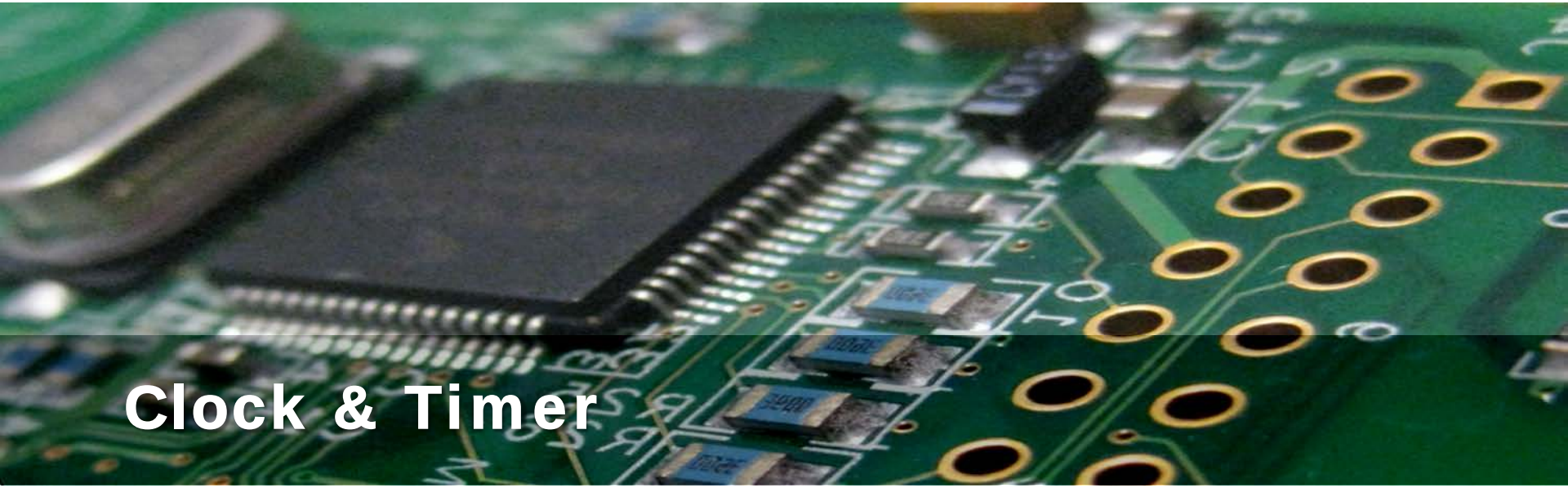
Lucerne University of
Applied Sciences and Arts
HOCHSCHULE
LUZERN
Technik & Architektur



Clock & Timer

„Realtime means we have some connection to the real time, right?“

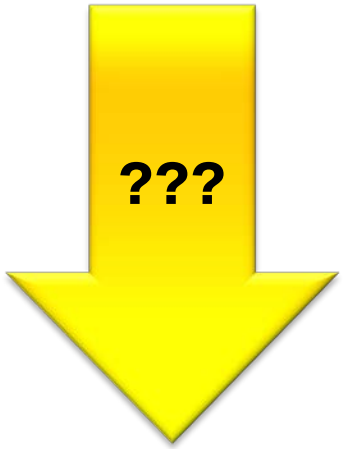
Prof. Erich Styger
erich.styger@hslu.ch
+41 41 349 33 01

Learning Goals

- Problem: Setting up a periodic timer

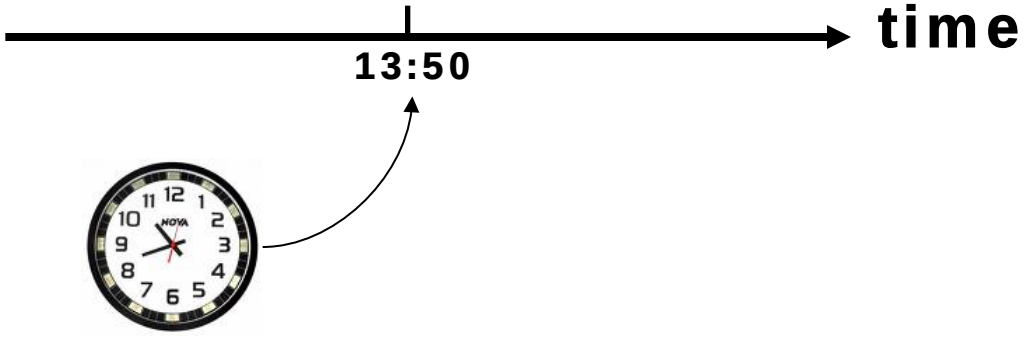
- Clock
  - Clock Block Diagram
  - Clock Configuration

