


Politechnika Wrocławska

Technika cyfrowa 2


wykład 8

Katedra Metrologii Elektronicznej i Fotonicznej
Andrzej Stępień




Jak w mikrokontrolerach

- zliczać pojedyncze impulsy N i paczki impulsów ?
- mierzyć czas t , częstotliwość f i przesunięcie fazy ϕ ?
- wygenerować pojedynczy sygnał prostokątny o zadanym okresie T i współczynniku wypełnienia $\tau = \frac{1}{2}$?
- wygenerować pojedynczy sygnał prostokątny o programowalnym okresie T i współczynniku wypełnienia τ ?
- wygenerować kilka synchronicznych sygnałów prostokątnych o programowalnych okresach T_i i współczynnikach wypełnienia τ_i ?
- uzyskać sygnał stałoprądowy o programowalnej wartości średniej U_{sr} ?
- stworzyć zmienny sygnał o założonych parametrach: kształcie, amplitudzie i okresie ?

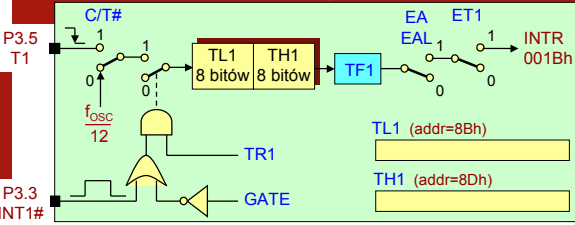


C51: Timers - Overview

- two 16-bit timers/counters (Timer 0 and 1) which are useful in many applications for timing and counting.
- Timer/counter 0 and 1 of the C51 and can be used in the same four operating modes:
 - Mode 0: 8-bit timer/counter with a divide-by-32 prescaler
 - Mode 1: 16-bit timer/counter
 - Mode 2: 8-bit timer/counter with 8-bit auto-reload
 - Mode 3: Timer/counter 0 is configured as one 8-bit timer/counter and one 8-bit timer; Timer/counter 1 in this mode holds its count.
- External inputs INT0 and INT1 can be programmed to function as a gate for timer/ counters 0 and 1 to facilitate pulse width measurements.



Licznik T0, T1 - tryb 1




TMOD (addr = 89h)

GATE C/T# M1=0 M0=1 GATE C/T# M1=0 M0=1

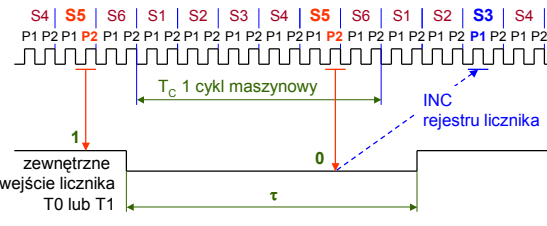
IE / IEN0 (addr = 0A8h)

EA EAL ET1 ET0 TF1 TR1 TF0 TR0

TCON (addr = 88h)



Zliczanie impulsów



Zliczenie zewnętrznego impulsu na wejściu licznika T0 lub T1:

