|  |  |
| --- | --- |
| Liste der Timerbefehle | |
| Befehl | C |
| Timer mit Takt versorgen | RCC->APB1ENR |= 0b10000; //TIM6  RCC->APB1ENR |= 0b100000; //TIM7 |
| Timerstarten | TIMx->CR1=1; //setzt CEN auf 1 |
| Timer stoppen | TIMx->CR1=0; //setzt CEN auf 0 |
| Autoreloadregister mit *Wert* laden  (soweit zählt der Timer bevor er wieder mit 0 beginnt) | TIMx->ARR=*Wert*; |
| Prescaler einstellen  Wert=31 bedeutet Zählperiode 1µs  Wert 31999 bedeutet Zählperiode 1ms | TIMx->PSC=*Wert*; |
| Zähler auf 0 setzen  (auch andere Werte sind möglich) | TIMx->CNT=0; |
| Update Interrupt Flag (UIFx) zurücksetzen | TIMx->SR=0; |
| Timerinterrupt freigeben | TIMx->DIER=1; |
| Update Interrupt Flag (UIFx) abfragen | if (TIMx->SR==1)  {  } |

Hinweise:

- Bei Timer TIM6 x=6, bei Timer TIM7 x=7



Aufgaben:

1. PC\_0 soll mit einer Frequenz f=1Hz blinken (0,5s = 500ms Aus und ebenso lange an.

Vorgehensweise:

Im Unterprogramm initTimer:

- Timer mit Takt versorgen

- Timer mit Autoreloadwert für 500ms versorgen

- Prescaler für 1ms Zählperiode einstellen

- Timerzähler CNT mit 0 initialisieren

- Timer starten

Im Hauptprogramm:

initTimeraufrufen

Port PC\_0 als Digitalausgang initialisieren

Endlosschleife:

- warten auf UIF6=1

- UIF6 wieder zurückseetzen

- PC\_0 komplementieren

2. Digitaluhr  
Erweitern Sie 1. zu einer Digitaluhr (LCD siehe Anhang)  
  
3. Erweitern Sie die Digitaluhr so, dass mit den Tasten PA\_1, PA\_6 und PA\_10 die Uhr gestellt werden kann.

(PA\_1: Stunden, PA\_6: Minuten, PA\_10: Sekunden jeweils mit externem Interrupt)

4. Stoppuhr

* Zeitauflösung: 10ms
* Globale Variable int z=0; zählt die Zeit mit einer Auflösung von 10 ms
* PA\_1: Start mit externem Interrupt
* PA\_6: Stop mit externem Interrupt
* PA\_10: Reset mit externem Interrupt
* Anzeige:
  + CNT auf PortC
  + LCD Format: **s:ms** nur bei gestopptem Timer

Anhang:

Hinweise zur Programmierung:  
Da die Ausgabe auf das LCD-Display erfolgen soll benötigen Sie folgende Library:

Ergänzen Sie in der main.cpp an den entsprechenden Stellen:



Doppelklick

Library LCD\_i2c\_GSOE suchen

Import



LCD-Display-Objekt mit Namen z.B. mylcd  
deklarieren und erzeugen

Library mit #include „LCD.h“  
einbinden



Befehle:

* mylcd.clear(); //löscht das Display
* mylcd.cursorpos(*wert*); //plaziert den Cursor  
  0..15: 1.Zeile, 64..79 2.Zeile (64=0x40)
* mylcd.printf(*Formatstrinng,Werte*); //Ausgabe. Doku: Internet printf