- Timer
  - Timer Setup
  - Timer Interrupt
  - LED flashing (1 sec interval)



Realtime Systems: Timeliness

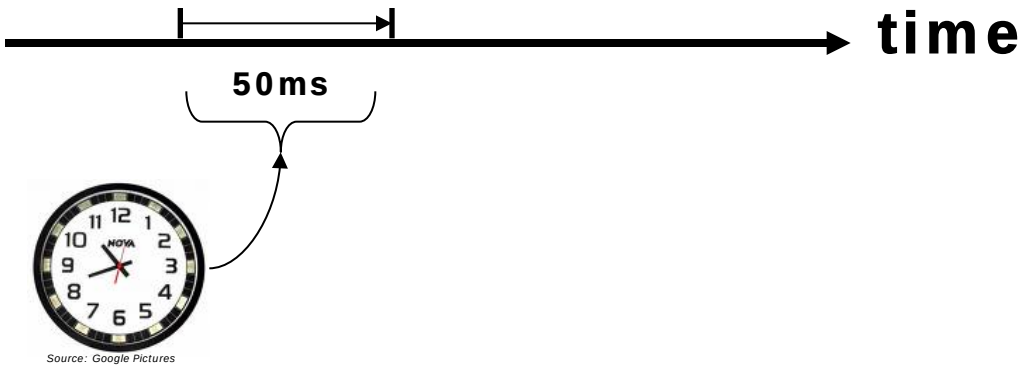
- Categories
  - Absolute
  - Relative



Source: Google Pictures

- Need
  - Time base
  - Clock
  - Interrupt Synchronization

- Derived
  - Timer
  - Time



Source: Google Pictures

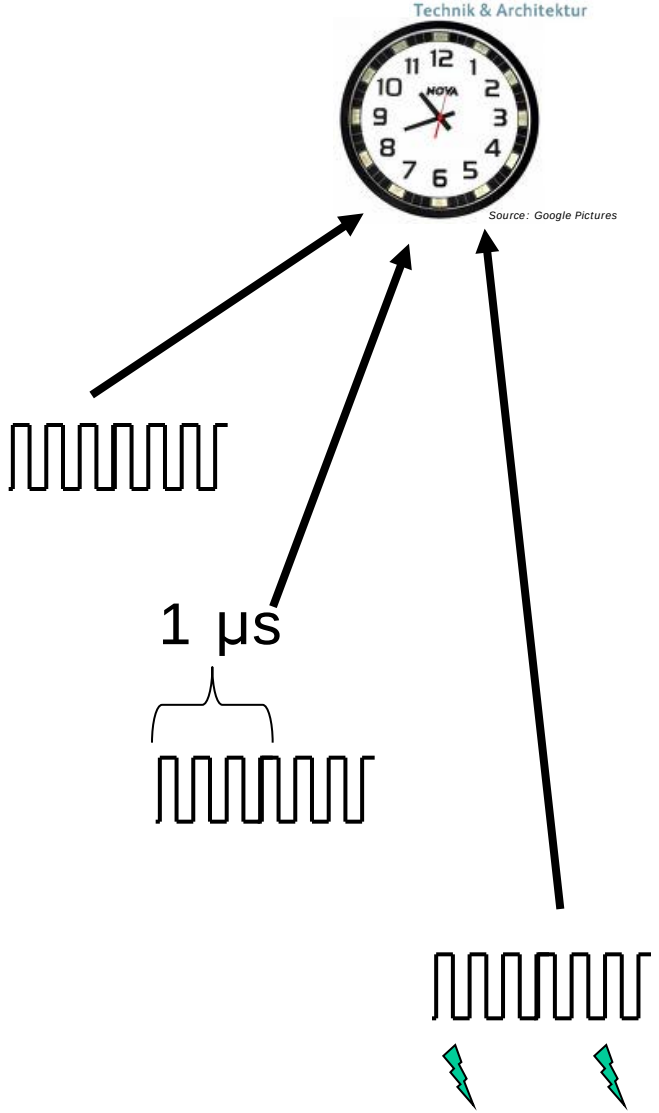
What do we need?

