

---

## Problemi di Frequenza

Posted by onibaka - 2008/03/11 15:06

---

Salve a tutti!

E' la prima volta che posto in questo forum, anche se sono mesi che lavoro con la Flex.

Ora vi illustro la mia situazione: in laboratorio ho una Flex Light programmata in modo tale che mi fornisca in uscita dei segnali PWM a 50Hz. Ora sto lavorando con una Flex Full, alla quale ho configurato i jumper come alla prima è l'ho programmata con lo stesso file elf, ma in uscita mi fornisce un'onda a 120Hz.

Sapreste dirmi quale potrebbe essere il problema?

Grazie per l'attenzione.

---

## Re:Problemi di Frequenza

Posted by paolo.gai - 2008/03/11 15:15

---

Ciao!

Intendi dire che lo stesso ELF file programmato sulle due board da un risultato diverso? Mi sembra abbastanza strano, dal momento che i chip dsPIC utilizzati sulle due schede sono identici. per caso hai delle schede di prototipazione con il chip versione GP e non MC?

Ciao,

Paolo

---

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/11 21:21

---

Giusto, è così. La vecchia Light monta un GP mentre la nuova un MC. Cosa dovrei fare?

---

## Re:Problemi di Frequenza

Posted by paolo.gai - 2008/03/11 21:41

---

Sì, mi ricordavo che l'esapode nella versione iniziale usava i primi prototipi della Flex...

Da quello che so i due chip dovrebbero essere uguali a parte qualche periferica. In ogni caso dovrebbe essere impossibile programmare i due chip con lo stesso binario... come hai fatto? l'ICD2 se non sbaglio dovrebbe protestare... se programmi un micro diverso...

per cui ti consiglio di:

- ricompilare il binario per la nuova scheda, modificando la MCU nel file OIL

- verificare le temporizzazioni... prendi come base una programmazione del PLL a 40 Mhz (vedi ad esempio gli esempi EDF che programmano il PLL a 2 o 40 MHz, sono citati anche nel manuale PIC30 nella sezione EDF) e partendo da quelli deriva la programmazione del PWM...

ciao,

PJ

---

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/14 12:59

---

Ho modificato la MCU nel conf.oil, e a ricontrrollare la sezione dei timer non mi sembra esserci incorrettezze. Ho provato

a riprogrammare a 40 MHz, come da manuale, ma le frequenze continuano a non tornare.  
Riporto di seguito la sezione dei timer e quella del conf:

```
void configureTimers(void)
{
    T2CON = 0x0100; //0000000100000000 @ 16bit

    PR2 = 0xFFFE;
    IEC0bits.T2IE = 0;

    T1CON = 0x0100; //0000000100000000

    TMR1 = 0;

    //setto il perio register di timer 1
    //voglio avere 2500 interrupt al secondo in modo da settare il relAlarm //del
    //task motori con 50 in modo che mi generi 50 allarmi al secondo
    // 80 MHz
    // PR1 = 0x3E80;
    // 60 MHz
    PR1 = 0x2EE0;

    IPC0bits.T1IP = 5;
    IFS0bits.T1IF = 0;
    IEC0bits.T1IE = 1;
    T1CONbits.TON = 1;
}

void configureClock(void){
    /* Clock setup */
    CLKDIVbits.DOZEN = 0;
    CLKDIVbits.PLLPRE = 0;
    CLKDIVbits.PLLPOST = 0;
    PLLFBDbits.PLLDIV = 58; //multiply for 60 (4 / 2 * 60 / 2 = 60MHz)
    // PLLFBDbits.PLLDIV = 78; //multiply for 80 (4 / 2 * 80 / 2 = 80MHz)

    /* Wait for PLL to lock */
    while(OSCCONbits.LOCK!=1);
}

CPU mySystem {

    OS myOs {
        EE_OPT = "DEBUG";

        CPU_DATA = PIC30 {
            APP_SRC = "code.c";
            MULTI_STACK = FALSE;
            ICD2 = TRUE;
        };

        MCU_DATA = PIC30 {
            MODEL = PIC33FJ256MC710;
        };

        BOARD_DATA = EE_FLEX {
            USELEDS = FALSE;
        };

        KERNEL_TYPE = FP;

    };
}
```

```

TASK PWM {
  PRIORITY = 1;
  STACK = SHARED;
  SCHEDULE = FULL;
};
TASK calculate {
  PRIORITY = 2;
  STACK = SHARED;
  SCHEDULE = FULL;
};

COUNTER myCounter;

ALARM alarmPWM {
  COUNTER = "myCounter";
  ACTION = ACTIVATETASK { TASK = "PWM"; };
};

};

