



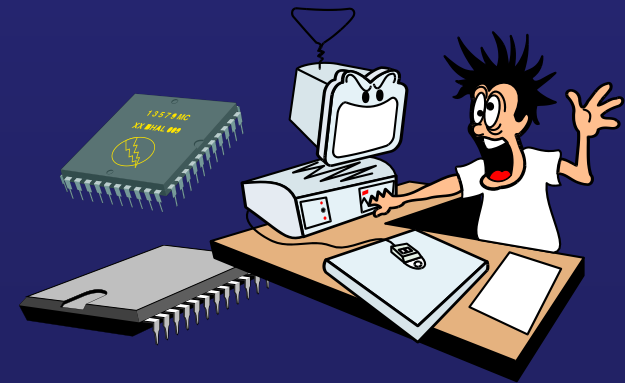
WYDZIAŁ FIZYKI
i INFORMATYKI STOSOWANEJ
Uniwersytet Łódzki



Systemy wbudowane



Witold Kozłowski



<https://std2.phys.uni.lodz.pl/mikroprocesory/>

Systemy wbudowane

Kierunek: Informatyka
PRACOWNIA DYDAKTYCZNA

Uwaga !!!

**Proszę o wyłączenie
telefonów komórkowych**

na wykładzie i laboratorium

Wykład 7.

Przykłady zegarów



Zegary



Pomiary czasu są jednym z głównych problemów z jakimi borykała się ludzkość od zarania dziejów

Zegar - przyrząd do ciągłego pomiaru czasu.

Zegary można podzielić na :

- wykorzystujące do wskazań tarczę i wskazówki oraz np. symbole, fazy księżyca
- wykorzystujące do wskazań wyświetlacze ciekłokrystaliczne lub diodowe.

W zegarach powszechnego użytku, wskazania obejmują najczęściej godziny, minuty i sekundy. W niektórych zegarach stosowany jest także system kalendarzowy do wskazywania dni tygodnia, dat, itp.

W zegarach specjalistycznych stosuje się bardzo precyzyjne mechanizmy wskazywania i rejestracji czasu z dokładnością do ułamków sekundy.

Jak ongiś mierzono upływ czasu

Powszechnie sądzi się, że w dawnych czasach człowiek — podobnie jak zwierzęta i rośliny — żył według Słońca, podporządkowując się następstwu dnia i nocy oraz porom roku.

To właśnie codzienna zmiana blasku dnia i mroku nocy regulowała życiową aktywność człowieka wyznaczając godziny jego pracy i odpoczynku.

Zapewne później pojawiła się potrzeba określenia stopniowo coraz mniejszych jednostek: „półgodzin”, kwadransów, minut i sekund. Warto tu wspomnieć, że nieco irracjonalny podział godziny czy stopnia — na akurat 60 mniejszych jednostek (minut), a tych na kolejne 60 jeszcze mniejszych — pochodzi od Babilończyków, którzy używali systemu sześćdziesiątego.

Historia

- zegar słoneczny
- 2700 p.n.e. - zegar wodny i piaskowy (Klepsydra)
- 724 - zegar mechaniczny Liang Ling-Son - Chiny
- 1335 - mechaniczny zegar wieżowy
- 1364 - data wydania pierwszego szczegółowego opisu mechanizmu zegarowego autorstwa Giovanni de Dondi
- ok. 1600 - zegarek kieszonkowy
- koniec XVI wieku - wskazówka minutowa
- 1657 - zegar wahadłowy - Ch. Huygens, A. A. Kochański
- 1665 - wskazówka sekundowa
- 1761 - chronometr okrętowy - John Harrison
- ok. 1860 - zegar elektromagnetyczny - Matthias Hipp
- 1880 - zegarek naręczny - Girard-Perregaux
- 1929 - zegar kwarcowy
- 1949 - zegar atomowy

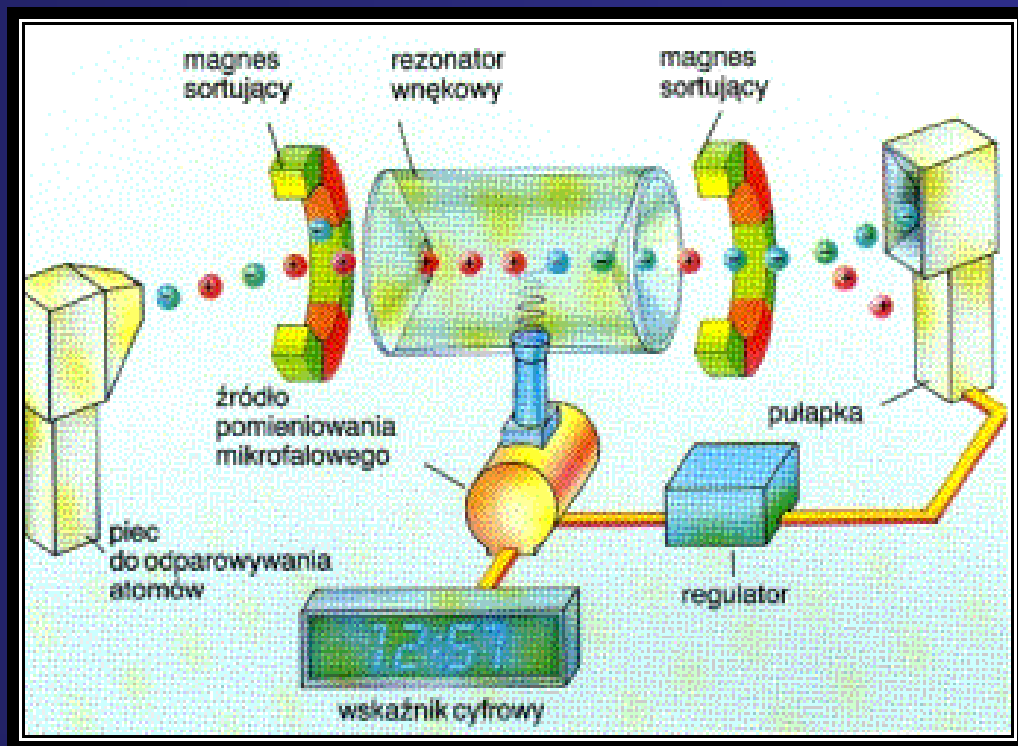
Definicje

Sekunda (od łac. *secunda* - następna, najbliższa) - jednostka czasu, jednostka podstawowa większości układów jednostek miar np. SI, MKS, CGS - oznaczenie s. Termin sekunda pochodzi z łacińskiego wyrażenia *pars minuta secunda* (druga mała część).

Jest to czas równy 9 192 631 770 okresów promieniowania odpowiadającego przejściu między dwoma poziomami $F = 3$ i $F = 4$ struktury nadsubtelnej stanu podstawowego $2S_{1/2}$ atomu cezu ^{133}Cs (powyższa definicja odnosi się do atomu cezu w spoczynku, w temperaturze 0K). Definicja ta obowiązuje od 1967 r. Definicja ta została ustalona przez Międzynarodowy Układ Jednostek Miar. Poprzednio sekundę definiowano jako $1/31\,556\,925,9747$ część roku zwrotnikowego 1900 lub $1/86400$ część doby.

Zegar atomowy

Zegar atomowy to rodzaj zegara, który używa atomowego wzorca częstotliwości jako licznika. Współcześnie najdokładniejsze zegary atomowe bazują na bardziej zaawansowanej fizyce, np. na związkach cezu. Zegar zbudowany na tej zasadzie wykazuje niedokładność rzędu 10^{-13} , czyli najwyżej jedną sekundę na milion lat.