- Linkage to the real time
  - ns, µs, ms, s, h, ...

- Periodic Ticks
  - Known real time tick period
  - External or Internal source
  - System/CPU/Bus clock

- Operations
  - Counting ticks
  - Sum/Calculation: real time entity

- Synchronization with counter(s)
  - Events
  - Flags



Remote: K20 Clocks

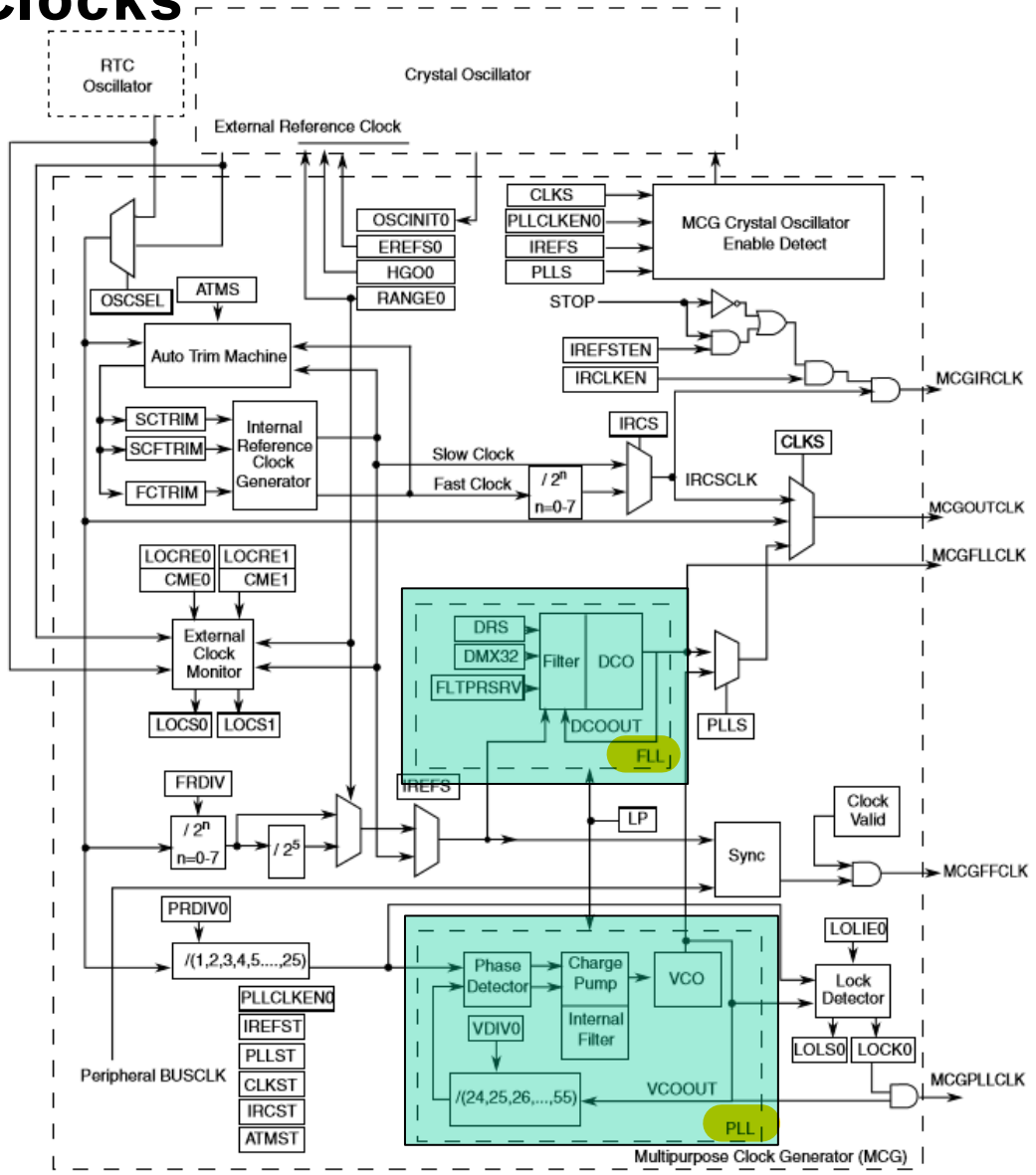
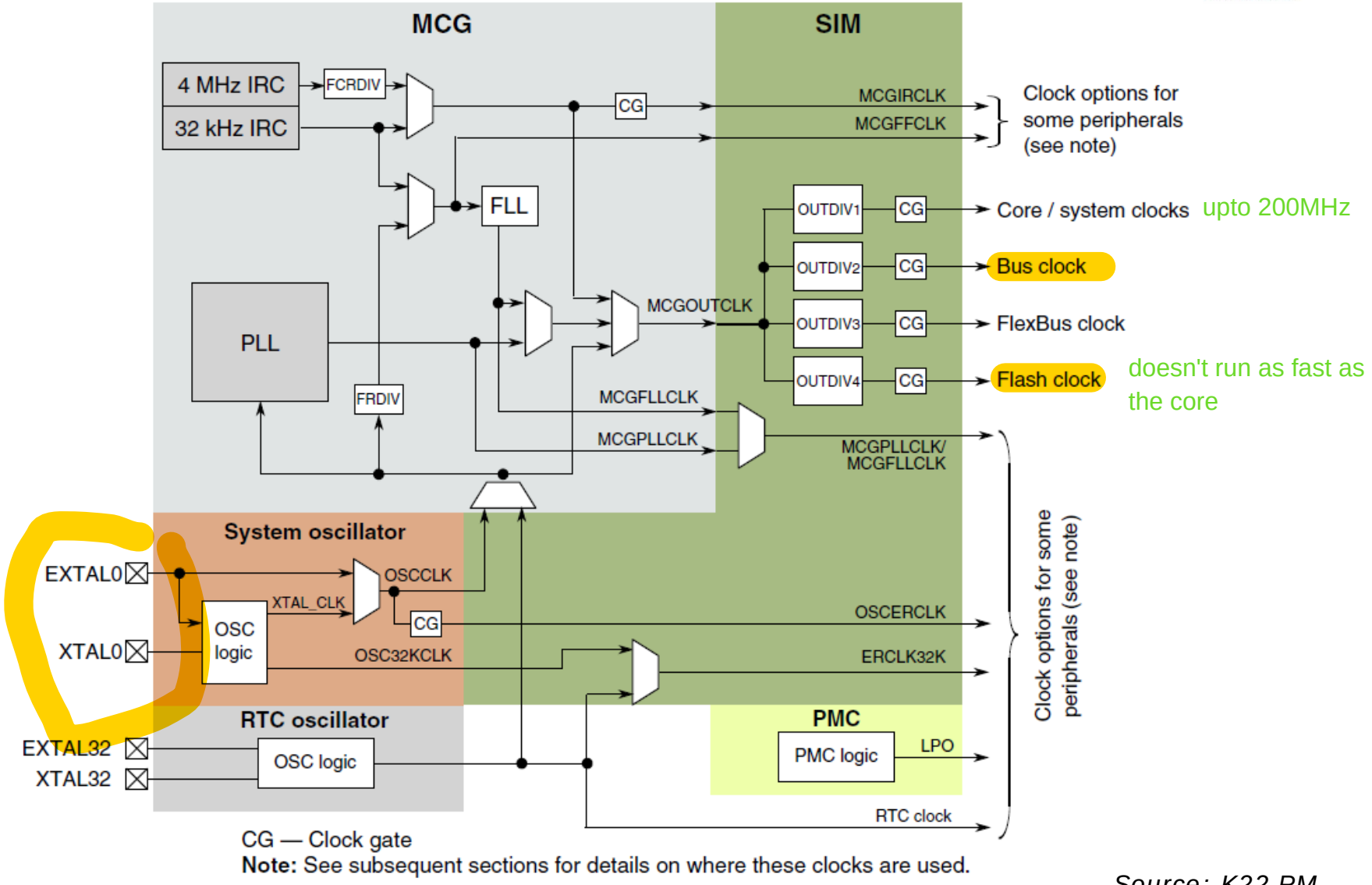