$\tau > T_c$

$f_x < f_{osc}/24$



Rejestry liczników T0 i T1

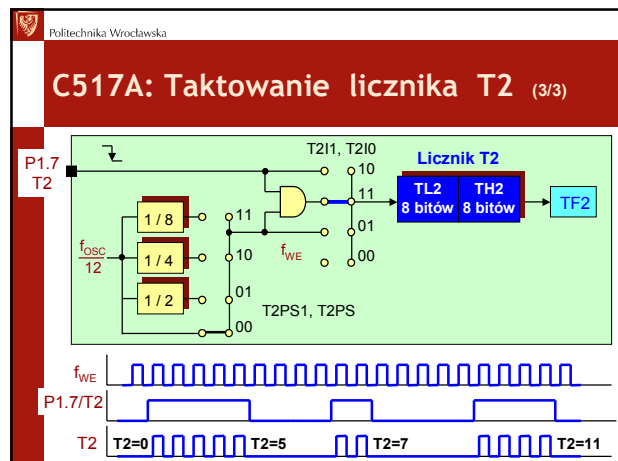
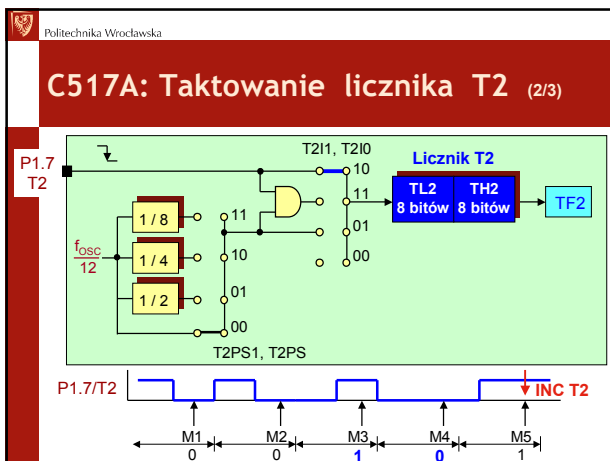
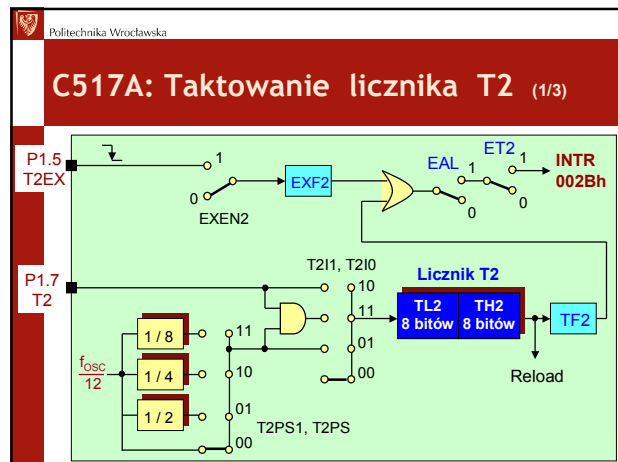
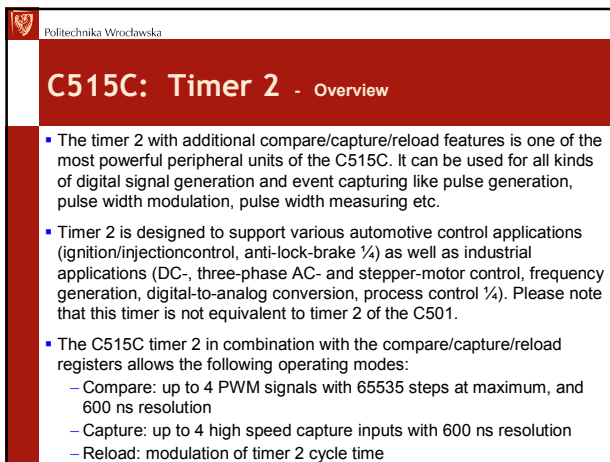
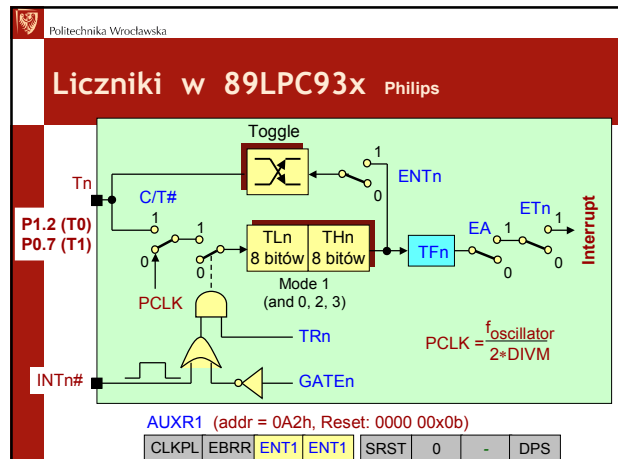
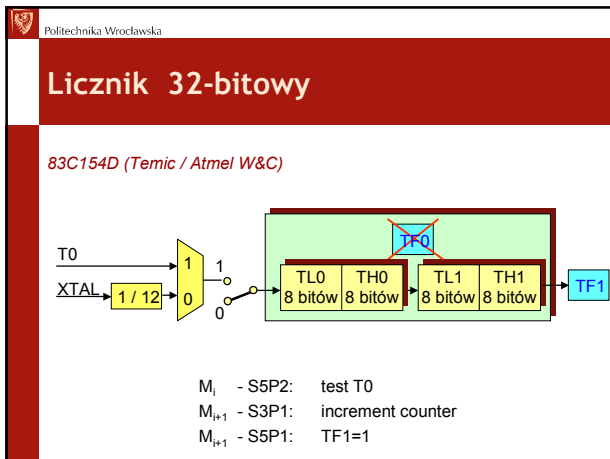
SFR


0B0h	P3								
0A8h	IE / IEN0								
0A0h	P2								
098h	SCON	SBUF							
090h	P1								
088h	TCON	TMOD	TL0	TL1	TH0	TH1			
080h	P0	SP	DPL	DPH					PCON

rejestry licznika T0

rejestry licznika T1

Brak możliwości zapisu zmiennych typu integer do rejestrów liczników T0 lub T1




Politechnika Wrocławska

C517A: Rejestry licznika T2

S

F

R

0FE0h	P5							
0F0h	B		CML6	CMH6	CML7	CMH7	CMEN	CMSEL
0E8h	P4							
0E0h	Acc	CTCON	CML3	CMH3	CML4	CMH4	CML5	CMH5
0D8h								
0D0h	PSW		CML0	CMH0	CML1	CMH1	CML2	CMH2
0C8h	T2CON	CC4EN	CRCL	CRCH	TL2	TH2	CCL4	CCH4
0C0h	IRCON	CCEN	CCL1	CCH1	CCL2	CCH2	CCL3	CCH3
0B8h	IEN1							
0B0h	P3							
0A8h	IEN0							
0A0h	P2	COMSETL	COMSETH	COMCLRL	COMCLRH	SETMSK	CLRMSK	
088h	TCON	TMOD	TL0	TL1	TH0	TH1		

Możliwość zapisu zmiennych typu integer do rejestrów licznika T2

Pojęcia

- **Pulse Width Modulation** - modulacja szerokości impulsu
- **Reload** – programowy/sprzętowy wpis wartości początkowej licznika
- **Compare** – sprzętowe porównanie wartości chwilowej licznika z zadany stanem
- **Capture** – programowy/sprzętowy wpis (przechwycenie) wartości chwilowej licznika

Pulse Width Modulation (1/2)

