

MIKROKONTROLER AT89C51RC2

tajmer 0 i tajmer 1



Tajmeri

AT89C51RC2 ima tri tajmera koji mogu da se kontrolišu, podešavaju i čitaju posebno.

Tajmeri 0 i 1 su dio osnovne arhitekture 8051 i mogu se koristiti za tri različite funkcije:

- Brojanje vremena i/ili računanje razmaka između dva događaja
- Brojanje događaja
- Generisanje baud rate-a za serijski port (tajmer 1)

Kod tajmera 0 i 1, tajmer uvijek broji na gore, mikrokontroler inkrementira njegovu vrijednost, bez obzira da li se koristi za brojanje vremena, događaja ili generisanje baud rate-a.

Radom tajmera 0 i 1 se upravlja pomoću sljedećih SFR:

Ime SFR	Opis	Adresa SFR
TH0	Viši bajt timera 0	8Ch
TL0	Niži bajt timera 0	8Ah
TH1	Viši bajt timera 1	8Dh
TL1	Niži bajt timera 1	8Bh
TCON	Upravljanje timerima 0 i 1	88h
TMOD	Određivanje moda rada timera 0 i 1	89h



TMOD

TMOD SFR se koristi da bi se kontrolisao mod rada tajmera 0 i 1. Svaki bit ovog SFR daje mikrokontroleru specifičnu informaciju o tome kako da upravlja radom tajmera. Viša 4 bita (biti 4 do 7) odnose se na Timer 1, a niža 4 bita (biti 0 do 3) obavljaju istu funkciju, ali za Timer 0

TMOD (89h) SFR:

Bit	Ime	Objašnjenje funkcije	Timer
7	GATE1	Kada je ovaj bit na 1 timer će da radi samo kada je signal na INT1# (P3.3) na visokom naponskom nivou. Kada je ovaj bit na 0, timer će da radi bez obzira na stanje INT1#.	1
6	C/T1#	Kada je ovaj bit na 1, timer će da broji događaje na pinu T1 (P3.5). Kada je ovaj bit na 0, timer će da se inkrementira u svakom mašinskom ciklusu.	1
5	T1M1	Bit za izbor moda rada timera (sljedeći slajd)	1
4	T1M0	Bit za izbor moda rada timera (sljedeći slajd)	1
3	GATE0	Kada je ovaj bit na 1 timer će da radi samo kada je signal na INT0# (P3.2) na visokom naponskom nivou. Kada je ovaj bit na 0, timer će da radi bez obzira na stanje INT0#.	0
2	C/T0#	Kada je ovaj bit na 1, timer će da broji događaje na pinu T0 (P3.4). Kada je ovaj bit na 0, timer će da se inkrementira u svakom mašinskom ciklusu.	0
1	T0M1	Bit za izbor moda rada timera (sljedeći slajd)	0
0	T0M0	Bit za izbor moda rada timera (sljedeći slajd)	0



Izbor moda rada i upravljanje timerima 0 i 1

Izbor moda rada timera:

TxM1	TxM0	Timer Mode	Opis moda
0	0	0	13-bit Timer.
0	1	1	16-bit Timer
1	0	2	8-bit auto-reload
1	1	3	Split timer mode