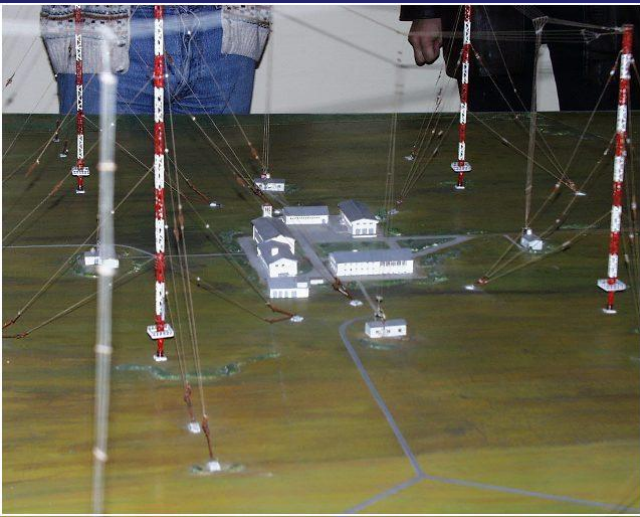


Pierwszy zegar atomowy został zbudowany w 1949 roku w amerykańskim National Bureau of Standards

Uproszczony schemat zegara atomowego

Zegar atomowy

Na świecie działa obecnie 10 takich super-zegarów. Dwa z nich pracują w Niemczech i dostarczają dokładnej informacji o czasie dla takich instytucji jak: instytuty naukowe, elektrownie atomowe, poczta, kolej, energetyka, telewizja. Informacja ta jest przesyłana na falach radiowych. Najbliższym nadajnikiem wysyłającym sygnał dokładnego czasu jest nadajnik o nazwie DCF-77 (DCF - Data Communication Frequency, 77

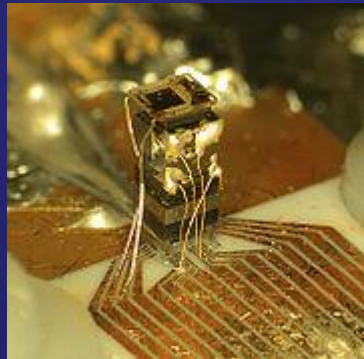


Signal zawiera kompletną informację o aktualnej dacie, czasie i typie czasu (letni /zimowy). Cała transmisja informacji przebiega w czasie 59-ciu sekund i dotyczy zawsze następnej minuty. Teoretyczny zasięg nadajnika to **2500 km** co daje pokrycie radiowe dla całej Europy. Sygnał nadawany jest na falach ultradługich o częstotliwości **77,5 kHz**. Moc nadajnika wynosi **50kW**. Nadajnik wyposażony jest w antenę główną (**150m wysokości**) i dwie rezerwowe aby zapewnić ciągłość pracy. System składa się z dwóch atomowych zegarów cezowych, jednego rubidowego, trzech nadajników i trzech anten.

Ciekawostki

W sierpniu 2004 roku, uczeni z amerykańskiego National Institute of Standards and Technology (NIST) zademonstrowali miniaturowy zegar atomowy: właściwa część zegara – komora z cezem – ma wielkość ziarna ryżu: średnicę 1,5 mm i długość 4 mm. Cały układ (komora wraz oprzyrządowaniem: diodą laserową, polaryzatorami, fotodiodą) zajmuje objętość około 1 cm³, a więc porównywalną z układami zegarów kwarcowych.

Jego dokładność jest jednak tysiąckrotnie wyższa niż zegarów kwarcowych: wynosi jedną dziesięciomiliardową (10^{-10}), co oznacza dopuszczalną odchyłkę **1 sekundy w ciągu 300 lat !!!**



Dla porównania - popularne kwarcowe zegary zwykle gubią sekundę co kilka dni.

Ciekawostki !!!



Zgodnie z teorią względności, zegar leżący na powierzchni Ziemi powinien chodzić wolniej, niż zegar umieszczony np. na szczycie wieży. Są to oczywiście minimalne różnice.

Zegary

Podsumowaniem wykładów i przeprowadzonych ćwiczeń może być zaprojektowanie i zbudowanie zegara. Elektroniczne zegary bardzo często wykorzystują mikrokontrolery, które z łatwością obsługują :

- przyciski lub klawiatury
- wyświetlacze LED i LCD
- oraz odmierzają określone odcinki czasu

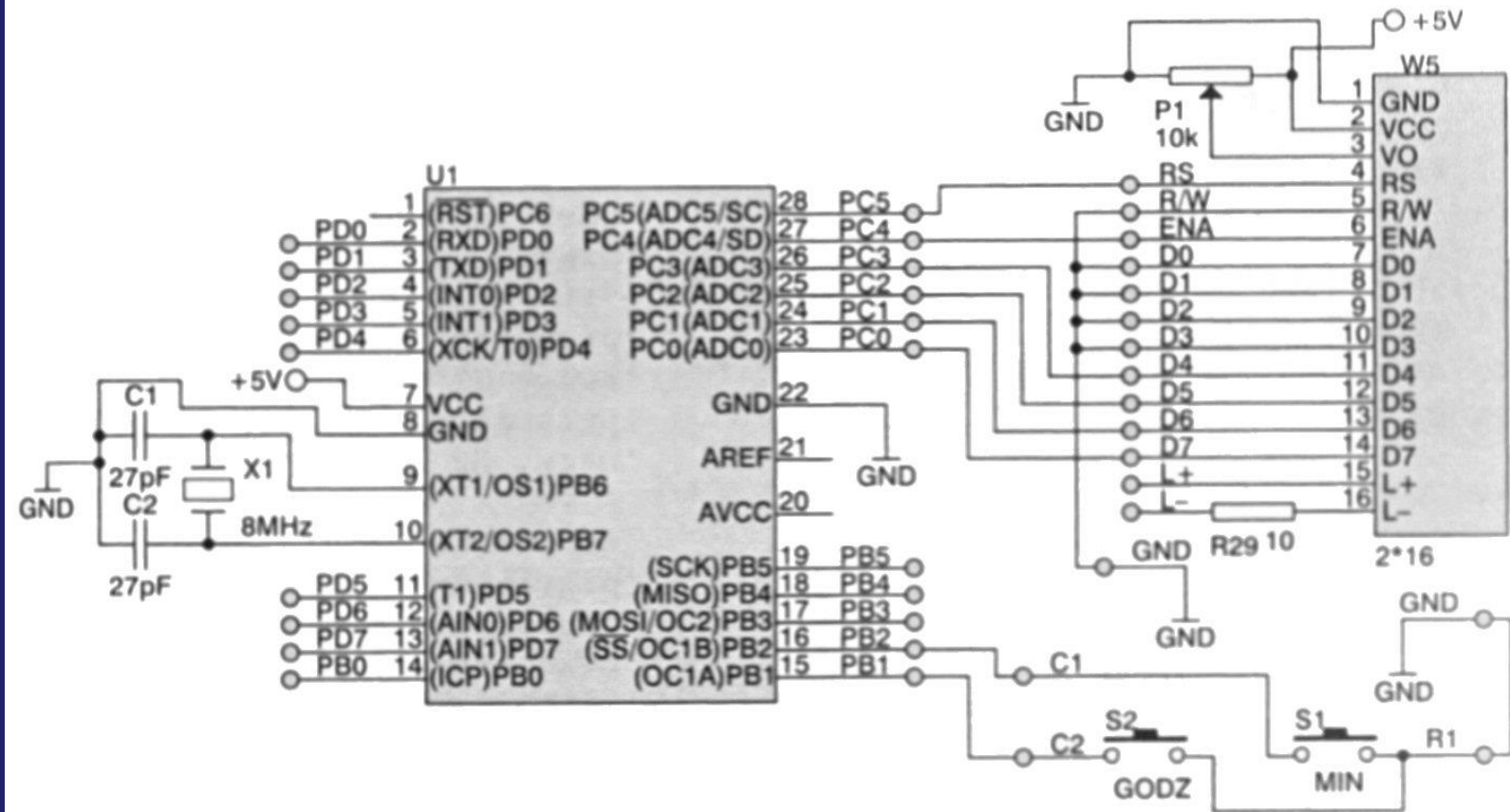
Ćwiczenia poświęcone będą budowie zegara, który jest zrealizowany w tradycyjny sposób, wykorzystując przerwania zgłaszane od przepelnienia Timera, lub wykorzystując zegar czasu rzeczywistego (RTC), który może być układem zewnętrznym, układem wbudowanym w mikrokontroler lub realizowanym programowo.