- modulacja szerokości impulsu τ

- wartość średnia napięcia: $U_{sr} = V_{cc} * \tau / T$

Pulse Width Modulation (2/2)

- wartość średnia napięcia: $U_{sr} = V_{cc} * \tau / T$
- generator sygnału

Changing the Amplitude without using any Multiplication – Instruction (1/2)

www.infineon.com

AP08022 Generating sinusoidal 3-Phase-Currents for Induction Machines with a

$$U = A * \sin B$$

- B corresponds to the angle variable high byte
- sin B corresponds to the value in the sine table to be multiplied with the amplitude A
- U corresponds to the value moved to the compare register
- To avoid the multiplication, following equation shows a solution:

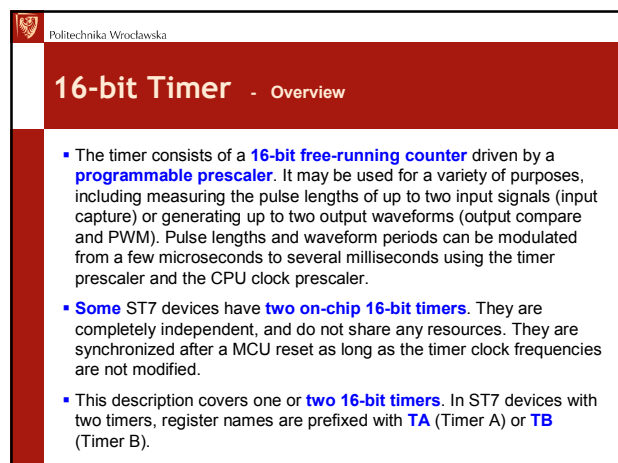
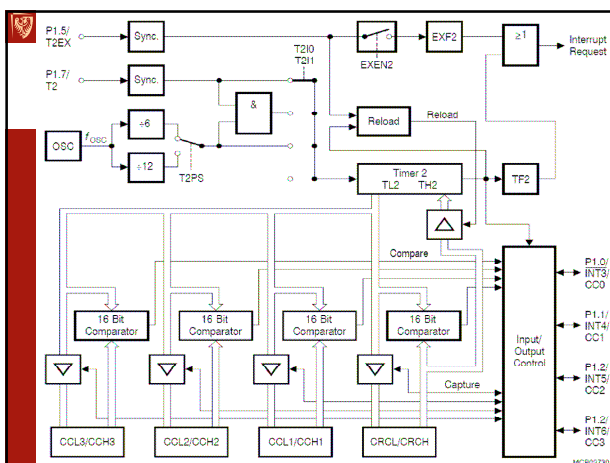
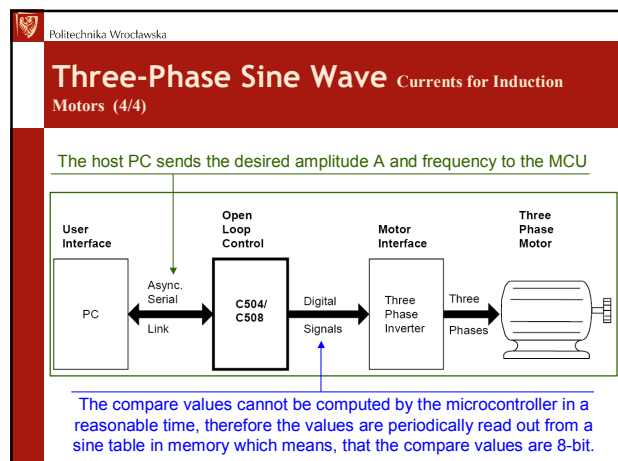
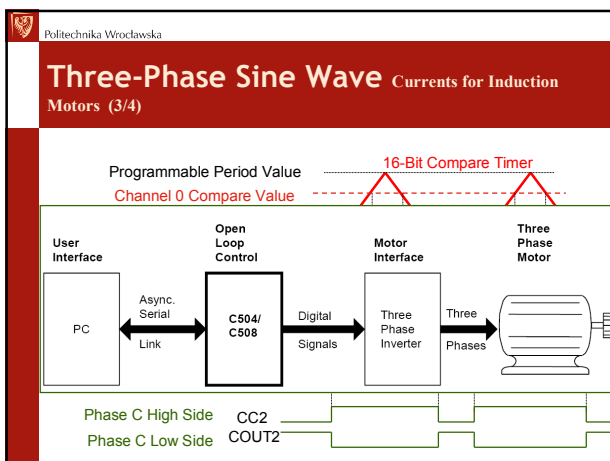
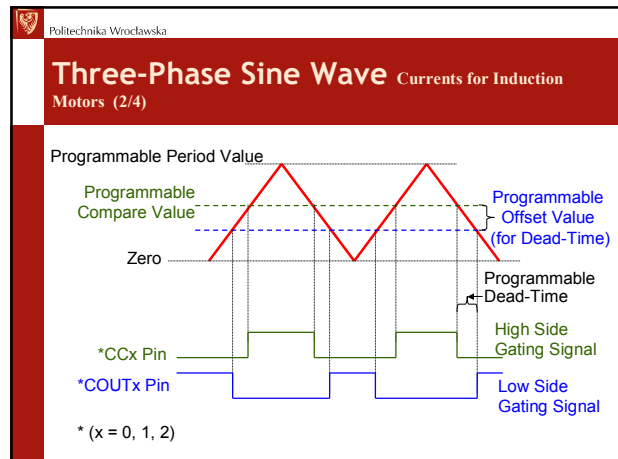
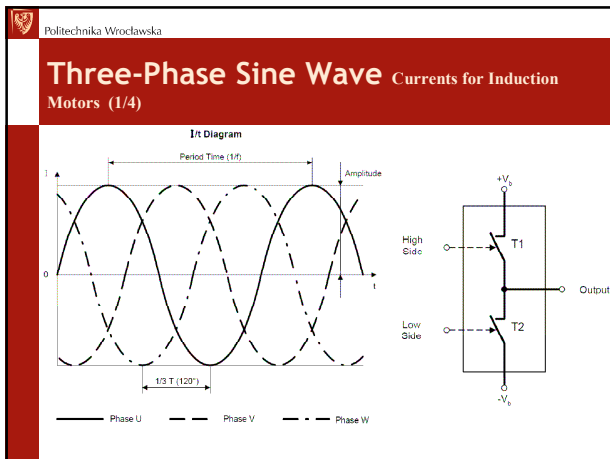
$$U = A * \sin B = \cos(\arccos A) * \sin B = \cos A' * \sin B = \frac{1}{2} * [\sin(B - A') + \sin(B + A')]$$