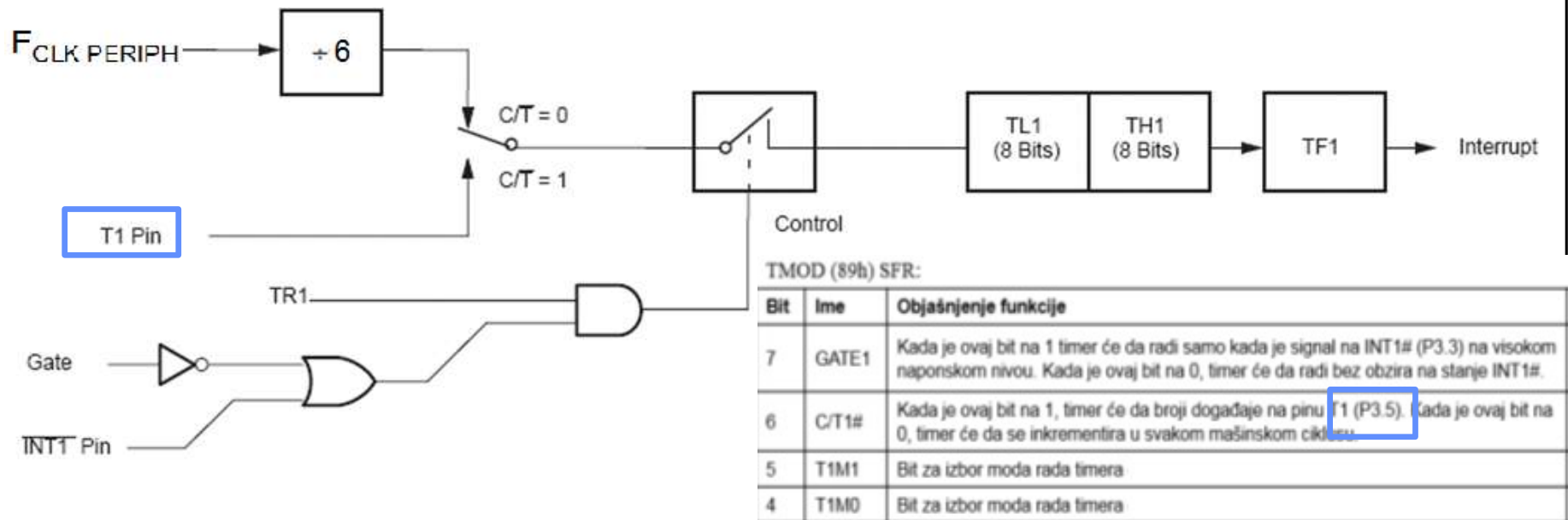
TCON SFR – registar preko koga se upravlja radom timera 0 i 1, a takodje i dobijaju informacije o njihovom radu.

Biti SFR registra TCON (88h) koji se odnose na rad tajmera 0 i 1:

Bit	Ime	Bit adresa	Objašnjenje funkcije	Timer
7	TF1	8Fh	Timer 1 Overflow. Ovaj bit se postavlja na 1 kada timer 1 pređe najvišu vrijednost koja u njega može da bude upisana.	1
6	TR1	8Eh	Timer 1 Run. Kada je ovaj bit setovan, timer 1 je uključen. Kada je ovaj bit na 0, timer 1 je isključen.	1
5	TF0	8Dh	Timer 0 Overflow. Ovaj bit se postavlja na 1 kada timer 0 pređe najvišu vrijednost koja u njega može da bude upisana.	0
4	TR0	8Ch	Timer 0 Run. Kada je ovaj bit setovan, timer 0 je uključen. Kada je ovaj bit na 0, timer 0 je isključen	0



Princip rada tajmera1 u modu 1



Ulaz za brojanje je povezan na tajmer kada je $TR1 = 1$ i ($GATE = 0$ ili $INT1\# = 1$). (Postavljanje bita $GATE = 1$ omogućava da se tajmer kontroliše pomoću externog ulaza $INT1\#$ da bi se omogućilo mjerenje dužine trajanja impulsa).

