

    ako je (za_koliko > 0) onda { //ima alarm?
        za_koliko = za_koliko - otk_u_us(UČITANO)
        ako je (za_koliko <= 0) onda { //akt. alarm?
            za_koliko = 0
            BROJLO = UČITANO = BMAX
            akcija()
        }
        inače ako je (za_koliko < TMAX) onda {
            //mijenjati BROJLO samo kada je neophodno!
            BROJLO = UČITANO = us_u_otk(za_koliko)
        }
        //inače već je u UČITANO vrijednost BMAX
    }
}
  
```