- The multiplication of A and sinB is now transferred with an addition theorem into the operations B-A' and B+A' and two sine table accesses.

Changing the Amplitude without using any Multiplication – Instruction (2/2)

$$\sin X + \sin X = 2 * \sin X$$

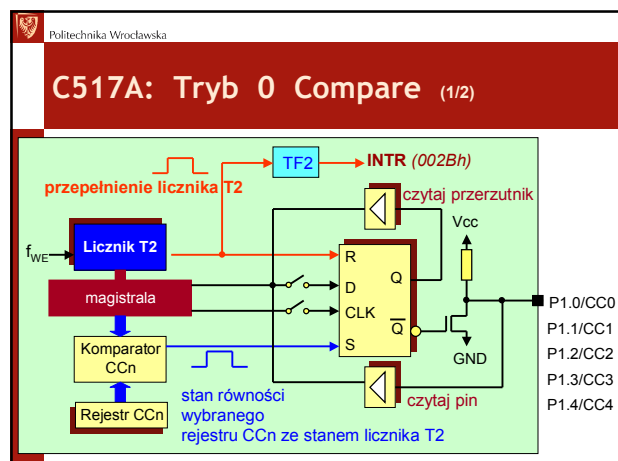
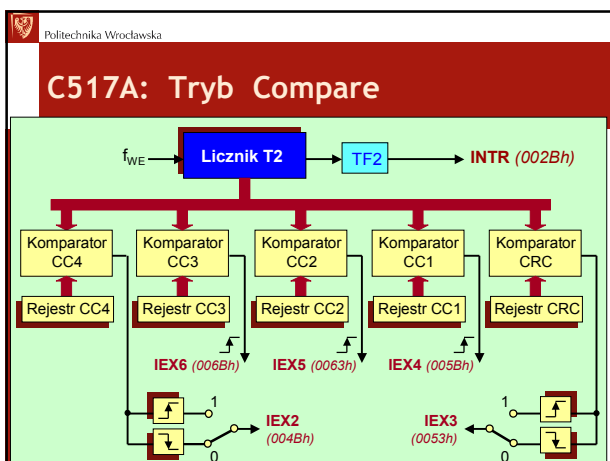
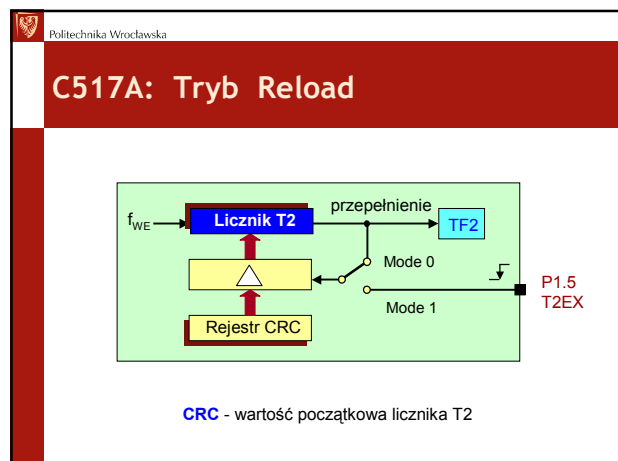
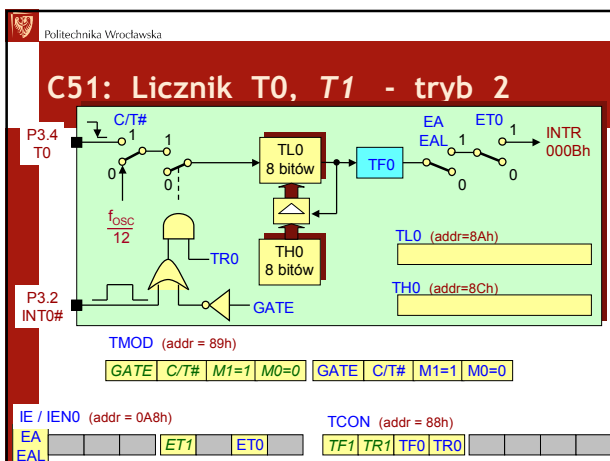
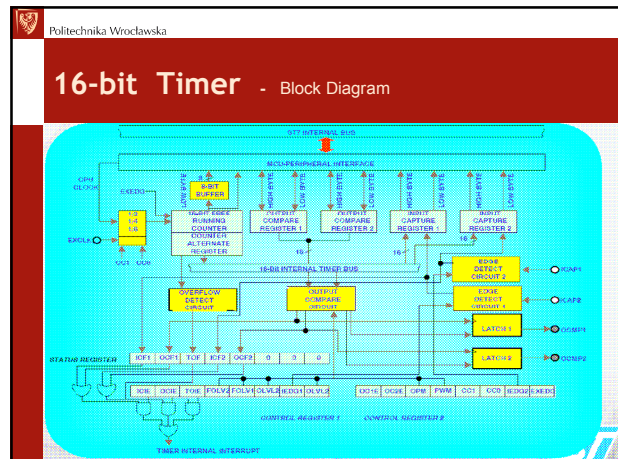
$$\sin X + \sin(X + 180^\circ) = 0$$