```

Nel main infine viene aggiunto "SetRelAlarm(alarmPWM, 10, 50);" per iniziare gli interrupt.

Volevo sapere quindi se dal prototipo alla versione definitiva non ci siano stati dei cambiamenti nel range delle frequenze o nel codice esecutivo.  
Grazie per l'attenzione.

---

## Re:Problemi di Frequenza

Posted by paolo.gai - 2008/03/14 14:59

Ciao,

No, non ci sono stati cambiamenti nel codice.

Prova a ripartire dal demo più semplice, quello con il task periodico. su quello la frequenza è sicuramente corretta. Inoltre fai attenzione al fatto che la frequenza va specificata in 2 punti, la prima all'inizio del main con quel ciclo infinito, la seconda all'inizio del file per settare il valore iniziale di alcuni registri.

In ogni caso ERIKA non fa che utilizzare i settaggi standard di Microchip, per cui puoi utilizzare i settaggi di default di Microchip per impostare la frequenza...

Ciao

Paolo

---

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/16 22:23

Vorrei porre un ultimo problema: quando si imposta il periodo di un timer non riesco a capire che tipo di calcolo si utilizza per far sì che si attivi nel momento desiderato. Su un esempio del manuale si imposta il periodo di timer1 a 1 ms avendo una Fosc=20Mhz, che diventa quindi 5000 (decimale).

```

/* Timer1 period for 1 ms with FOSC = 20 MHz */
#define TMR1_PERIOD 0x1388"

```

Come si passa da 20MHz a 1388?

Grazie

---

=====

## Re:Problemi di Frequenza

Posted by paolo.gai - 2008/03/17 09:11

---

Su un esempio del manuale si imposta il periodo di timer1 a 1 ms avendo una Fosc=20Mhz, che diventa quindi 5000 (decimale).

Come si passa da 20MHz a 1388?

(non ho trovato questo esempio all'interno dei manuali di ERIKA... a che pagina si trova?)

5000 decimale = 0x1388 (esadecimale)

(per fare la prova puoi utilizzare una calcolatrice (tipo quella di linux o windows) in modalità scientifica.

con 40MHz, ad esempio, hai un clock a 25ns, e servono 4000 cicli per un timer ad 1ms , che corrisponde a 0x0fa0

ciao,

Paolo

---

---

=====

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/17 09:50

---

è un esempio che ho trovato a pagina 82 del "MPLAB® C30 C COMPILER USER'S GUIDE" che riguarda i timer

---

---

=====

## Re:Problemi di Frequenza

Posted by paolo.gai - 2008/03/17 10:20

---

Ok... (in effetti non mi ricordavo di aver mai pubblicato una temporizzazione a 20MHz per la Flex...)