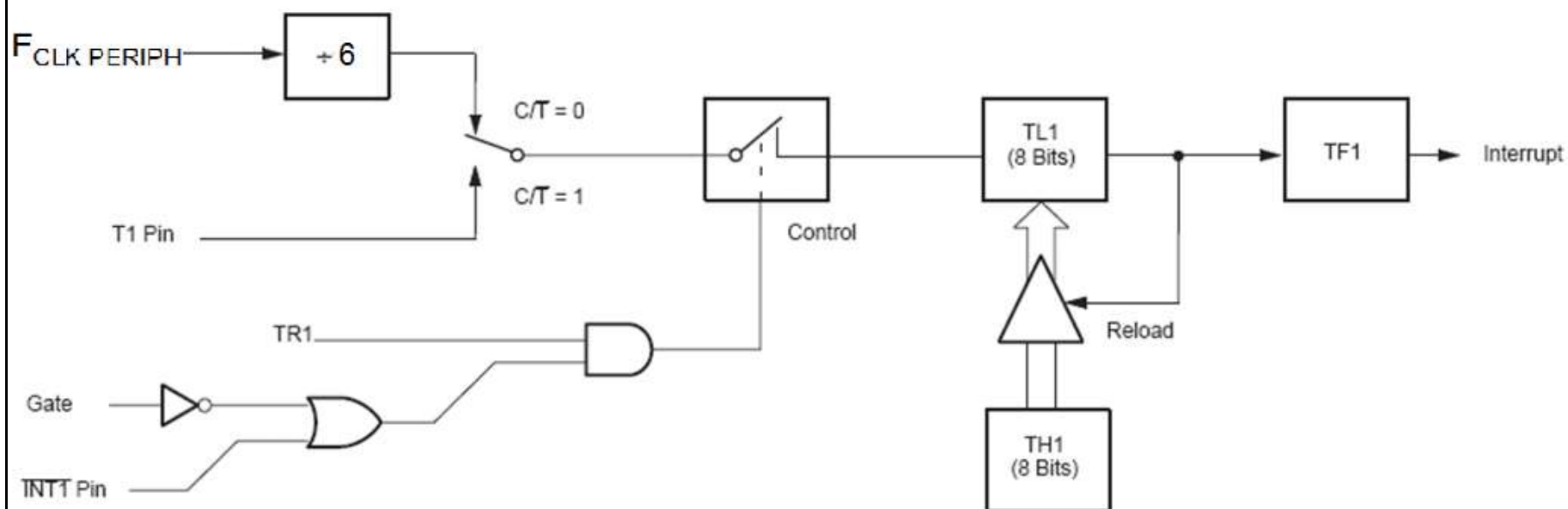
$TR1$ je kontrolni bit u SFR registru TCON. $GATE$ se nalazi u TMOD registru.

U ovom modu, registar timera je konfigurisan kao 16 bitni registar. Ovaj 16-bitni registar sastoji se od 8 bita $TH1$ i 8 bita $TL1$.

Kada je dozvoljeno brojanje, vrijednost u registru se inkrementira. Kada broj u registru pređe sa svih 1 na sve 0, setuje se tajmerov interrupt flag $TF1$. Ukoliko je omogućen prekid tajmera 1 i postoji globalna dozvola prekida, to će dovesti do generisanja prekida tajmera 1 i pozivanja procedure za njegovu obradu.

Mod 0 je isti kao mod 1, s tim što se ne koristi svih 8 bita registra $TL1$, nego samo viših 5 bita, tako da imamo 13-bitni registar tajmera.