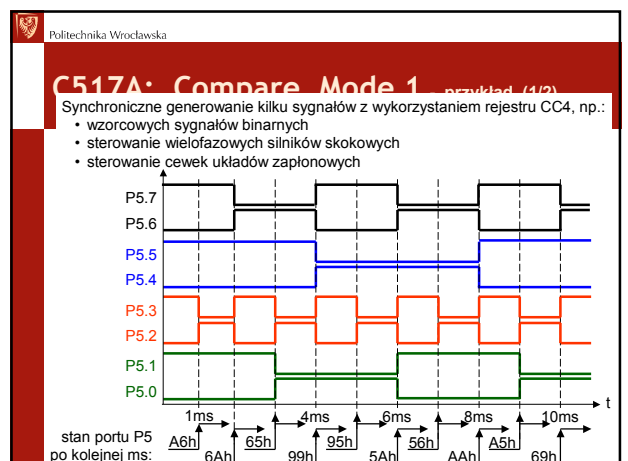
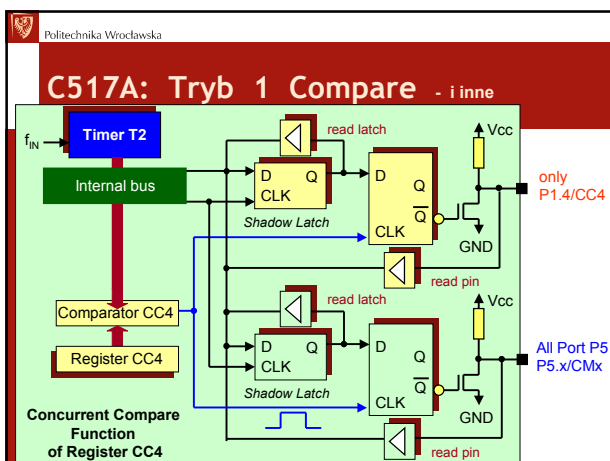
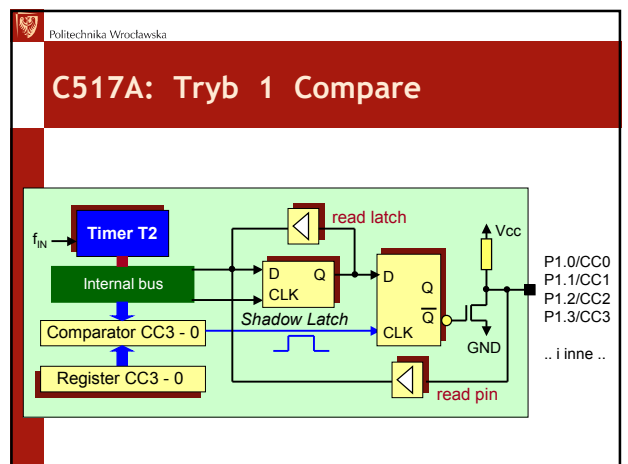
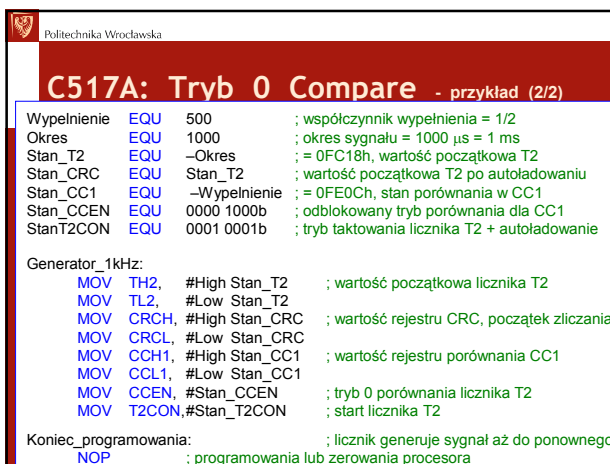
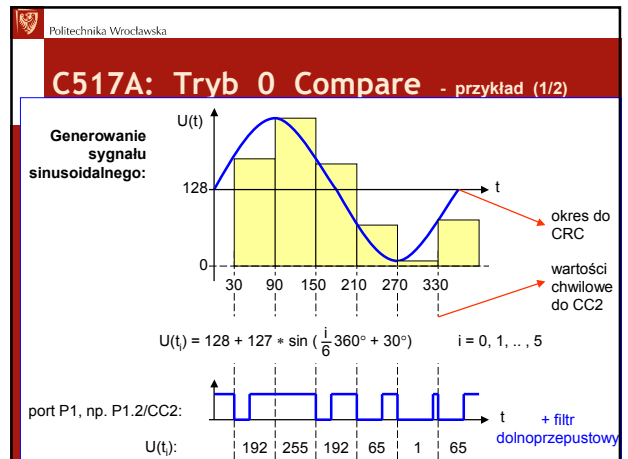
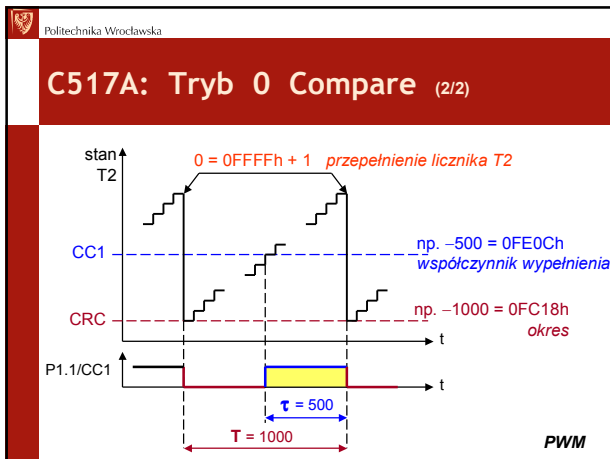


