```
/**************************************************
  Simple 1MHz counter  INTERNAL OSC
   1602 display by 4040B counter with PIC16F628A
             By nobcha all right reserved

  Ver 1.1 09/12/2010
  Hitech C & MPLAB

   PIC16F88 + LCD via TC4040B
   PIN Assign   #3 RA4:counter input
                #6 RB0:pulse out for 4040
                #7 RB1:EN

   TMR0 is counter with 1/2 prescaler
   TMR1 is gate time controller as set 10ms(65536-10000)


   4040B pin
                #10 clock in
                #11 rst (70uS delayed #10 high state )
                #9  seg-a
                #7  seg-b
                #5  seg-c
                #4  seg-d
                #6  seg-e
                #13 seg-f
                #12 seg-g
                #14 seg-dp

   OSC  INTERNAL 4MHz

 Development Circumstance
  MPLAB IDE V8.56   HiTECH C V9.71a

 Counter data is put on to RA4 which is TMR0 input.
 TMR1 is worked for 10mS gate. TMR0 is 8 bits counter
 It is counted TMR0 overflow on count_1.
**************************************************/

#define pic_clk 4000000

#define MHz 000000
#define _XTAL_FREQ 4MHz


#include <htc.h>
#include "delay.h"
#include <pic.h>
#include <stdio.h>
#include "628_lcd.h"

__CONFIG(BORDIS & UNPROTECT & PWRTEN & WDTDIS & LVPDIS & MCLREN & INTIO  );


unsigned char timeup ;
short read_data, count_l;

void cnt_setup(void){
        TRISA = 0b10000000;             // RA4 disable
        TMR0 = 0 ;                      // TMR0 clear
        TMR1L = 0;                      // Clear Low Byte of TMR1
        TMR1H = 216;                    // Set 216*256=55296
        TMR1L = 240;                    // Set (240) =55536=65536-10000
                                        // This is 10mS gate time settinng
        T0IF=0;                         // TMR0 flag off
        TMR1IF=0;                       // TMR1 flag off
        TMR1IE=1;                       // TMR1 INT ENABLE
        T0IE=1;                         // TMR0 INT ENABLE


        timeup=0;                       // Reset timeup flag
        count_l=0;                      // Reset overtime flag

        INTCON=0b11100000;      // GIE 1,PEIE 1,TMR0IE 1,INT0IE 0,RBIE 0,TMR0IF 0,INT0IF 0,RBIF 0
```

```
        T1CON = 0b00000001;     // T1RUN 1,T1CKPS 00,T1OSCDIS,T1SYNC 1,TMR1CS 0,TMR1ON 1
        TRISA = 0b10010000;     // RA4 enable
        }

void interrupt cnt_int(void){
        GIE=0;
        if(T0IF){
                count_l++;              // Counter0 overflow occured
                T0IF=0;
                GIE=1;
                }
        if(TMR1IF){
                TRISA = 0b10000000;     //  Stop count RA4 disable
                timeup=1;               // Gate time over
                TMR1IF =0;
                }

        }

void main(){
        unsigned char i, zero_sup,disp_data ;
        short decimal;

        for(i=0;i<50;i++){
        __delay_ms(10);
        }

        TRISA = 0b10110000;     // RA4,5,7 input the others output
        CMCON = 0b00000111;     // No Compalator
        TRISB = 0b00000000;     // No input

        PORTA = 0b00000000;     // PORTA clear
        PORTB = 0b00000000;     // PORTB CLEAR

        OPTION=0b10110000;      // PORTB pullup,INTEDG 0,TOCS TOCKI 1,ToSE1,PSA TIMER0,1/4

        GIE=0;                          // INT off

        PCON = 0b00001000;      // 4MHz Internal  4MHz
        T1CON = 0;              // Timer1 off
        INTCON=0b01100000;      // GIE 0,PEIE 1,TMR0IE 1,INT0IE 0,RBIE 0,TMR0IF 0,INT0IF 0,RBIF 0
        PIE1 = 0b00000001;      // ADIE 0,RCIE 0,TXIE 0,SSPIE 0,CCP1IE 0,TMR2IE 0,TMR1IE 1
        PIR1 = 0b00000000;      // ADIF 0,RCIF 0,TXIF 0,SSPIF 0,CCP1IF 0,TMR2IF 0,TMR1IF 1
        T2CON = 0x0;            // No use

        RB4 = 1;
        lcd_init();
        __delay_ms(20);
        lcd_goto(0);            // select first line
        __delay_ms(20);
        lcd_puts("Frequency   Gate ");

        lcd_goto(0x4C);         // select second line
        __delay_ms(2);
        lcd_puts("10mS");
        lcd_goto(0x40);
        RB4=0;

        cnt_setup();            // counter initial set up

while(1){
        GIE=1;
        if(timeup){
                timeup=0;
                GIE=0;

                if(count_l>100){lcd_puts("OF ");}       // more than 64 could not be covered short
                else{lcd_puts("   ");}

                read_data=(TMR0+(char)count_l*256)*2;   // (tmr0+(overflow count_1*256))*prescaler
                i=0;
                decimal = 10000;
                zero_sup=1;                     // Zero suppres flag

                while(i<5){                     // 5digit
```

```
                disp_data = ((char)((read_data/decimal)%10)) | 0x30;
                                        // Get digit data

                if((disp_data==0x30)&zero_sup & i!=3 ){
                        disp_data=0x20; // zero suppres
                        }
                else{ zero_sup=0;}      // zero suppress release
                                        //
                lcd_putch(disp_data );  // Display digit
                decimal = decimal / 10; // 10->1
                i++;
                if(i==4){lcd_putch( 0x2e);      // Display DP
                }
                }
                lcd_puts("kHz");        //

                lcd_goto(0x40);                 // select second line

        for(i=0;i<50;i++){
        __delay_ms(10);
        }
        RB4=RB4+1;

        cnt_setup();                    // Gate open Counter start
        }
    }
}
```