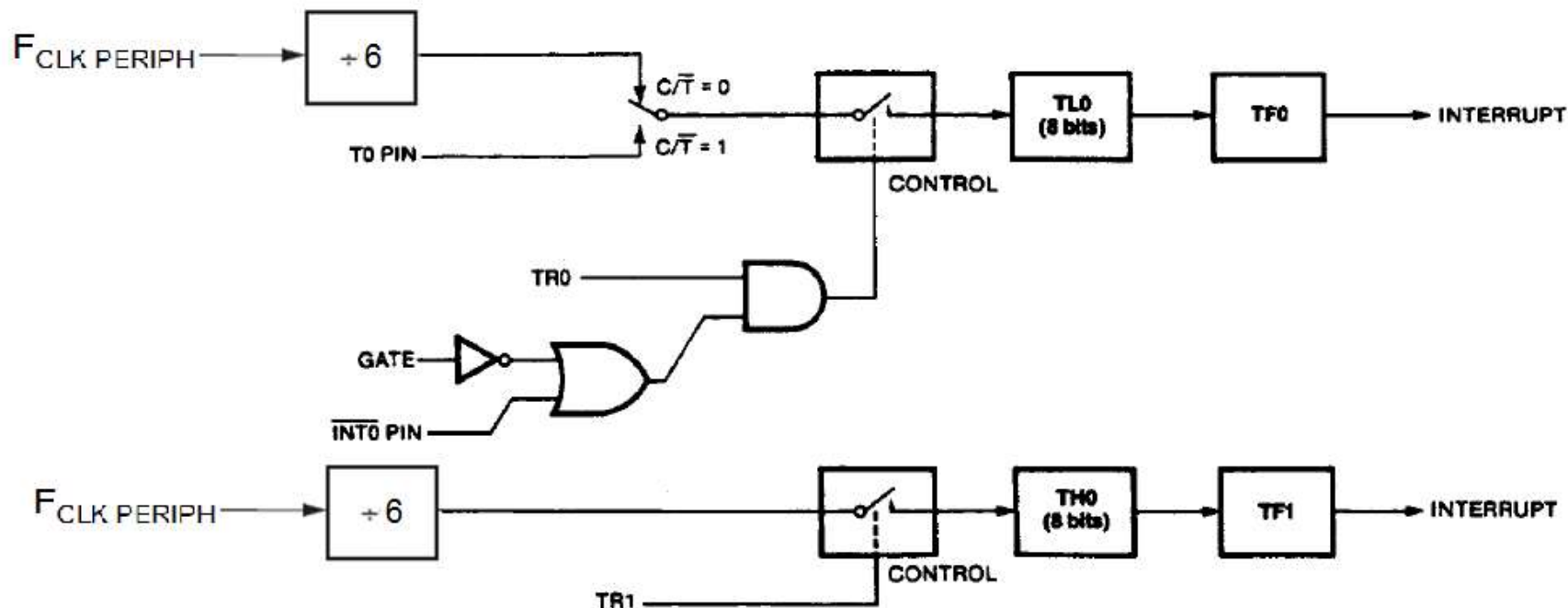
Princip rada tajmera 1 u modu 2



Mod 2 konfiguriše tajmerski registar kao 8-bitni sa automatskim ponovnim upisivanjem (Reload) kao na slici. Overflow (prelazak sa svih 1 na sve 0) vrijednosti u registru $TL1$ ne samo što setuje flag $TF1$, nego i automatski upisuje u $TL1$ vrijednost iz $TH1$ koja se definiše softverski. Ovo upisivanje vrijednosti u $TL1$ ne utiče na vrijednost upisanu u $TH1$.

Modovi rada 0-2 su isti za tajmere 0 i 1.

Princip rada tajmera 0 i 1 u modu 3



Tajmer 1 u modu 3 ne inkrementira registar, efekat je isti kao kada je postavljeno $TR1=0$. Tajmer 0 u modu 3 posmatra $TL0$ i $TH0$ kao dva posebna brojača. $TL0$ se konfiguriše pomoću flagova za tajmer 0 i može da radi kao tajmer ili kao brojač. $TH0$ registar se može koristiti samo za funkciju tajmera, a za rad koristi $TR1$ i $TF1$ flag tajmera 1. Kada je tajmer 0 u modu 3, tajmer 1 može da se koristi u bilo kom od modova rada 0-2, ali ne generiše prekid, tako da može da se koristi samo u aplikacijama koje ne zahtjevaju prekid (npr. baud rate generator).

Zadatak

Napisati program koji broji vrijeme proteklo od uključivanja kontrolera i smješta ga u četiri promjenljive koje definišu broj sati, broj minuta u tekućem satu, broj sekundi u tekućem minutu i broj stotinki u tekućoj sekundi. Oscilator koji se koristi za clock mikrokontrolera ima frekvenciju 24MHz. Koristiti tajmer 0 u modu 2.

Rješenje:

$24\text{MHz}/12=2\text{MHz}$,

trajanje mašinskog ciklusa je $1/2\text{MHz}=0.5\ \mu\text{s}$

Ako stavimo reload vrijednost od 56, tajmer će da odbroji od 56 do 256, odnosno 200 mašinskih ciklusa i onda generisati prekid.

Znači da će prekid biti generisan svakih $200*0.5\ \mu\text{s} = 100\ \mu\text{s}$.

Da bi dozvolili prekid treba da postavimo globalnu dozvolu prekida EA na 1 i dozvolu prekida tajmera 0 ET0 na 1.

Da bi tajmer mogao da broji treba postaviti TR0 bit na 1, kao i GATE bit na 0 da brojanje tajmera ne bi zavisilo od ulaza INT0.

Da bi tajmer vršio brojanje mašinskih ciklusa, a ne externih događaja potrebno je postaviti bit C/T0 na logičku 0.