Program 16

Przykład zegara wykorzystującego
przerwania od przepełnienia licznika Timer1

Program 16

Schemat układu zegara



Program 16

```
D:\Mikroprocesory\Bascom Colege\basAVR_listingi\Dziala_zaj...
Sub
Label
$regfile = "m8def.dat"
$crystal = 8000000

Config Pinb.1 = Input
Config Pinb.2 = Input
Config Lcd = 16 * 2
Config Timer1 = Timer , Prescale = 256

Declare Sub Wysw czas
On Timer1 Odmierz 1s

Dim Sekundy As Byte
Dim Minuty As Byte
Dim Godziny As Byte
Dim Nowa_w As Bit
Dim Wart_bcd As Byte

S1 Alias Pinb.2
S2 Alias Pinb.1

Enable Interrupts
Enable Timer1
Counter1 = 34286

Set Nowa_w
Set Portb.1
Set Portb.2
```

informuje kompilator o pliku dyrektyw mikrokontrolera

informuje kompilator o częstotliwości oscylatora taktującego mikrokontroler

Konfiguracja: portów, wyświetlacza LCD, Timer1 jako timer z podziałem preskalera przez 256

procedura wyświetlająca czas na LCD w formacie gg:mm:ss

przerwanie od przepełnienia Timer1 o etykiecie Odmierz_1s

Deklaracja zmiennych. Wart_bcd - zmienna pomocnicza, w której jest zapisywana wartość BCD czasu

Przypisanie nazw portom

Odblokowanie przerwania od przepełnienia Timer1

wartość początkowa wpisywana do Timer1 ($65536 - 31250 = 34286$)

zezwozenie na wyświetlenie aktualnego czasu

dołączenie do linii PB1 rezystora podciągającego

dołączenie do linii PB2 rezystora podciągającego

Program 16

```
D:\Mikroprocesory\Bascom Colege\basAVR_listingi\Dziala_zaj...
Sub      Label
Do
  Call Wysw czas
  If S1 = 0 Then
    Waitms 25
    If S1 = 0 Then
      Incr Minuty
      Sekundy = 0
      If Minuty = 60 Then
        Minuty = 0
      End If
      Set Nowa_w
      Call Wysw czas
      Waitms 200
    End If
  End If
  If S2 = 0 Then
    Waitms 25
    If S2 = 0 Then
      Incr Godziny
      If Godziny = 24 Then
        Godziny = 0
      End If
      Set Nowa_w
      Call Wysw_czas
      Waitms 200
    End If
  End If
Loop
End
```

wywołanie procedury wyświetlającej aktualny czas

jeśli przyciśnięty S1, to:

opóźnienie dla eliminacji drgań styków

sprawdzenie, czy przycisk S1 dalej naciśnięty, jeśli tak, to:

zwiększenie wartości minut wyzerowanie zmiennej Sekundy

jeśli wartość Minuty = 60, to zerowanie wartości Minuty

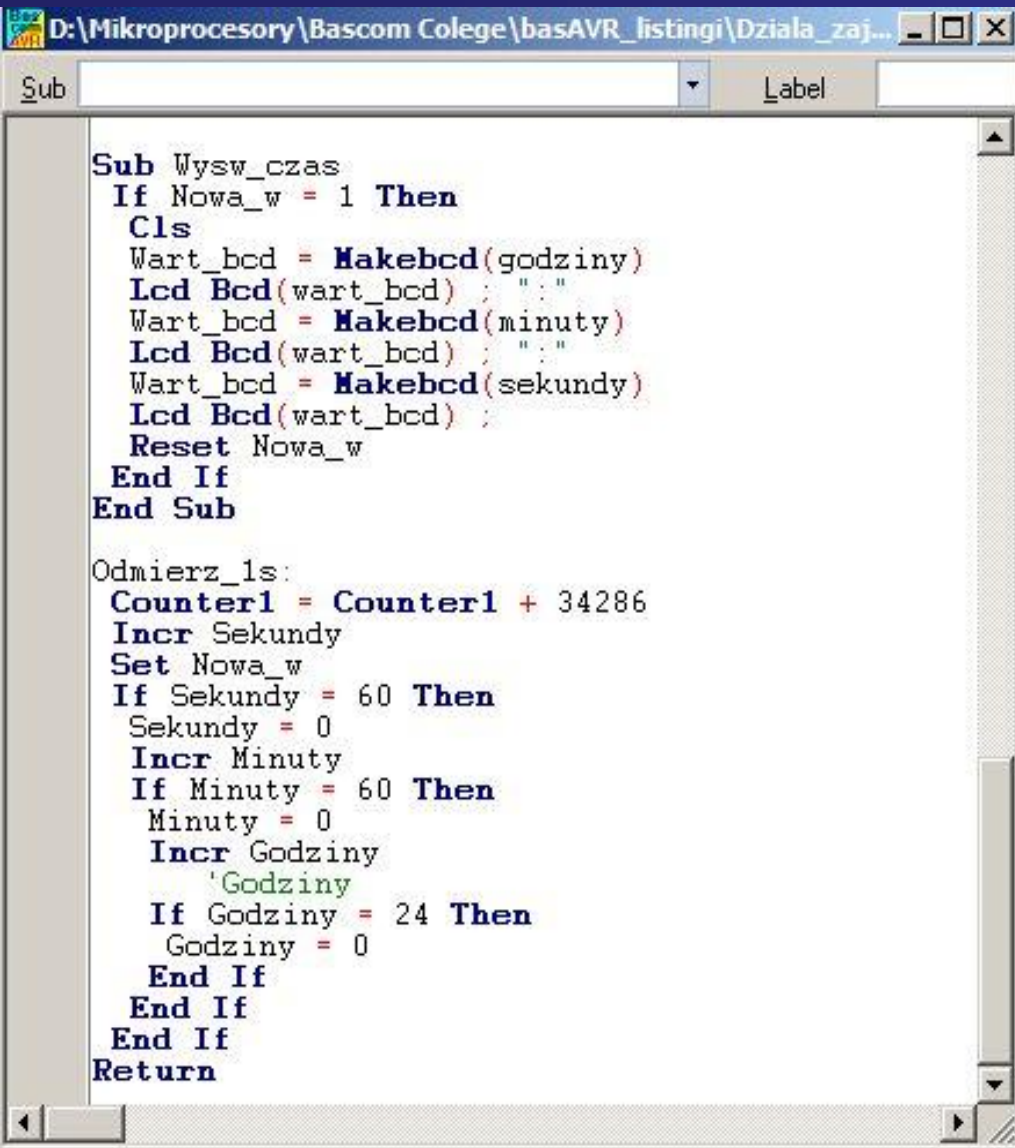
zezwole nie na wyświetlenie czasu

wywołanie procedury wyświetlenia aktualnego czasu

opóźnienie 200 ms oraz koniec instrukcji warunkowej

koniec programu

Program 16



```
D:\Mikroprocesory\Bascom Colege\basAVR_listingi\Dziala_zaj...
Sub
Label
Sub Wysw_czas
  If Nowa_w = 1 Then
    Cls
    Wart_bcd = Makebcd(godziny)
    Lcd Bcd(wart_bcd) ; ":"
    Wart_bcd = Makebcd(minuty)
    Lcd Bcd(wart_bcd) ; ":"
    Wart_bcd = Makebcd(sekundy)
    Lcd Bcd(wart_bcd) ;
    Reset Nowa_w
  End If
End Sub

Odmierz_1s:
  Counter1 = Counter1 + 34286
  Incr Sekundy
  Set Nowa_w
  If Sekundy = 60 Then
    Sekundy = 0
    Incr Minuty
    If Minuty = 60 Then
      Minuty = 0
      Incr Godziny
      'Godziny
      If Godziny = 24 Then
        Godziny = 0
      End If
    End If
  End If
  End If
  Return
```

Program 16

Zegar powinien wyświetlać czas w formacie :

01 : 03 : 19

Wyświetlając bezpośrednio zmienne *Sekundy*, *Minuty*, *Godziny* na wyświetlaczu LCD otrzymalibyśmy format:

