Timer / Counter

Third Release
retriggerable timer

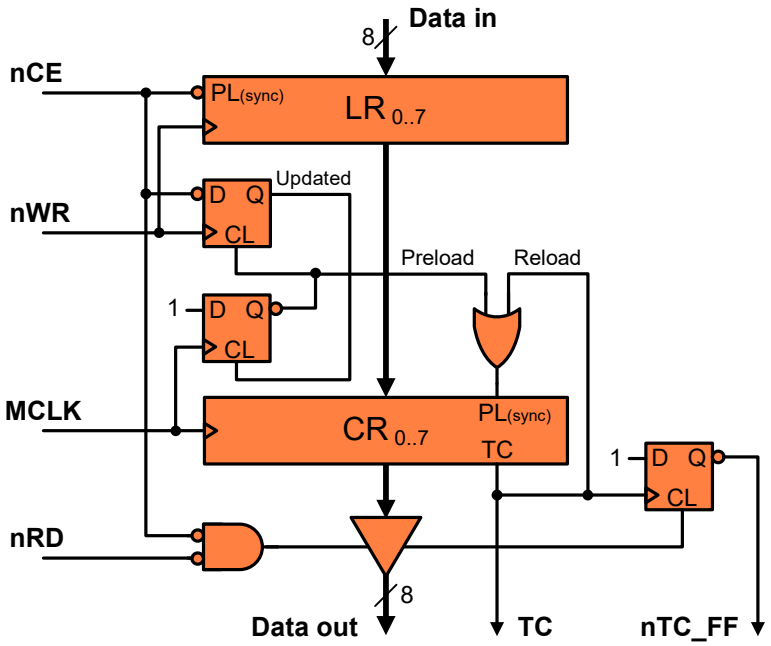
ISEL
INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

SASO_TIMER_v3

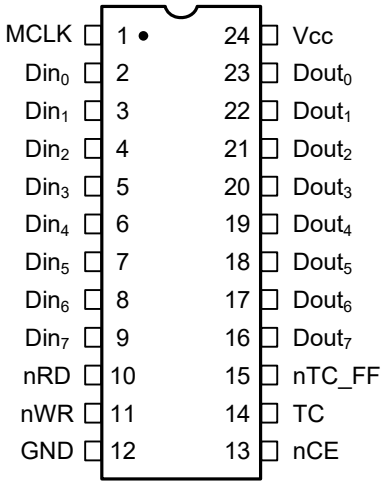
General Description

The SASO_TIMER implements a down counter/timer with 8-bit length. On the rising edge of the *MCLK,* the *Counter Register* (*CR*) value is decremented. When *CR0..7* equals to zero *Terminal Count (TC)* is set active during the second half of MCLK and the flag *nTC_FF* is activated. The *Counter Register* is then reloaded with the value that has been previously loaded into the *Load Register* (*LR*) renewing consecutive operation cycles.
In the present version the *Counter Register* is also loaded on the rising edge of the *MCLK* whenever the *Load Register* is updated.
Flag *nTC_FF* is cleared when a reading of the timer occurs.

Functional Block Diagram



Pin Configuration



SkinnyDIP
Top View

Refer to next page for CUPL implementation

CUPL Implementation

```
Name     TIMER_v3      ;
PartNo   00            ;
Date     24/05/2017    ;
Revision 3.0           ;
Designer PM/HM         ;
Company  CCISEL        ;
Assembly None          ;
Location               ;
Device   v750c         ;

/* ***** INPUT PINS ********* OUTPUT PINS ********/
PIN  1      =  MCLK      ;  PIN  14      =  TC       ;
PIN [2..9] = [Din0..7] ;  PIN  15      = !TC_FF    ;
PIN  10     = !RD        ;  PIN [16..23] = [CR7..0] ;
PIN  11     = !WR                                   ;
PIN  13     = !CE                                   ;
/* ***************** PINNODES ********************/
          PINNODE [27..34] = [LR0..7]              ;
          PINNODE  26      = Updated               ;
          PINNODE  25      = X                     ;

/* ****************** BODY ***********************/

/*            Load Register definition           */
[LR0..7].sp = 'b'0;
[LR0..7].ar = 'b'0;
[LR0..7].ck = !WR;
[LR0..7].d  =  CE & [Din0..7]
            # !CE & [LR0..7] ;

/*          Counter Register definition          */
[CR0..7].ar = 'b'0;
[CR0..7].sp = 'b'0;
[CR0..7].ckmux = MCLK;

      CR0.t = !PL  # (CR0 $ LR0) ;

$REPEAT   i = [1..7]
    CR{i}.t = !PL & ![CR0..{i-1}]:&
            #  PL & (CR{i} $ LR{i}) ;
$REPEND

       zero = ![CR0..7]:& ;

/* TC is an output signal available as glitch free
 since it is constrained to the 2nd half of MCLK */
         TC = zero & !MCLK ;

/*    CR ctrl for autoreloading & retriggering   */
         PL = Preload # Reload ;
     Reload = TC ;
    Preload = !X ;

/*       Terminal Count flip-flop definition     */
   TC_FF.sp = 'b'0;
   TC_FF.d  = 'b'1;
   TC_FF.ck = TC;

/* reading the Timer <=> nTC_FF acknowledgement  */
   RD_Timer = RD & CE ;
[CR0..7].oe = RD_Timer ;
   TC_FF.ar = RD_Timer ;

/* updating of LR detection => ask for Preload   */
 Updated.sp = 'b'0;
 Updated.d  = CE;
 Updated.ck = !WR;
 Updated.ar = Preload ;

/* auxiliar register: Preload requested if X==0  */
    X.sp    = 'b'0;
    X.d     = 'b'1;
    X.ckmux = MCLK;
    X.ar    = Updated ;
```

ISEL
INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

SASO_TIMER_v3