```
/*
============================================================================
=========
 Name        : display_capture.c

 Author      : Concepcion Alvarado Erik
 Version     :
 Copyright   : $(copyright)
 Description : main definition
============================================================================
=========
*/

#define AddrFIO0DIR      0x2009C000
#define AddrFIO0SET      0x2009C018
#define AddrFIO0CLR      0x2009C01C

//-- To P1[26] CAP0.0
#define AddrFIO1DIR     0x2009C020
#define AddrFIO1PIN     0x2009C034

//-- To Display P2[]
#define AddrFIO2DIR     0X2009C040
#define AddrFIO2SET     0X2009C058
#define AddrFIO2CLR     0x2009C05C
#define AddrFIO2PIN     0x2009C054

#define AddrPINSEL0     0X4002C000
#define AddrPINSEL3     0X4002C00C

#define AddrPCONP               0X400FC0C4
#define AddrPCLKSEL0    0X400FC1A8
//-------TIMER0----------
#define AddrT0EMR               0X4000403C
#define AddrT0MCR               0X40004014
#define AddrT0MR0               0X40004018
#define AddrT0TCR               0X40004004
#define AddrT0IR                0X40004000
#define AddrT0PR                0X4000400C
#define AddrT0CCR               0X40004028
#define AddrT0CR0               0X4000402C
//------TIMER1-----------
#define AddrT1EMR               0X4000803C
#define AddrT1MCR               0X40008014
#define AddrT1MR0               0X40008018
#define AddrISER0               0XE000E100
#define AddrT1TCR               0X40008004
#define AddrT1IR                0X40008000
#define AddrT1PR                0X4000800C

unsigned int volatile *const FIO1DIR = (unsigned int*) AddrFIO1DIR ;
unsigned int volatile *const FIO1PIN = (unsigned int*) AddrFIO1PIN ;

unsigned int volatile *const FIO2DIR  = (unsigned int*) AddrFIO2DIR;
unsigned int volatile *const FIO2SET  = (unsigned int*) AddrFIO2SET;
unsigned int volatile *const FIO2CLR  = (unsigned int*) AddrFIO2CLR;
```

```
unsigned int volatile *const FIO2PIN  = (unsigned int*) AddrFIO2PIN;

unsigned int volatile *const PINSEL0  = (unsigned int*) AddrPINSEL0;

unsigned int volatile *const FIO0DIR           = (unsigned int*)
AddrFIO0DIR;
unsigned int volatile *const FIO0SET           = (unsigned int*)
AddrFIO0SET;
unsigned int volatile *const FIO0CLR           = (unsigned int*)
AddrFIO0CLR;

unsigned int volatile *const PCONP    = (unsigned int*) AddrPCONP;
unsigned int volatile *const PCLKSEL0 = (unsigned int*) AddrPCLKSEL0;
unsigned int volatile *const PINSEL3  = (unsigned int*) AddrPINSEL3;
// to P1[26]-> Modo Capture "CAP0.0"

unsigned int volatile *const T0EMR    = (unsigned int*) AddrT0EMR;
unsigned int volatile *const T0MCR    = (unsigned int*) AddrT0MCR;
unsigned int volatile *const T0MR0    = (unsigned int*) AddrT0MR0;
unsigned int volatile *const T0TCR    = (unsigned int*) AddrT0TCR;
unsigned int volatile *const T0IR     = (unsigned int*) AddrT0IR;
unsigned int volatile *const T0PR     = (unsigned int*) AddrT0PR;
unsigned int volatile *const T0CCR    = (unsigned int*) AddrT0CCR;
unsigned int volatile *const T0CR0    = (unsigned int*) AddrT0CR0;

unsigned int volatile *const T1EMR    = (unsigned int*) AddrT1EMR;
unsigned int volatile *const T1MCR    = (unsigned int*) AddrT1MCR;
unsigned int volatile *const T1MR0    = (unsigned int*) AddrT1MR0;
unsigned int volatile *const ISER0    = (unsigned int*) AddrISER0;
unsigned int volatile *const T1TCR    = (unsigned int*) AddrT1TCR;
unsigned int volatile *const T1IR     = (unsigned int*) AddrT1IR;
unsigned int volatile *const T1PR     = (unsigned int*) AddrT1PR;

unsigned int cuenta;
unsigned int decena,unidad;

void config_GPIO(void);
void config_TIMER0(void);
void config_TIMER1(void);
void display(unsigned int);
int debounce(int);

void TIMER1_IRQHandler(void);
void TIMER0_IRQHandler(void);

// a[6],b[5],c[4],d[3],e[2],f[1],g[0]
//
const char dec_to_7seg_cc[10]={  //abcdefg
                0b1111110, //0
                0b0110000, //1
                0b1101101, //2
                0b1111001, //3
                0b0110011, //4
                0b1011011, //5
                0b1011111, //6
                0b1110000, //7
                0b1111111, //8
```

```
                0b1110011  //9
               };


////////////////////////////////////////////////////////////////////////
///////



/* ------------------------------------------
 * @Output:
 *  P2[6][5][4][3][2][1][0] : En ése orden!
 *      "g,f,e,d,c,b,a" Display cátodo común
 *      pin48 al pin42 de la placa (respetando el orden)
 *  P2[7][8] : Enable of Unidad and Decena
 *      pin 49,50
 *      ------------------------------------
 * @Input: Timer0 Capture "CAP0.0" P1[26] (Pad9 de la placa)
 */

//-- Init Output



#define Port2_Pin(x)  x
#define LED         (1 << 22)
#define LED_ON      *FIO0SET |= LED
#define LED_OFF     *FIO0CLR |= LED
#define PULL_UP 1 //enable!