1 : 3 : 19

Zastosowanie pomocniczej zmiennej *Wart_bcd* umożliwia przekształcenie wartości zmiętych pamiętających czas (*Sekundy*, *Minuty*, *Godziny*) z kodu dwójkowego do dwucyfrowego kodu BCD. Efektem tego przekształcenia będzie wyświetlenie zmiennych mniejszych od „10” poprzedzonych „0”

2 ⇒ 02

Program 16

Funkcja Makebcd ()

Składnia:

`zmienna1 = Makebcd(zmienna2)`

gdzie:

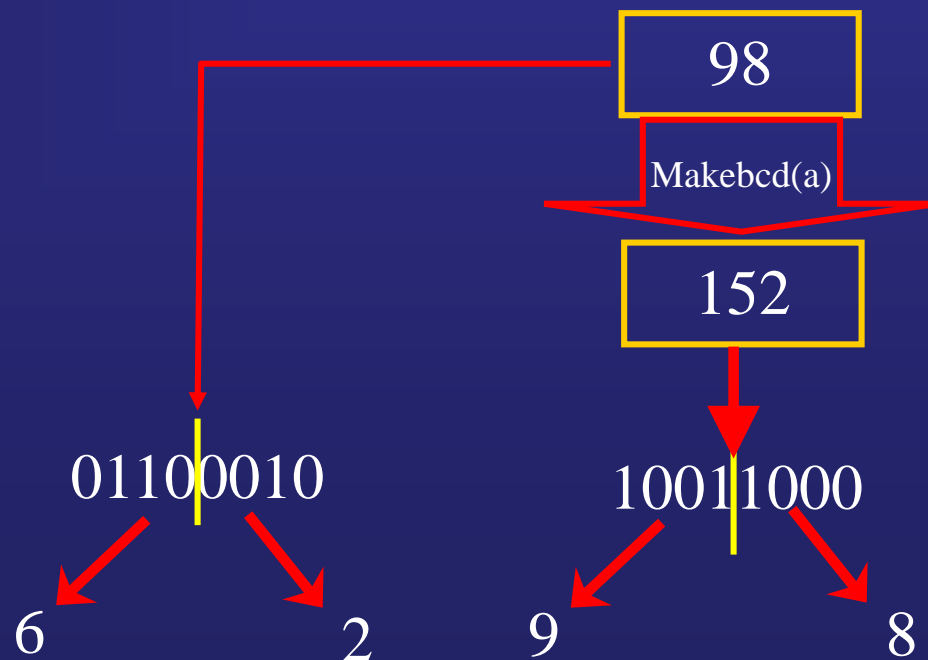
`zmienna1` zmienna, do której zapisany będzie rezultat działania funkcji,

`zmienna2` zmienna zawierająca liczbę dziesiętną.

Przykład:

a = 98

b = Makebcd(a)



Program 16

```
D:\Mikroprocesory\Bascom Colege\basAVR_listingi\Dziala_zaj...
Sub
Label

Sub Wvsw czas
If Nowa_w = 1 Then
Cls
Wart_bcd = Makebcd(godziny)
Lcd Bcd(wart_bcd) ; ":"
Wart_bcd = Makebcd(minuty)
Lcd Bcd(wart_bcd) ; ":"
Wart_bcd = Makebcd(sekundy)
Lcd Bcd(wart_bcd) ;
Reset Nowa_w
End If
End Sub

Odmierz 1s:
Counter1 = Counter1 + 34286
Incr Sekundy
Set Nowa_w
If Sekundy = 60 Then
Sekundy = 0
Incr Minuty
If Minuty = 60 Then
Minuty = 0
Incr Godziny
'Godziny
If Godziny = 24 Then
Godziny = 0
End If
End If
End If
Return
```

procedura wyświetlająca czas

jeśli flaga Nowa_w = 1, to

zamiana wartości Godziny na wartość w kodzie BCD

wyświetlenie wartości godzin w kodzie BCD

zamienionych na tekst oraz znaku :

zerowanie flagi Nowa_w

koniec procedury

podprogramu obsługi przerwania przepelnienia od
Timer1

wpisanie wartości początkowej licznika

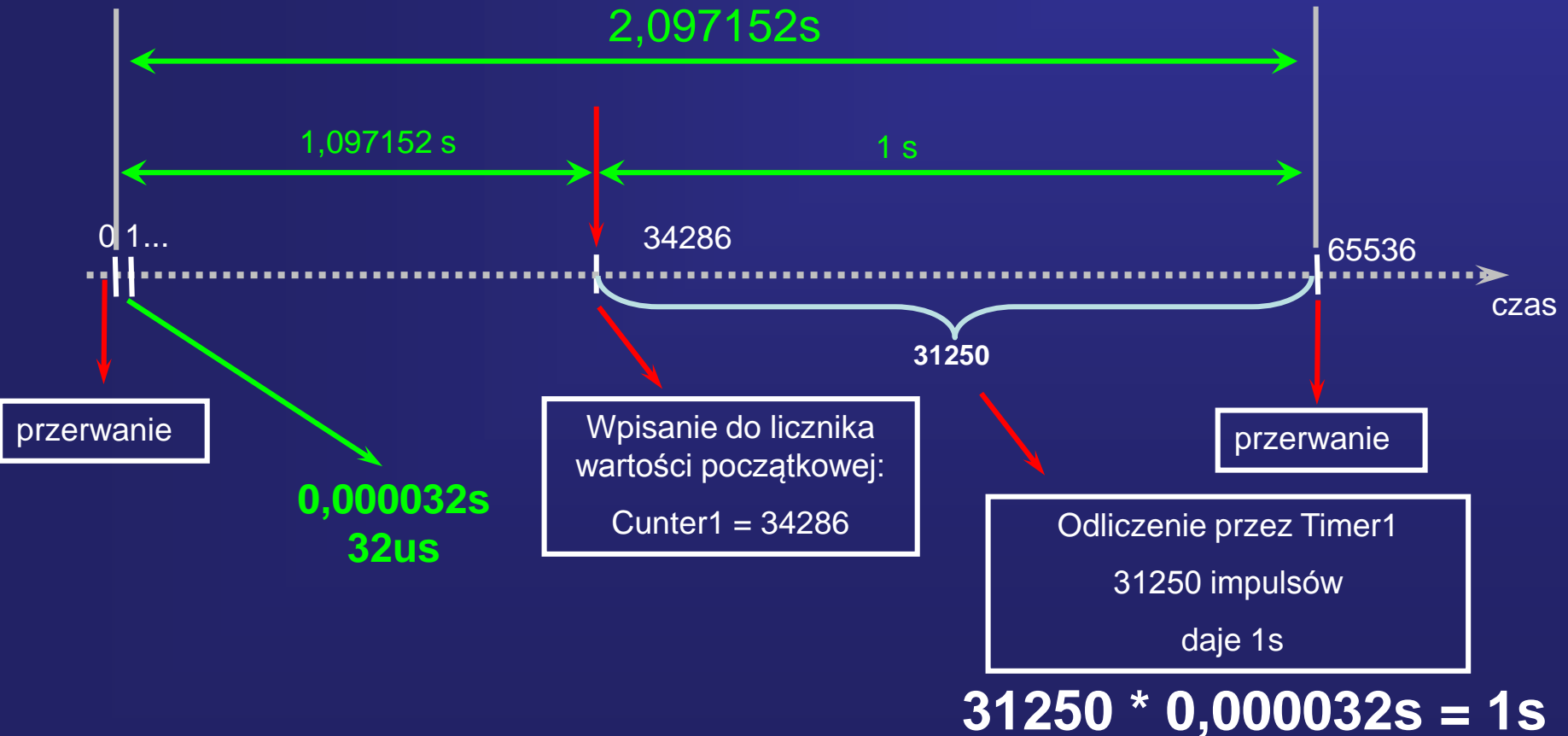
Odliczanie sekund, minut, godzin w pętlach
warunkowych

