EXPERIMENT 5 - TIMER PROGRAMMING WING MSP432n

LAB TITLE AND COOK : EMBEDDED COMPUTING LAB (MCCE283) EXPERIMENT NUMBER: 5 DATE: M/05/2022, TUESDAY

Control an LED by developing a timer program using system Tick, Timer 32 and Timer A timers that can be Enterfaced with an MSP43x microcontroller.

a) LED CONTROL USING SYSTICK TIMER :

O configure functionality of P2-1 as simple GP10 pins. O configure direction of P2-1 as output for GREEN-LED.

3 Reload register value per generating 1 12 or 1 sec delay: MCIK = 30,00,000 HZ

1 Clear ST Current Value Register.

@ Grable systick to begin counting down and disable interrupt generation. For the clock source, system clock (MCIK) is only in plemented. implemented.

(i) check whether the systic das counted down to zero and of COUNTFLAG is set.

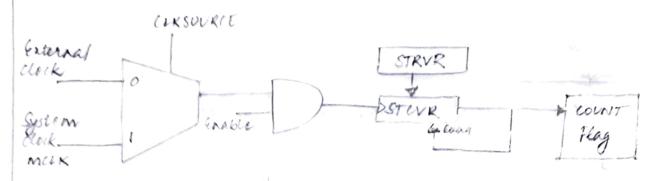
(11) If yes, thigger the GREEN-LED.

C PROGRAM CODE -# include "msp. h"

Il Main Function: int main (void)

> P2 -> SEO L= ~2; Il configure functionality of P2.1 as simple 6110 pins PZ -> SELI

2- Il Configure direction of P2-1 as august for GREEN-1ED



24 Bit Down Counter
Figure 1 - System Tick Timer Internal standard

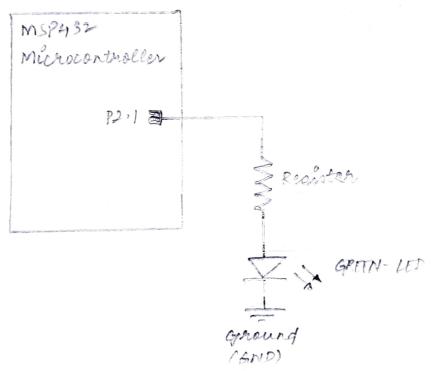


Figure 2 - IED Control using System Tick Timer

```
Systick > 10AD = 3000000-1; If Reload neglister value for generating 1 H2 or 1 sec delay: mc+k = 30,00,000 H2

Systick > CTR1 = 5: If Enables Systick to begin counting down:

Interrupt generation is disabled: System clock MCLK,

Systick > VAL = 0: If Clear ST(when t Value Register

If Infinite Loop (An embedded pragram does not stop):

while (1)

{

If The Systick > CRL & Ox10000) If If COUNTING is set

P2 > OUT 1=2; If Thigger the GRITN-LTD

}
```

b) LED CONTROL USING TIMER 32 TIMER :

- O Configure functionality of P2.1 as simple GP10 pins.
- O configure direction of PD I as autput for GREEN-LED.
- B Reland negister value for generating 1 Hz or 1 sec delay: Assume prescale unit to be equal to 1; ser MCIK: 30,00,000 Hz
- made The counter generates an interrupt at a constant interval, neloading the value from T3210ADn register.
- Disable times interrupt enable bit and set prescale bits to oo clock is divided by 1. Select 32-bit counter aperation.
- Select wrapping made, i.e., the timer continues counting when it reaches to zero.
- (ii) Trigger the GREEN-LED.