#define PULL_DOWN !PULL_UP
/*-------------------------------------------------
          CONFIGURACION
-------------------------------------------------*/

//-- Config GPIO
void config_GPIO(void){
  //--FIO2DIR (Select Input/Output)
  //-- Config Display 7Seg comun cathode as Output:
  // Output (Set 1):
  *FIO2DIR |= (
          (1<<Port2_Pin(0)) |
          (1<<Port2_Pin(1)) |
          (1<<Port2_Pin(2)) |
          (1<<Port2_Pin(3)) |
          (1<<Port2_Pin(4)) |
          (1<<Port2_Pin(5)) |
          (1<<Port2_Pin(6)) |
          (1<<Port2_Pin(7)) |
          (1<<Port2_Pin(8))
        );
  *FIO0DIR |= LED ;
}

//-- Config Timer0
void config_TIMER0(void){
  // Config P1[26], mode Capture-PullUp:
  *FIO1DIR &=~ (1<<26); // P1[26] como entrada
```

```
  *PINSEL3 |= (3<<20);  // modo CAP0.0

  *PCLKSEL0&=~(3<<2);       // (ya estaba en el codigo, va?)
peripheral clock: system clock Timer0 cclk/4

  // CCR: Count Control Register (CTCR): Count Control Register
  *T0CCR|=(3<<1);          // (segun lo q estaba)
  //*T0CCR|=((1<<0)|(1<<1)|(1<<2)); // (segun tutorial)
  *T0TCR|=1;          // habilito el registro del control del timer
             // start time

  *ISER0|=(3<<1);   // Enable Interrupt
}

//-- Config Timer1
void config_TIMER1(void){
  // Config Timer1: Mode Match (Multiplexado 20mseg)
  *PCONP|=(3<<1);   //enciendo el periferico del timer
  *PCLKSEL0|=(3<<4);  //peripheral clock: system clock Timer1 cclk/8

  *T1EMR|=(15<<4);
  *T1EMR|=1;
  *T1EMR&=~(1<<1);
  *T1MCR|=3;      //reset on MR0 the TC will be reset if MR0 matches
it. interrup on MR0: an interrup is
                  //generated when MR0 matches the value in the TC
  *T1MR0=6000;    //match register 0 6450

  *T1TCR|=1;      //habilito el registro del control del timer

  *ISER0|=(1<<2);   // Enable Interrupt

}

/*-----------------------------------------------------
          METODOS
-----------------------------------------------------*/

//-- Display
void display(unsigned int decimal){
  if(decimal<10 )
  {
    *FIO2PIN = dec_to_7seg_cc[decimal];
  }
}

// Antirebote:
int debounce(int SampleA){
  //-- Set static variables:
  static int SampleB=0;
  static int SampleC=0;
  static int LastDebounceResult = 0;

  //-- Logical Function:
  LastDebounceResult = (LastDebounceResult &&
            (SampleA || SampleB || SampleC)) ||
            (SampleA && SampleB && SampleC);
```

```
  //-- Update Sample
  SampleC=SampleB;
  SampleB=SampleA;

  return LastDebounceResult;
}

/*-------------------------------------------------
          SUBRUTINAS
-------------------------------------------------*/

void TIMER1_IRQHandler(void){
    static int i=0;
    if (i==0){
      //Enable decena and display decena
      //*FIO2PIN = dec_to_7seg_cc[decena];
      display(decena);
      *FIO2SET|= (1<<Port2_Pin(8));
      i=!i;
    }
    else if (i==1){
      //Enable unidad and display unidad
      //*FIO2PIN = dec_to_7seg_cc[unidad];
      display(unidad);
      *FIO2SET|= (1<<Port2_Pin(7));
      i=!i;
    }

  // if((*T1EMR&1)==1){
  // //Enable decena and display decena
  //   //*FIO2PIN = dec_to_7seg_cc[decena];
  //   display(decena);
  //   *FIO2SET|= (1<<Port2_Pin(8));
  // }
  // if(((*T1EMR>>1)&1)==1){
  //  //Enable unidad and display unidad
  //   //*FIO2PIN = dec_to_7seg_cc[unidad];
  //   display(unidad);
  //   *FIO2SET|= (1<<Port2_Pin(7));
  // }
  *T1IR|=1;       //MR0 interrupt flag for match channel 0
}

void TIMER0_IRQHandler(void){
  //if(LPC_GPIO1->FIO1PIN & (1<<26))//high?
  /*

  */
  static int Puls,PulsAnt;
  static int alternar = 0;
  //Antirebote
  for(unsigned int j=0; j<3; j++)
  {
    for(int p=0;p<20000;p++);
    if(PULL_UP==1)
      PulsAnt= !( *FIO1PIN & (1<<26) );
    else if (PULL_DOWN==1)
```

```
      PulsAnt= ( *FIO1PIN & (1<<26) );

    Puls=debounce(PulsAnt);
  }

  if(Puls==0 && PulsAnt==0)//high?
  {
    if(alternar==0){
      LED_ON;
      alternar =! alternar;
    }
    else{
      LED_OFF;
      alternar =! alternar;
    }
    cuenta=*T0CR0;
    decena=0;
    unidad=0;
    cuenta=cuenta&127; // lo convierte a un num de 2 byte
    if(cuenta>100){
      cuenta-=100;
    }
    decena= cuenta / 10 ; //20
    unidad= cuenta-decena*10;
  }

  *T0IR|=(1<<4);        // Bajo bandera de CAP0.0
}

/*****************************************************
          MAIN
*****************************************************/
int main(void){
  cuenta=0;
  decena=0;
  unidad=0;
  config_GPIO();
  config_TIMER0();
  config_TIMER1();

  while(1){

  }
  return 0;
}
```