Program 16

Timer1 16 bit

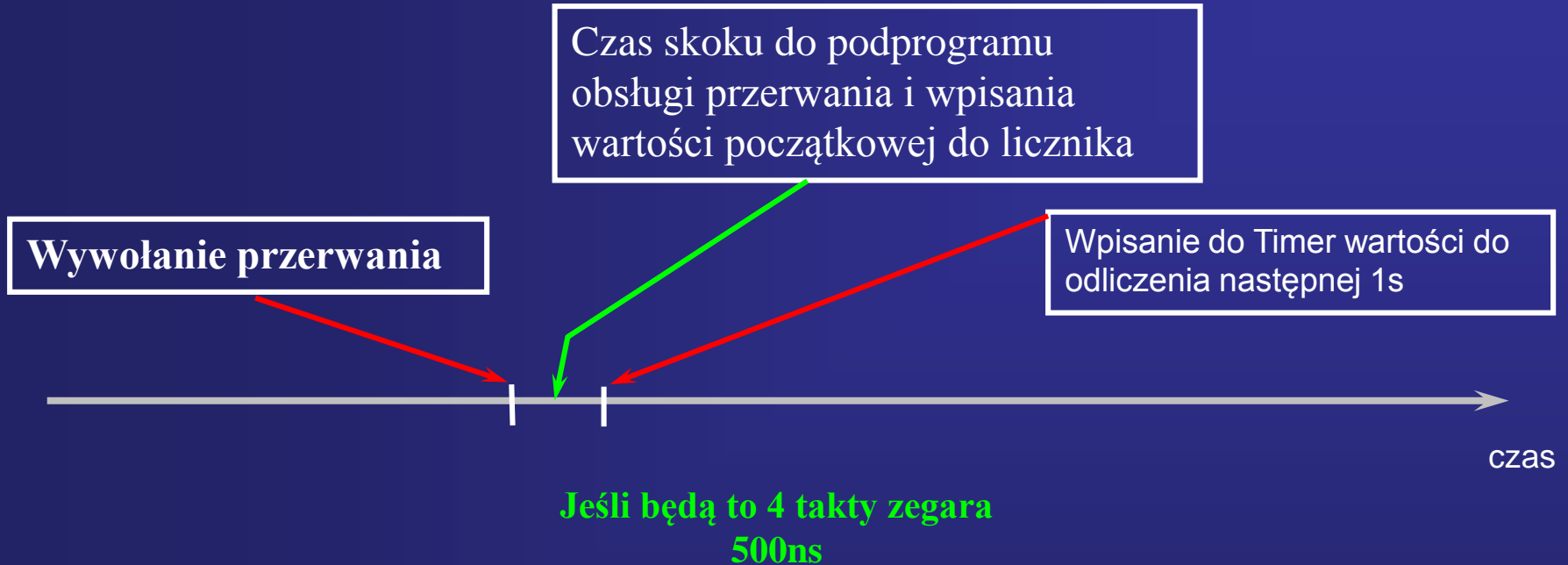
$$8\text{MHz}/256 = 31250\text{Hz}$$

$$0,000032\text{ s}$$



Program 16

Zwiększenie dokładności odmierzanego czasu



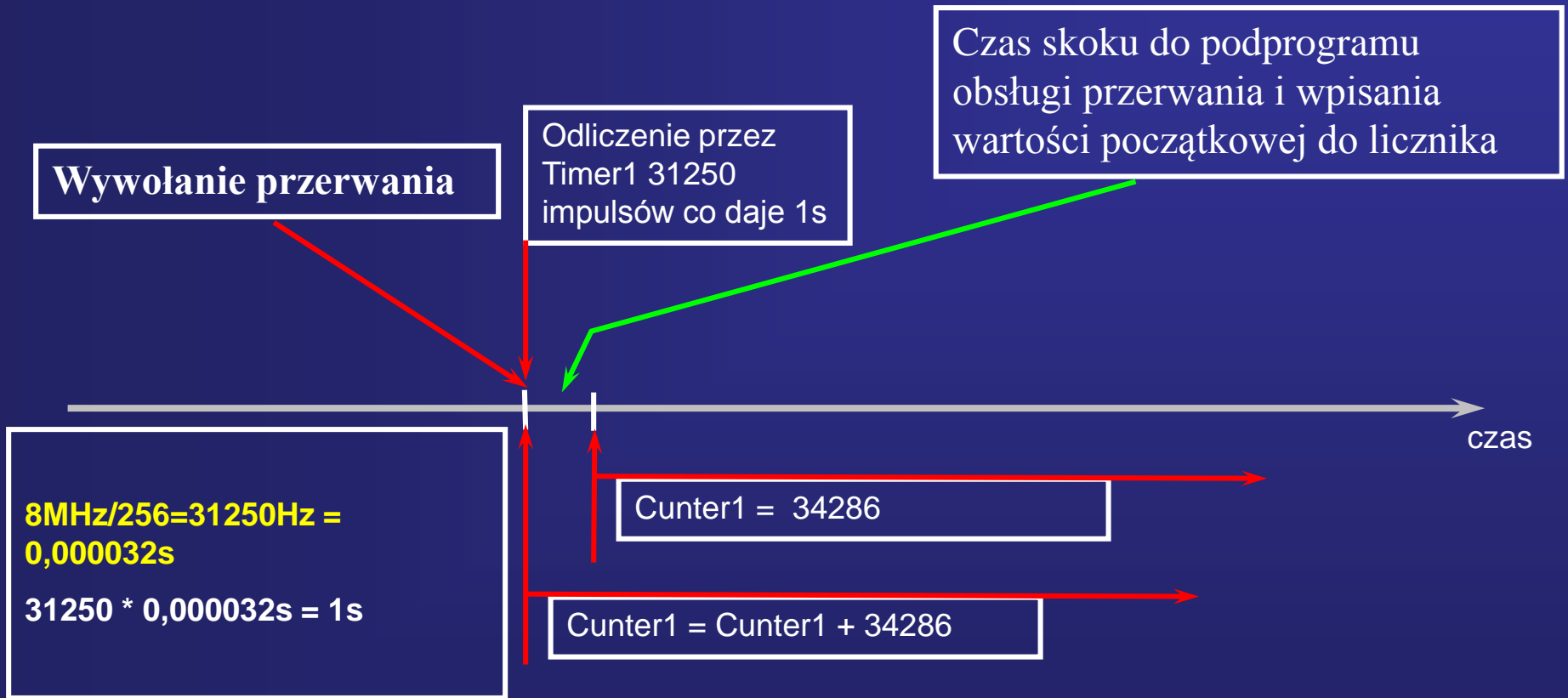
To odmierzymy więcej niż 1s

1min	1,0000005s	0.00003s
1h		0,0018s
24h		0,0432s
1M		1,296s

Program 16

Zwiększenie dokładności odmierzanego czasu

Dlaczego używamy komendy $\text{Cunter1} = \text{Cunter1} + 34286$



Program 17

Przykład zegara czasu rzeczywistego RTC

Program 17

Zegar czasu rzeczywistego RTC

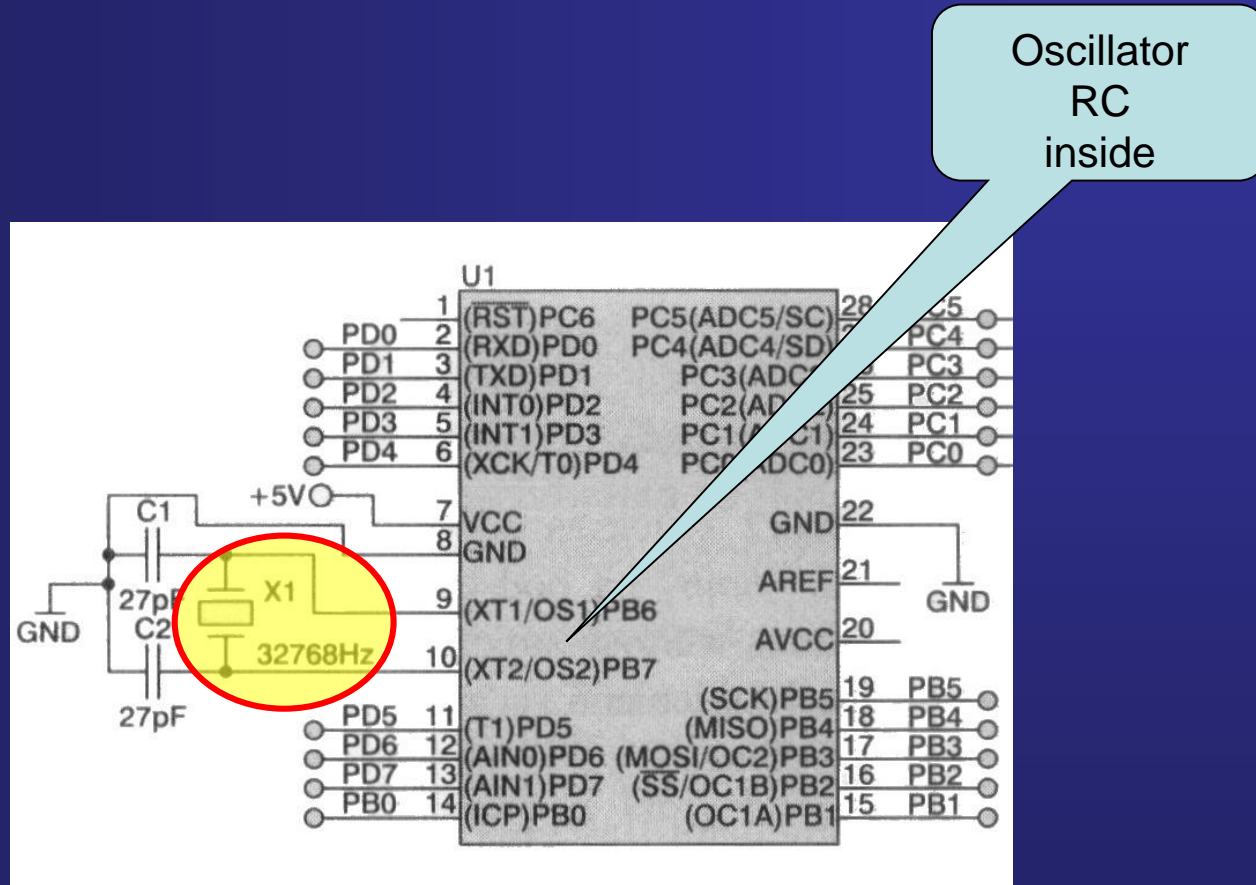
W Bascomie są instrukcje, które przy wykorzystaniu asynchronicznej pracy licznika Timer2 umożliwiają w prosty sposób zrealizowanie programowego zegara czasu rzeczywistego (RTC). Mikrokontroler powinien być wyposażony w timer, który może pracować w trybie asynchronicznym. To znaczy, że może być taktowany sygnałem innym niż sam mikrokontroler.

Dołączając do mikrokontrolera oscylator, którego sygnałem będzie taktowany Timer2 „zegarkowy” rezonator kwarcowy o częstotliwości 23768 Hz, można w prosty sposób odmierzać odcinki czasu równe 1 sekundę.

$$2^{15} = 23768$$

(Łatwo można uzyskać 1s, ponieważ częstotliwość 23768 Hz jest podzielna przez dwa dzieląc 23768 przez 2 piętnaście razy otrzymamy 1)

Schemat układu mikrokontrolera z dołączonym rezonatorem 23768 Hz



Program 17

Instrukcja *Config Clock*

Składnia:

`Config Clock = Soft | User [, Gosub = Sectic]`

Opis:

Pierwszy parametr `CLOCK=` definiuje sposób odmierzania czasu przez mikrokontroler. Przy podaniu `SOFT`, zliczaniem czasu zajmuje się specjalna procedura obsługi przerwania. Określenie tego parametru jako `USER` pozwala na utworzenie procedury odmierzania czasu przez użytkownika. Może on skorzystać z rozwiązania programowego lub sprzętowego, np. wykorzystując scalony zegar RTC.

Drugi parametr `GOSUB = SECTIC` pozwala na skonstruowanie przez użytkownika dodatkowego podprogramu, który będzie wywoływany co sekundę. Ważne jest by podprogram miał nazwę (rozpoczynał się etykietą) `SECTIC` i był zakończony instrukcją `Return`.

Jeśli w programie użyto instrukcji `CONFIG CLOCK`, kompilator automatycznie generuje specjalne zmienne nazwane: `_sec` , `_min` , `_hour` , `_day` , `_month` , `_year`.

W ten sam sposób są definiowane specjalne zmienne `TIMES$` oraz `DATE$`. Jest w nich pamiętany zliczany czas i data, w postaci tekstowej.

Zmienne `_sec`, `_min` `_hour`, `_day` , `_month` , `_year`, mogą być odczytywane i modyfikowane przez program. Oczywiście zmiana jednej z nich powoduje także zmianę zmiennych specjalnych `DATE$` i `TIMES$`.

Program 17

Instrukcja *Config Date*

Składnia:

Config Date = Dmy | Mdy | Ymd, Separator = znak_oddzielający

gdzie:

znak_oddzielający - znak oddzielający poszczególne składniki daty. Można używać: / , - (minus) i . (kropka).

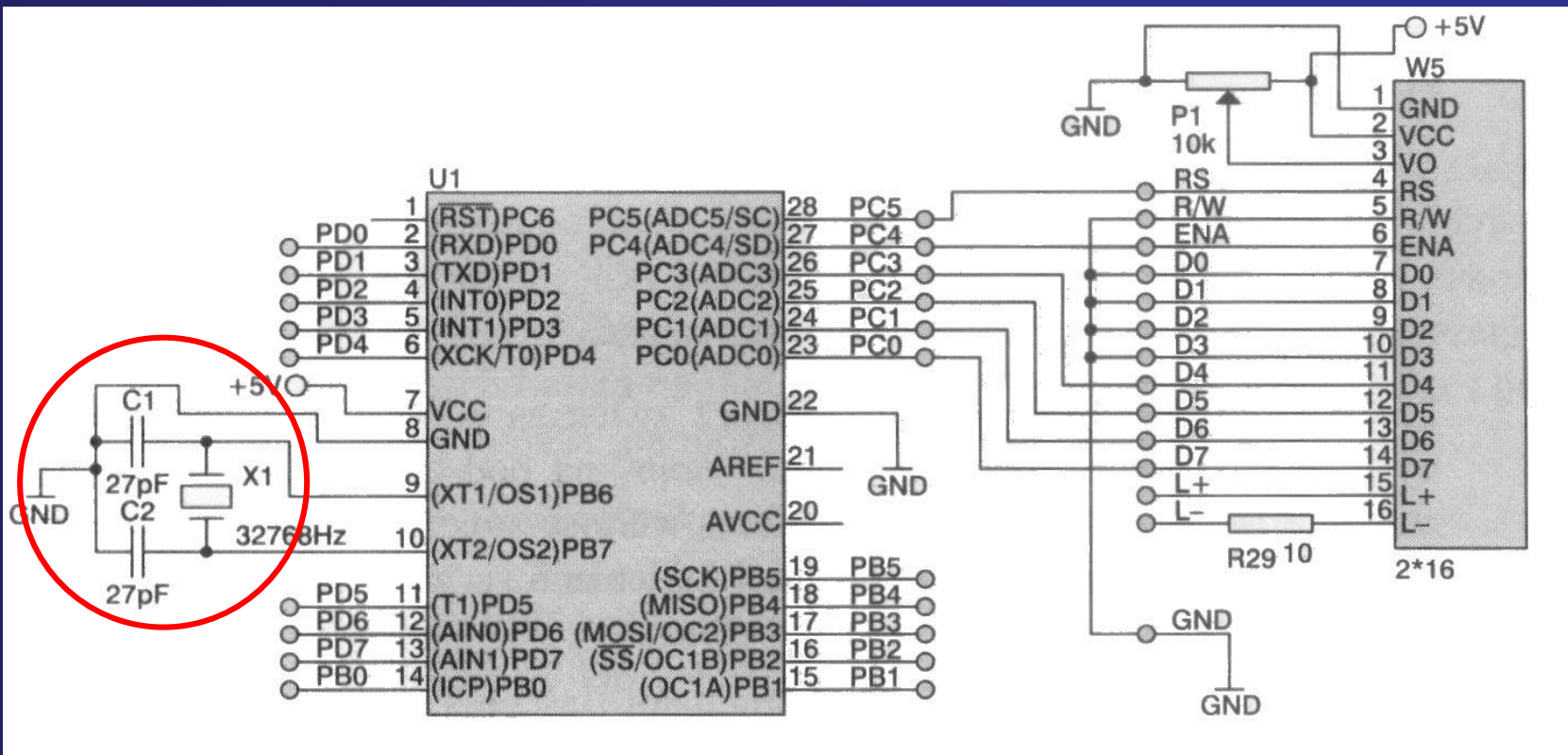
Opis:

Poniższa tabela pokazuje kilka stosowanych formatów przedstawiających datę.