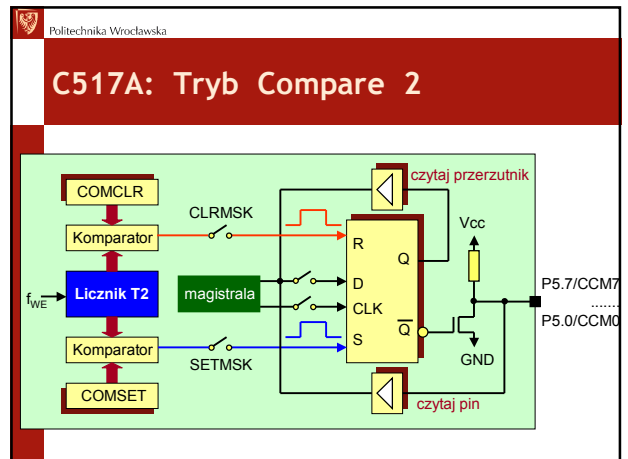
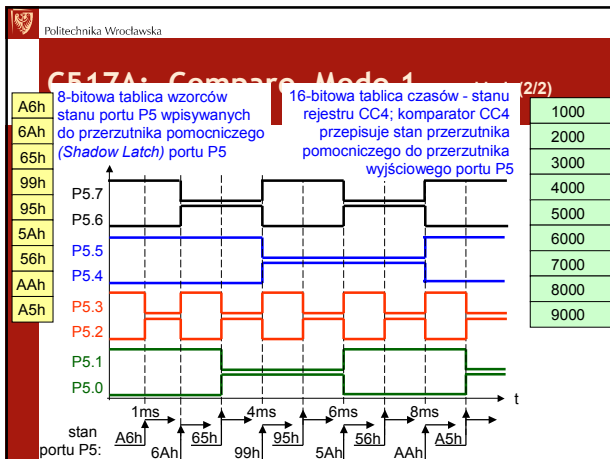
Politechnika Wrocławska

16-bit Timer - Main Features

- Programmable prescaler: f_{CPU} divided by 2, 4 or 8.
- Overflow status flag and maskable interrupt
- External clock input (must be at least 4 times slower than the CPU clock speed) with the choice of active edge
- Output compare functions with:
 - 2 dedicated 16-bit registers
 - 2 dedicated status flags
 - 2 dedicated programmable signals
 - 1 dedicated maskable interrupt
- Input capture functions with:
 - 2 dedicated 16-bit registers
 - 2 dedicated active edge selection signals
 - 2 dedicated status flags
 - 1 dedicated maskable interrupt
- Pulse Width Modulation mode (PWM)
- One Pulse mode
- alternate functions on I/O ports: ICAP1, ICAP2, OCMP1, OCMP2, EXTCCLK



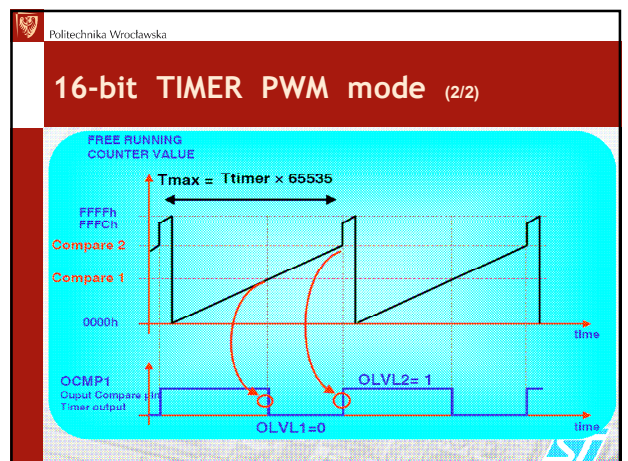




Politechnika Wrocławska

16-bit TIMER PWM mode (1/2)