Figure 24-1. Multipurpose Clock Generator (MCG) block diagram

Source: K20 RM

Robo: K22 Clocks



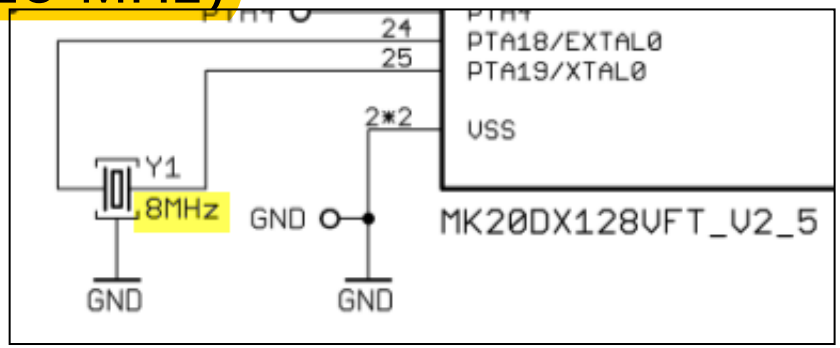
Source: K22 RM

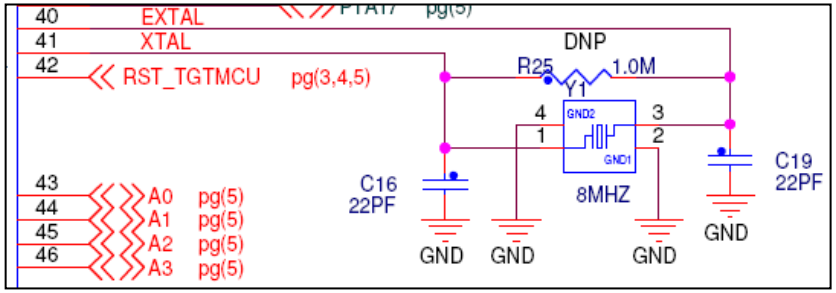
FRDM & Remote Board

- 8 MHz Crystal
- Enable System Oscillator
- Set PEE mode
- 48 MHz Core/Bus Clock (Remote: 50 MHz)
- 24 MHz Flash Clock (Remote: 25 MHz)
- Details in Lab Assignment

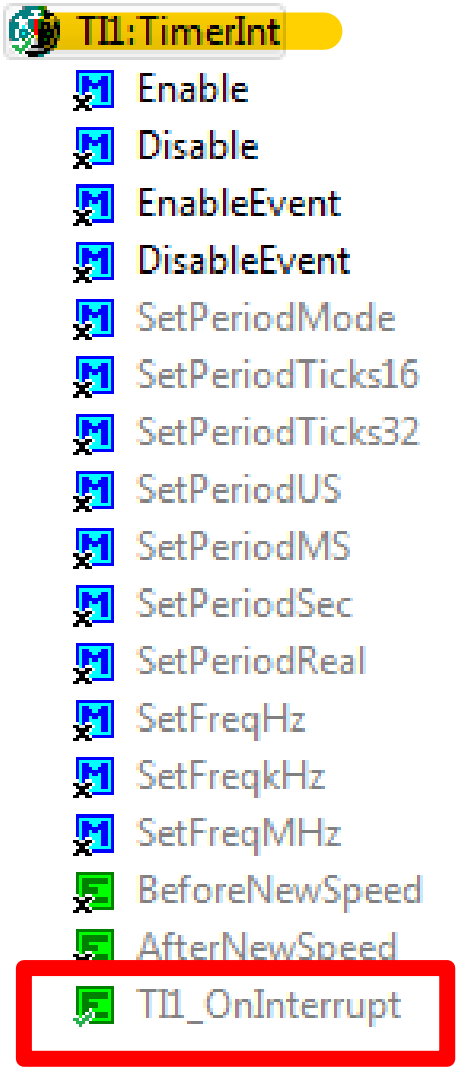
System oscillator 0	Enabled
Clock source	External crystal
Clock input pin	
Pin name	EXTAL0/PTA18/UART1_RX/TPM_C...
Pin signal	EXTAL
Clock output pin	
Pin name	XTAL0/PTA19/UART1_TX/TPM_CL...
Pin signal	XTAL
Clock frequency [MHz]	8.0
Capacitor load	0pF
Oscillator operating mode	Low power