PRODUCT SO STOCK PROPERTY OF SOME

Figure 3- Timer 32 Timer Internal structure

Microcontroller

P201

Resister

GRAN-HD

Ground

GRAND

Figure 4- LED CONTROL woing Timer 32 Timer

include "msp. h"

Main Function ; int main (void)

 $P2 \rightarrow ST10 L= ^{12}$ // Configure functionality of $P2 \cdot 1$ as simple GP10 pins $P2 \rightarrow ST21 L= ^{12}$

P2 -> DIR 1 × 2: Il configure d'une chan est P2:1 as autput for GREEN-LED

TIMER 32-1 → LOAD = 3000000 -1: // Reload negister value for generating 1 MZ or 1 sec delay: MCLK > 30,00,000; Prescale unit =1

/* Enables timer to begin counting down; Periodic made;
Disable timer interrupt enable; set prescale bits to 00;
clock is divided by 1; select 32-bit counter aperation;
wrapping made (Timer 1 Timer Control Register)*/
TIMER 32-1→ CONTROL = Ox (2;

Is Infinite Loop (An embedded program does not stop):
while (1)
{

while ([TiMER 32-1 -> RTS & 1) == 0); Il wait until times is completed (Timer 1 Raw Interrupt status Register)

TIMER 32_1 -1 NTCLR = 0: 11 Any write to the T32 INTCLR 1
register clears the interrupt output from the counter.

P2 -> OUT ^=2; Il Trigger the GREEN-LED

3

6) LED CONTROL USING TIMERA :

O Configure functionality of P2.1 as simple GP10 pins.

1 Configure direction of P2.1 as output for GREEN-LED.

3 Set TimerA interrupt flag to 1 (timer overflowed). THIFG will be cleared on writing 1 to this bit. Disable interrupt.

Set made control buts to 01, up made. The timer counts up to TAX CCRO. Set input divider buts to 11, divide by 8. Thex buts select the divider for the input clock.

@ Set clock source select bits to 10, SMCIK (internal clock).

(8) For generating I Hz or I see delay, set CCRTO) = 46875-1. Assume 200 = 8 and TAIDEX = 7. Set system clock (SMCIK) equal to 30,00,000 Hz

(i) Wait until the CCIFG is set.

(ii) Clear interrupt flag.

(iii) Trigger the GRFFN- UTD.

→ C PROGRAM CODE -# include "mep. h"

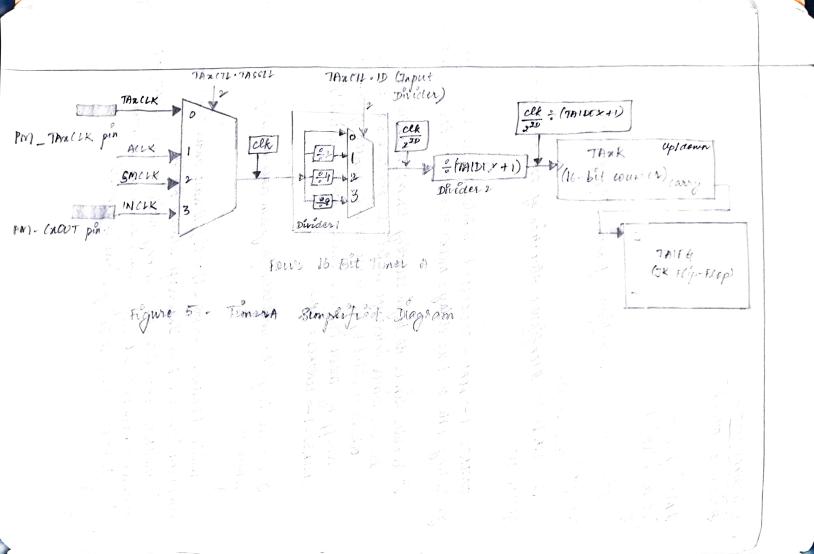
> Il main Function: int main (vaid)

> > $P2 \rightarrow SELO L = ^{2}$; N Configure functionality of P2-1 as simple GP10 pins. $P2 \rightarrow SELI L = ^{2}$;

> > P2 -> DIR 1 = 29 11 Configure direction of P2-1 as output for GREEN-15

Timer overflowed; Interrupt disabled; Nat clear: up made; Interrupt divider; Select clock source as SMCLK

TIMER_AI -> CTL = 0x02DV;



JIMER_AI → EXO = 7; / Durder 2 = TAIDEX + 1 > 7+1 = 8 TIMER_AI - CCR [0] = 47685-1; HFor generating 1 Hz or 1 sec delay; Assume 2° ID = 8, TAIDEX = 7; SMCLK = 30,00,000 Il Infinite Loop (An embedded pragham does not stop): while ((TIMER - A) -> CCTL [0] & 1) == 0); Il wait until the CCIFG flag is set TIMER_AI -> CCTL[0] b= M; Il clear interrupt flag

LED CONTROL USING TIMER PROGRAMMING : (Twen LED ON for 30 ms and LED OFF for 40 ms)

O Initialize a variable "temp for iteration, to turn HD ON and OFF consecutively. 10 Configure functionality of P21 as simple GP10 pins.