- Automatic generation of a Pulse Width Modulated signal
- Period & pulse length set by software:
 - The first Output Compare Register OC1R contains the length of the pulse
 - The second Output Compare Register OCR2 contains the period of the pulse
- Resolution up to 100 steps at 20 KHz (fCPU = 4 MHz): 1% of accuracy on the duty cycle



Politechnika Wrocławska

C517A: Tryb Compare 1 (1/2)

- wpisywana wartość do licznika : 0x0100
- czas trwania instrukcji: MOV addr, #dana ; 2 cykle maszynowe
CLR TR0 ; 1 cykl maszynowy

```

MOV TMOD, #1 ; tryb 1 (16-bitowy) obu liczników
MOV TL0, #LOW StanPoczT0 ; StanPoczT0:
MOV TH0, #HIGH StanPoczT0 ;
SETB TR0
MOV TL0, #00
MOV TH0, #01
CLR TR0 ; stan
Stop1: ; licznika T0: 0103h 0103h
MOV TL0, #LOW StanPoczT0
MOV TH0, #HIGH StanPoczT0
SETB TR0
MOV TH0, #01
MOV TL0, #00
CLR TR0 ; stan
Stop2: ; licznika T0: 0201h 0101h

```

Politechnika Wrocławska

Wpis/odczyt synchroniczny (1/2)

- programowy wpis synchroniczny:**
 - zatrzymanie licznika na czas wpisu
 - wpis części mniej znaczącej
 - wpis części bardziej znaczącej
 - ponowne uruchomienie licznika
 - korekta wpisywanej wartości o czas wpisu, zatrzymania, uruchomienia licznika
- sprzętowy wpis synchroniczny**, np. Atmega8L:

```

Write: LDI R17, 0x01
       LDI R16, 0xFF

       OUT TCNT1H, R17 ; TempReg ← R17, high byte first
       OUT TCNT1L, R16 ; TCNT1L ← R16, low byte second
                          ; TCNT1H ← TempReg

```


- Writing the counter LSB resets the timer at FFFCh

- Read MSB first and then the LSB
- The counter LSB is buffered during the MSB read, byte value is buffered automatically.

-
- Beginning of the sequence*
- At t_0 [Read MS Byte] → [LS Byte is buffered]
- ↓
[Other instructions]
- At $t_0 + \Delta t$ [Read LS Byte] → [Returns the buffered LS Byte value at t_0]
- Sequence completed*

16-bit TIMER: Capture

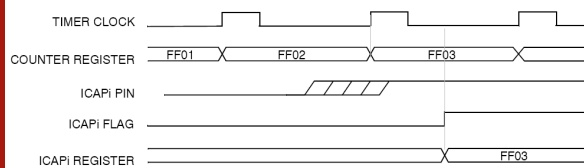
- Captures the counter value upon input signal edge detection
 - Allows an external pulse length measurement
 - Internal safety process in case of critical interrupts timing
-
- The ICFI bit is set.
 - The ICIR register contains the value of the free running counter on the active transition on the ICAP1 pin.
 - A timer interrupt is generated if the ICIE bit is set and the I bit is cleared in the CC register. Otherwise, the interrupt remains pending until both conditions become true.

16-bit TIMER: Capture - Block Diagram

-
- The diagram illustrates the control logic for ICAP1 and ICAP2. ICAP1 and ICAP2 are inputs to two edge detect circuits. The outputs of these circuits are connected to the IC1N and IC2N registers, respectively. These registers then feed into a 16-bit free-running counter. The counter's output is split into two 16-bit registers: ICF1-ICF4 and ICF5-ICF8. The ICF1-ICF4 register is connected to Control Register 1 (CR1), and the ICF5-ICF8 register is connected to Control Register 2 (CR2). The ICF5-ICF8 register also feeds into the Status Register (SR). Red circles highlight the IC1N input in CR1, the IC1N input in SR, and the IC1N input in CR2.

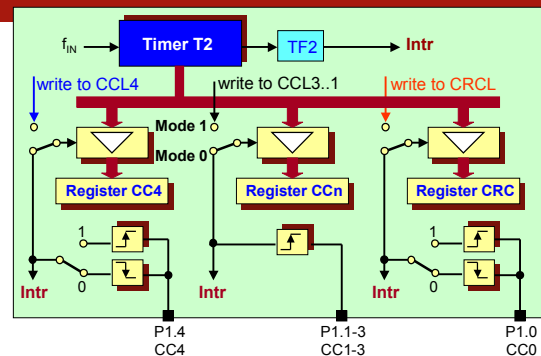
- **CC[1:0]** Select the timer clock
- Set the **ICIE** bit to generate an interrupt after an input capture coming from either the ICAPn pin

16-bit TIMER: Capture - Timing Diagram

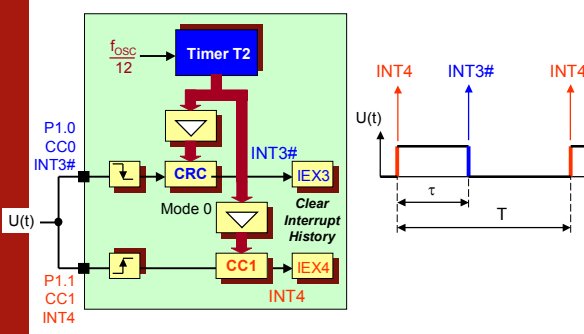


Note: Active edge is rising edge.