robot and remote





Timer (TimerInt)



Not (!!)
LPTMR

Properties | Methods

Name	Value	Details
Component name	TI1	
Periodic interrupt source	TPM11	TPM11
Counter	TPM1	TPM1 [shared c
Interrupt service/event	Enabled	
Interrupt	Vtpm1ch1	Vtpm1ch1
Interrupt priority	medium priority	not supported
Interrupt period	10 ms	10.000 ms
Same period in modes	yes	
Component uses entire timer	no	
Initialization		
Enabled in init. code	yes	
Events enabled in init.	yes	
CPU clock/speed selection		
High speed mode	This component...	This component
Low speed mode	This component...	This component
Slow speed mode	This component...	This component

timer interrupt will fire every 10ms

Timer Interface

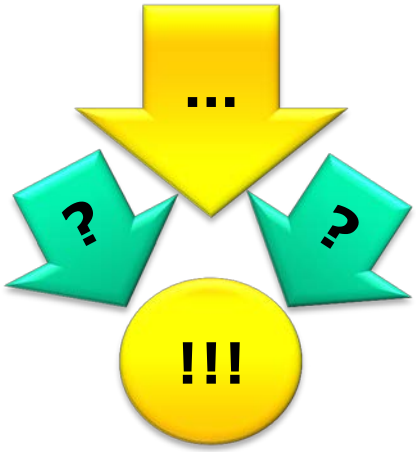
```
#define TMR_TICK_MS  10
  /* we get called every 10 ms */

/*!
 * \brief Function called from timer interrupt
 every TMR_TICK_MS.
 */
void TMR_OnInterrupt(void);

/*! \brief Timer driver initialization */
void TMR_Init(void);
```

Summary

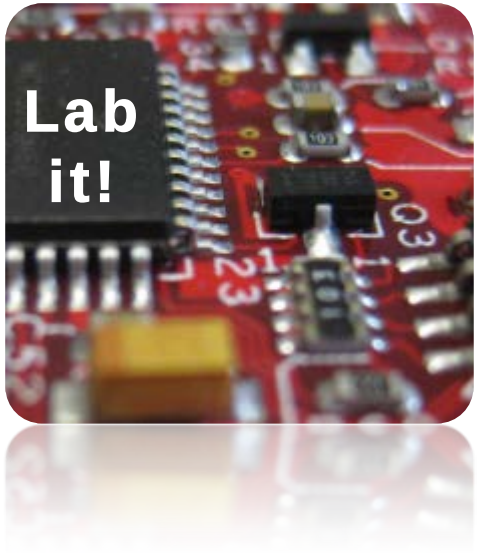
- Different clock sources
  - External Crystal/Oscillator
  - Internal Clock
- CPU clock vs. Bus Clock
- Setting up periodic Clock



Lab: Clock and Timer

- Set/Verify Clock settings

Board	Core	Bus	Flash
Remote	50 MHz	50 MHz	25 MHz
Robo	120 MHz	60 MHz	24 MHz



- Add 10ms Timer
  - Processor Expert: TimerInt
  - Timer.c/Timer.h
  - Blink LED every 1 s
    - use an event