PJ

---

---

=====

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/18 18:15

---

Eccomi di ritorno. Il problema della frequenza dei task e dei timer è stato risolto. Siamo partiti da un esempio per il controllo di un led e abbiamo portato il task a ripetersi ogni 20 ms. Da questa base abbiamo iniziato ad aggiungere volta volta il codice per il controllo dei motori in PWM, ma appena abbiamo inserito il codice che segue nel task i valori di frequenza d'uscita non erano più gli stessi.

```
/*Questa è l'inizializzazione*/
void setPortsAddress(void)
{
  portsAddress=(unsigned int *)&LATA;
  portsAddress=(unsigned int *)&LATB;
  portsAddress=(unsigned int *)&LATC;
  portsAddress=(unsigned int *)&LATD;
  portsAddress=(unsigned int *)&LATE;
  portsAddress=(unsigned int *)&LATF;
  portsAddress=(unsigned int *)&LATG;
```

```
}  
  
void setDCMatrix(void){  
  //FR  
  DCMAtrix = 6; DCMAtrix = 0b1111111111111110; //RG0  
  DCMAtrix = 0; DCMAtrix = 0b1111111101111111; //RA7  
  DCMAtrix = 4; DCMAtrix = 0b1111111111111101; //RE1  
  //CR  
  DCMAtrix = 6; DCMAtrix = 0b1110111111111111; //RG12  
  DCMAtrix = 4; DCMAtrix = 0b1111111111111011; //RE2  
  DCMAtrix = 4; DCMAtrix = 0b1111111111110111; //RE4  
  //BR  
  
  DCMAtrix = 4; DCMAtrix = 0b1111111111011111; //RE5  
  DCMAtrix = 4; DCMAtrix = 0b1111111101111111; //RE7  
  DCMAtrix = 2; DCMAtrix = 0b1111111111111101; //RC1  
  //BL  
  DCMAtrix = 3; DCMAtrix = 0b1111111111011111; //RD5  
  DCMAtrix = 3; DCMAtrix = 0b1111111101111111; //RD7  
  DCMAtrix = 5; DCMAtrix = 0b1111111111111101; //RF1  
  //CL  
  DCMAtrix = 3; DCMAtrix = 0b1111111111111101; //RD1  
  DCMAtrix = 3; DCMAtrix = 0b1111111111111011; //RD3  
  DCMAtrix = 3; DCMAtrix = 0b1101111111111111; //RD13  
  //FL  
  DCMAtrix = 0; DCMAtrix = 0b0111111111111111; //RA15  
  DCMAtrix = 3; DCMAtrix = 0b1111110111111111; //RD9  
  DCMAtrix = 3; DCMAtrix = 0b1111011111111111; //RD11
```

---

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/18 18:16

Questo è il codice che da errore nel task modificando la frequenza di uscita

```
(*portsAddress)=(*portsAddress)&(DCMAtrix);
```

Abbiamo utilizzato una diversa implementazione ma più "dispendiosa". Vorrei sapere se sbagliamo qualcosa nell'utilizzo degli indirizzi porte e delle maschere.

Grazie

---

## Re:Problemi di Frequenza

Posted by paolo.gai - 2008/03/18 18:39

Se ho capito bene, la situazione è:

- il led è acceso da un task
- il task è attivato ogni 20 ms
- aggiungendo del codice al task il led non si accende più periodicamente

Allora una possibile causa potrebbe essere il task che dura più dei 20ms.

Inoltre, installate la patch del kernel che abbiamo messo nella pagina download, che risolve un problema riscontrato nella gestione degli interrupt (potrebbe essere la causa del ritardo).

Ciao,

---

Paolo

=====

## Re:Problemi di Frequenza

Posted by onibaka - 2008/03/18 18:51

---

Quello che succede è:

- il task si avvia ogni 20ms
- ogni volta che si avvia il task accende o spegne il led (come da esempio)ed è stata controllata la frequenza con un oscilloscopio e risulta corretta
- inserito il codice per il controllo delle uscite tramite indirizzi e maschere il led rimane acceso sempre (almeno ad occhio sembrerebbe, come se si attivasse a una frequenza molto più alta dei 50 Hz) e l'oscillazione d'uscita si aggira sui 260 Hz circa.

Abbiamo fatto diverse prove levando e rimettendo il codice, e le inizializzazioni non danno problemi durante la compilazioni. Se la situazione si risolve tramite la patch allora non ci dovrebbero essere più problemi.

=====