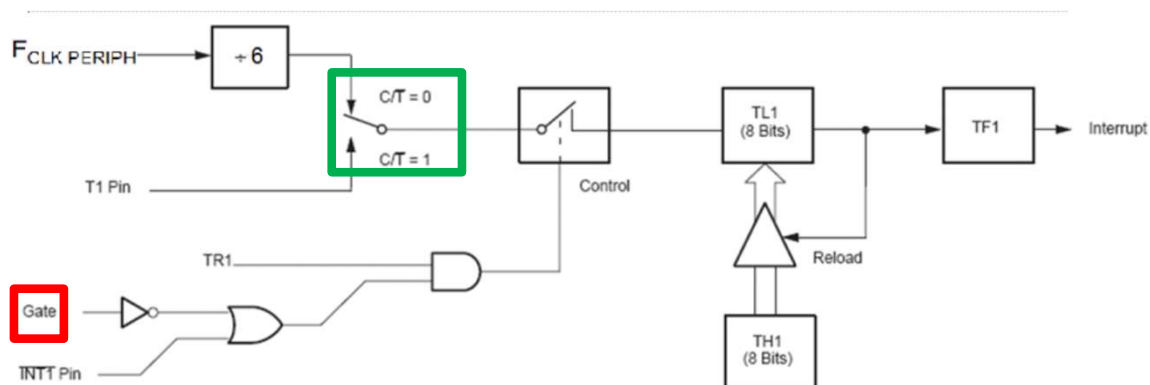
Programski kod rješenja 1/2

```
#include<REG51RC2.h>
```

```
unsigned char data brojStotinki=0;
unsigned char data brojSekundi=0;
unsigned char data brojMinuta=0;
unsigned int data brojSati=0;
unsigned char data brojPrekida=0;
```

```
void InicijalizacijaKontrolera(void)
```

```
{
EA=0; //UKIDANJE GLOBALNE DOZVOLA PREKIDA
TL0=56; //VRIJEDNOST ZA PRVO BROJANJE TAJMERA
TH0=56; //RELOAD VRIJEDNOST
TMOD=0x02; //GATE0=0(bit 3), C/T0=0 (bit 2) mod 2(biti 0-1)
TR0=1; //DOZVOLA RADA TAJMERA 0
ET0=1; //DOZVOLA PREKIDA TAJMERA 0
EA=1; //GLOBALNA DOZVOLA PREKIDA
}
```



3	GATE0	Kada je ovaj bit na 1 timer će da radi samo kada je signal na INT0# (P3.2) na visokom naponskom nivou. Kada je ovaj bit na 0, timer će da radi bez obzira na stanje INT0#.	0
2	C/T0#	Kada je ovaj bit na 1, timer će da broji događaje na pinu T0 (P3.4). Kada je ovaj bit na 0, timer će da se inkrementira u svakom mašinskom ciklusu.	0
1	T0M1	Bit za izbor moda rada timera (sljedeći slajd)	0
0	T0M0	Bit za izbor moda rada timera (sljedeći slajd)	0

Programski kod rješenja 2/2

```
//PROCEDURA ZA OBRADU PREKIDA TAJMERA 0,  
//PREKID SE GENERISE SVAKIH 100us  
//ZA IZVODJENJE OPERACIJA KORISTI SE REGISTAR BANKA 0  
void Tajmer0 (void) interrupt 1 using 0 {  
    if (++brojPrekida == 100) { // odbrojao 100*100us=10ms  
        brojPrekida=0;//resetuje se brojac prekida  
        if(++brojStotinki == 100) { //odbrojao sekundu  
            brojStotinki=0;//postavlja se na 0 odbrojanih stotinki tekuće sekunde  
            if(++brojSekundi == 60) { // odbrojao minut  
                brojSekundi=0;//postavlja se na 0 odbrojanih sekundi tekuće minute  
                if(++brojMinuta == 60) { //odbrojao sat  
                    brojMinuta=0;//postavlja se na 0 odbrojanih minuta tekućeg sata  
                    brojSati++; //inkrementira se broj sati  
                }  
            }  
        }  
    }  
}  
  
void main(void)  
{  
    InicijalizacijaKontrolera();  
    while(1){  
    }  
}
```