C517A: Capture



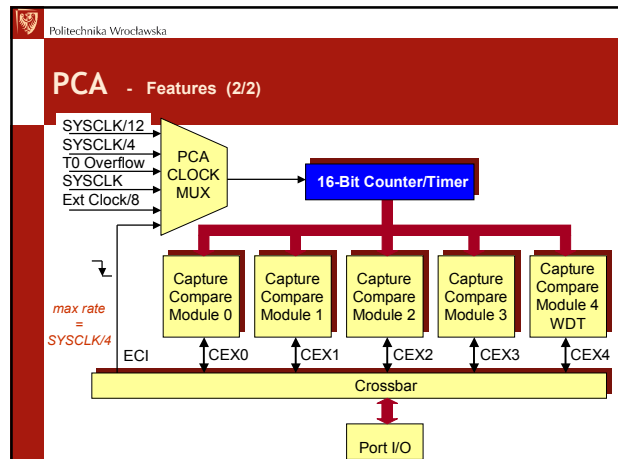
C517A: Capture - example



Politechnika Wrocławska

PCA - Features (1/2)

- The **PCA** consists of a :
 - dedicated 16-bit counter/timer
 - five 16-bit capture/compare modules
 - each capture/compare module has its own associated I/O line (CEXn)
 - I/O lines are routed through the Crossbar to Port I/O when enabled.
- Each capture/compare module may be configured to operate independently in one of six modes:
 - Edge-Triggered Capture
 - Software Timer, High-Speed Output,
 - Frequency Output,
 - 8-Bit PWM
 - 16-Bit PWM
- The **PCA is configured and controlled** through the system controller's Special Function Registers (SFR).



Politechnika Wrocławska

Sygnal dźwiękowy - PROGRAMOWO (1/3)

; generowanie sygnału dźwiękowego o częstotliwości 2 kHz:
 ; - czas trwania zera i jedynki logicznej: $T = 0,25 \text{ ms} = 250 \mu\text{s}$,
 ; - częstotliwość oscylatora kwarcowego mikrokontrolera: $f_{\text{OSC}} = 12 \text{ MHz}$

```

Buzzer EQU P1.1 ; adres przetwornika piezoelektrycznego
BuzSoft: CPL Buzzer ; 1 cykl maszynowy, negacja linii sterującej
        LCALL Czas ; 2 wywołanie podprogramu opóźnienia
        SJMP BuzSoft ; 2 powtórzenie programu

Czas: MOV A, #242/2 ; 1 podprog. opóźnienia czasowego
      ; 250 - 8 = 242
      ; w akumulatorze A liczba powtórzeń pętli
      DJNZ ACC, $ ; 2 zasadnicza pętla opóźnienia
      RET ; 2 powrót z podprogramu
  
```

Politechnika Wrocławska

Sygnal dźwiękowy - SPRZĘTOWO (2/3)

; generowanie sygnału dźwiękowego o częstotliwości 2 kHz (Timer T0):
 ; - czas trwania zera i jedynki logicznej: $T = 0,25 \text{ ms} = 250 \mu\text{s}$,
 ; - częstotliwość oscylatora kwarcowego mikrokontrolera: $f_{\text{OSC}} = 12 \text{ MHz}$

```

Buzzer EQU P1.1 ; adres przetwornika piezoelektrycznego
TMOD_T0 EQU 0000$0010b ; GATE=0, C/T#=0 (taktowanie wewn.)
          ; M1,M0=10 (8-bit autoreload)
T0_Pocz EQU -250 ; wartość początkowa licznika T0

BuzHard: MOV TMOD, #TMOD_T0 ; programowanie trybu licznika T0
         MOV TL0, #T0_Pocz ; wartość początkowa licznika T0
         MOV TH0, #T0_Pocz
         SETB TR0 ; start licznika T0

Czekaj: JNB TF0, $ ; 2 oczekiwanie na przepelnienie T0
        CPL Buzzer ; 1 negacja linii z przetwornikiem piezo
        CLR TF0 ; 1 kasowanie znacznika przepełnienia
        JMP Czekaj ; 2 powtórzenie pętli programu
  
```

Politechnika Wrocławska

Sygnal dźwiękowy - PRZERWANIE (3/3)

; czasy trwania sygnałów jak poprzednio (Timer T1)

```

Buzzer EQU P1.1 ; adres przetwornika piezoelektrycznego
TMOD_T1 EQU 0010$0000b ; GATE=0, C/T#=0 (taktowanie wewn.)
          ; M1,M0=10 (8-bit autoreload)
T1_Pocz EQU -250 ; wartość początkowa licznika T1 (-250=6)
IE_T1 EQU 1000$1000b ; odblokowanie przerwań: EA = ET1 = 1

BuzHard: MOV TMOD, #TMOD_T1 ; programowanie trybu licznika T1
         MOV TL1, #T1_Pocz ; wartość początkowa licznika T1
         MOV TH1, #T1_Pocz
         SETB TR1 ; start licznika T1
         MOV IE, #IE_T1 ; odblokowanie przerwań
         ; cd programu użytkownika

ORG 1Bh
_Int_T1: CPL Buzzer ; TL1 = 9 = 6 (wart. pocz. liczn. T1) + 1 + 2
        RETI ; koniec procedury obsługi przerwania
  
```