Configure direction of P2-1 as autput for GREEN-LED.

P2 - OUT ^= 2) // Trigger the GREEN-LED

Reload register value for generating 1 Hz or 1 sec delay: MCK = 30,00,000 HZ

Clear STCurrent Value Register. Enable Systick to begin counting down and disable interrupt generation. For the clock source, system clock (MCLK) is only implemented.

TWEN ON P2-1 GREEN-LED.

In an infinite while-loop, perform the fallowing operations.

On theck whether the Systick has counted down to zero and if COUNTFLAG is set.

(ii) If yes, check whether (temp % 2 = = 0), i.e. even number -- If yes, twen OFF P2-1 GREEN-LED. - alse, turn ON P2-1 GREEN-LED

(MI) Increment "stemp" by +1.

mSP432 Microcontroller General .

Figure 6- LED Control using Times A.

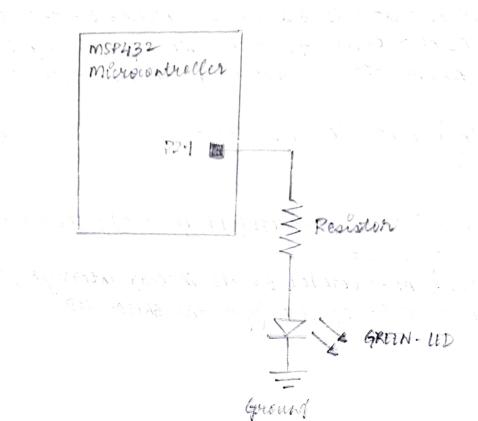
Table of Bullion

ing the contract of the contract of the contract of

and the second of the second

```
- C PROGRAM CODE-
  # include "mp. h"
  int temp = 0; Il variable for iteration to twen LED ON and OFF
  consecutively.
  11 main Function :
  int main (vaid)
    PI -> SELO L: N2; Il Configure functionality of P2.1 as simple
  gf10 pers
    P2 - SEL1 6 = N2)
    PI -> DIR 1 = 2; 11 Configure direction of P2.1 as output for
  GREEN- LED
     SysTick -> LOAD = 90000-1; // Reload negister value for
  generating 30 milliserands delay: MCIK = 30, 00, 000 HZ
     Systick -> VAI = 0; 11 Clear ST current Value Register
     Systick -> CTRL = 5; Il Enables Systick to begin counting down;
 Interrupt generation is disabled; System clock MCIK
      PI - OUT 1= 2; Il TWYN ON P2-1 GREEN-LED
      Il Infinite Loop (An embedded pregram does not stop):
      while (1)
        If The Syrick has counted down to zero;
        of (SysTick → CTRL & OX10000) A of COUNTFLAG is set
         11 Generate 40 milliseconds delay:
        if (temp % 2 = = 0)
          SysTick -> 10AD = 120000 -1;
          Systick -> VAL =0;
          SypTick → CTRL =5;
```

N → OUT &> ~2) // TWEN OFF P2.1 GREEN-LED



Commission for some that the first for fire Figure 7 - 200 Control using Timer Programming (Juin IED ON for 30 ms and IID OFF for 40, mi)

& GAMERICA ARMOD & ADMIT CONDITION OF THE

```
I Generate 30 milliseconds delay:

else

{
    SysTick → LOAD > 90000 -1;
    SysTick → VAL > 0;
    SysTick → CTRL > 5;
    P2→00T 1 > 2; 11 Twen ON P2 · 1 GREEN LED
}

temp++;
}
```

* RESULTE

Thus, an LED was controlled by developing a timer program using system Tick, Timer 32 and Timer A timers that can be interfaced with an MSP 43 n microcontroller. All simulation results were verified successfully.