Kraj/Norma	Format	Postać instrukcji
Ameryka	mm/dd/yy	Config Date = MDY, Separator = /
ANSI	yy.mm.dd	Config Date = YMD, Separator = .
W. Brytania/Francja	dd/mm/yy	Config Date = DMY, Separator = /
Niemcy	dd.mm.yy	Config Date = DMY, Separator = .
Włochy	dd-mm-yy	Config Date = DMY, Separator = -
Japonia/Taiwan	yy/mm/dd	yy/mm/dd Config Date = YMD, Separator = /
USA	mm-dd-yy	Config Date = MDY, Separator = -

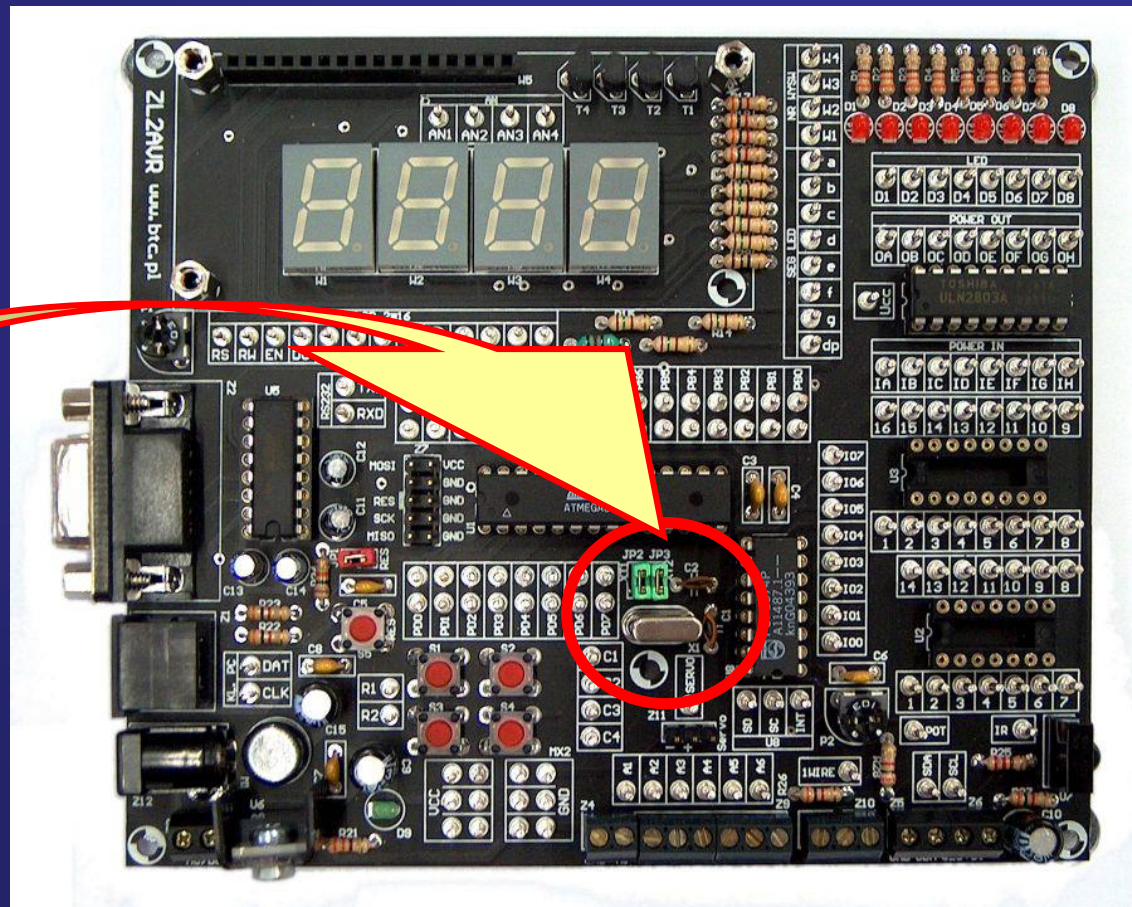
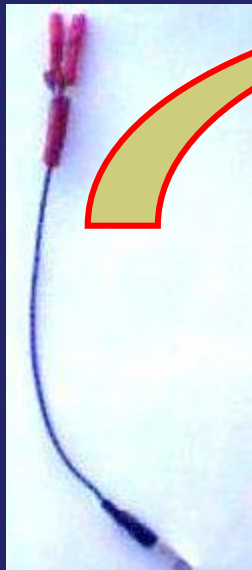
Program 17

Schemat układu mikrokontrolera z dołączonym rezonatorem 23768 Hz



Program 17

Dołączenie rezonatora kwarcowego 23768 Hz



Program 17

```
D:\Mikroprocesory\Bascom Colege\basAVR_listingi...
Sub                               Label
$regfile = "m8def.dat"
$crystal = 8000000

Config Lcd = 16 * 2
Config Clock = Soft
Config Date = Mdy , Separator = /

Enable Interrupts

Date$ = "08/22/04"
Time$ = "23:59:00"

Do
  Cls
  Lcd Time$

  Lowerline

  Lcd Date$

  Waitms 100
Loop
End
```

informuje kompilator o pliku dyrektyw mikrokontrolera

informuje kompilator o częstotliwości oscylatora taktującego mikrokontroler

konfiguracja typu wyświetlacza LCD

konfiguracja zegara RTC

konfiguracja sposobu formatowania daty
odblokowanie przerwań globalnych

zapis daty do zmiennej systemowej Date\$
przechowującej datę

zapis czasu do zmiennej systemowej Time\$
przechowującej czas

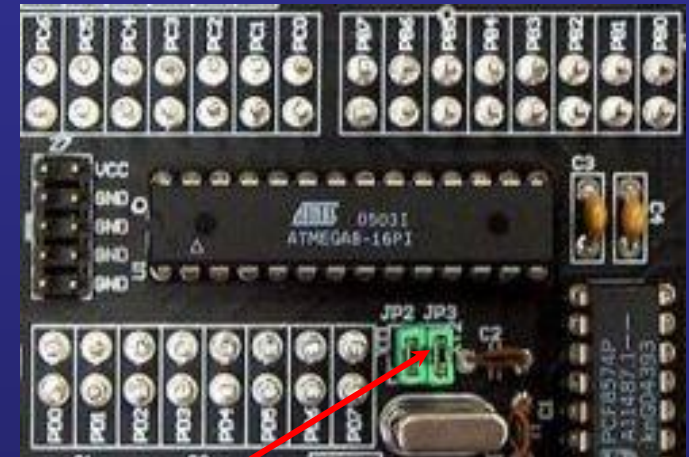
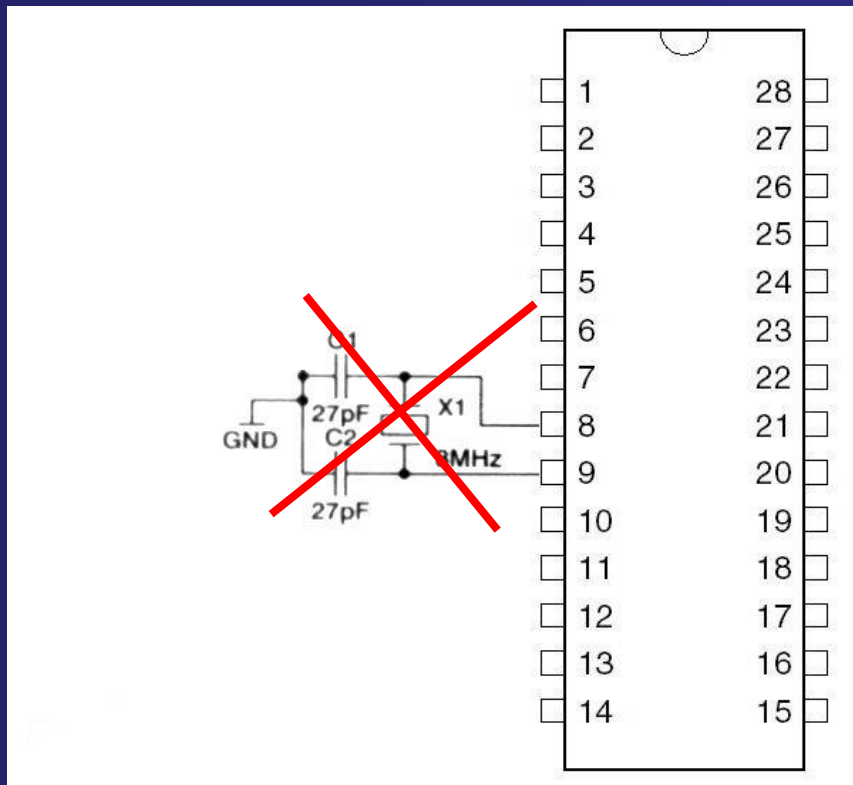
wyświetlenie na LCD w pierwszej linii aktualnego czasu
przejście kursora do drugiej linii

wyświetlenie w drugiej linii LCD aktualnej daty
opóźnienie 100 ms

koniec programu

Program 17

Mikrokontroler jest taktowany wewnętrznym oscylatorem RC o częstotliwości 8MHz



Usunąć zwoje JP2, JP3

Konfiguracja *Fuse bits* w celu uaktywnienia wewnętrznego oscylatora RC

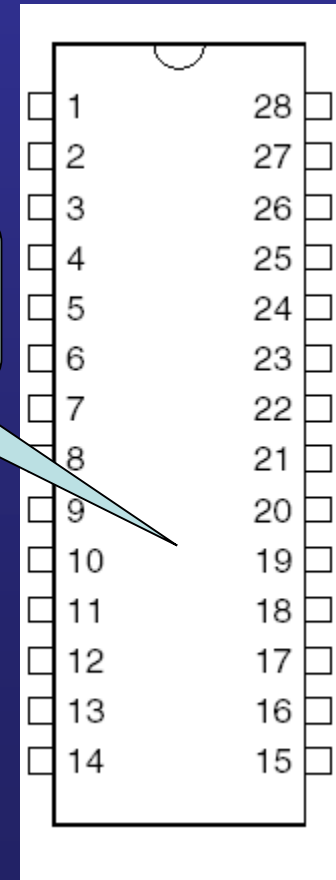
Table 9. Internal Calibrated RC Oscillator Operating Modes

CKSEL3..0	Nominal Frequency (MHz)
0001 ⁽¹⁾	1.0
0010	2.0
0011	4.0
0100	8.0

Note: 1. The device is shipped with this option selected.

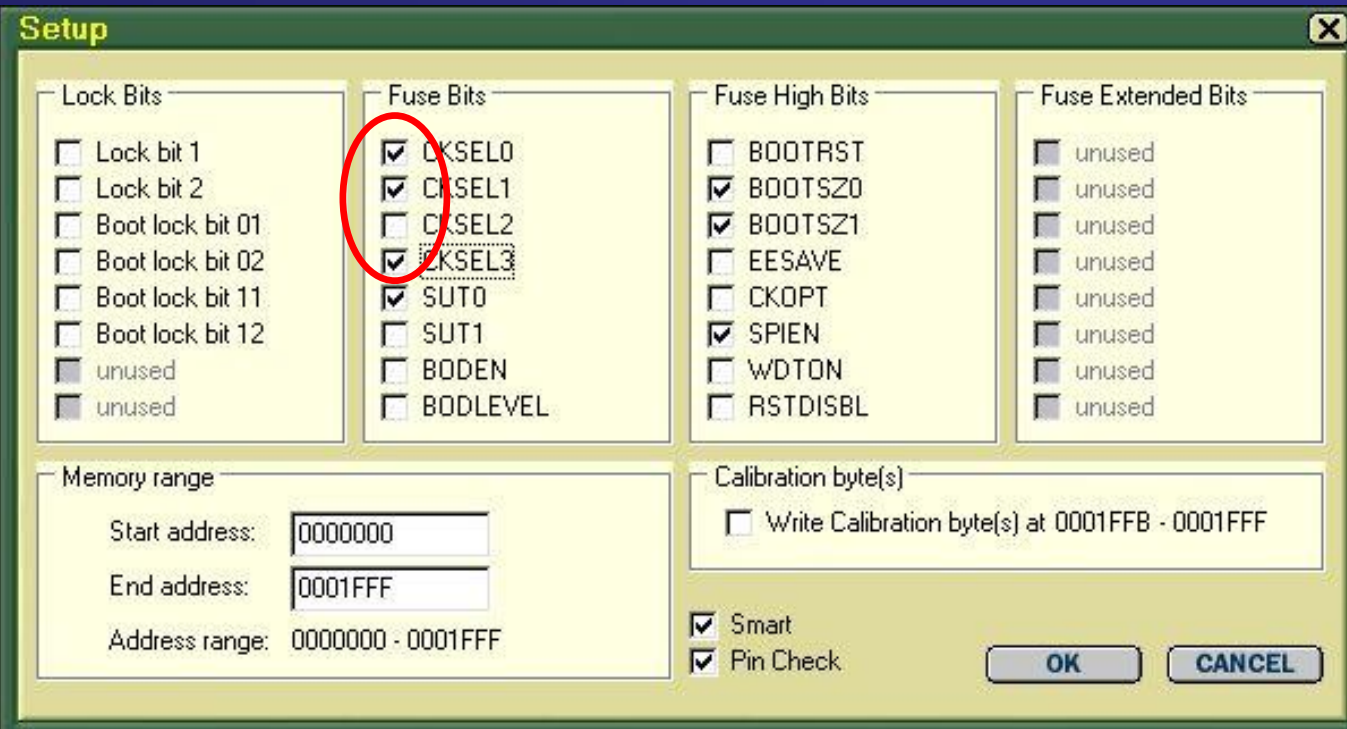
Nr bitu	Nazwa Fuse Bitu	Wartość Fuse Bitu
0	CKSEL 0	0
1	CKSEL 1	0
2	CKSEL 2	1
3	CKSEL 3	0
4	SUTO 0	0
5	SUTO 1	1
6	BODEN	1
7	BODLEVEL	1

Oscillator RC inside



Program 17

Programowanie Zapisanie odpowiedniej konfiguracji Fuse Bits



Setup

Lock Bits	Fuse Bits	Fuse High Bits	Fuse Extended Bits
<input type="checkbox"/> Lock bit 1	<input checked="" type="checkbox"/> CKSEL0	<input type="checkbox"/> BOOTRST	<input type="checkbox"/> unused
<input type="checkbox"/> Lock bit 2	<input checked="" type="checkbox"/> CKSEL1	<input checked="" type="checkbox"/> BOOTSZ0	<input type="checkbox"/> unused
<input type="checkbox"/> Boot lock bit 01	<input type="checkbox"/> CKSEL2	<input checked="" type="checkbox"/> BOOTSZ1	<input type="checkbox"/> unused
<input type="checkbox"/> Boot lock bit 02	<input checked="" type="checkbox"/> CKSEL3	<input type="checkbox"/> EESAVE	<input type="checkbox"/> unused
<input type="checkbox"/> Boot lock bit 11	<input checked="" type="checkbox"/> SUT0	<input type="checkbox"/> CKOPT	<input type="checkbox"/> unused
<input type="checkbox"/> Boot lock bit 12	<input type="checkbox"/> SUT1	<input checked="" type="checkbox"/> SPIEN	<input type="checkbox"/> unused
<input type="checkbox"/> unused	<input type="checkbox"/> BODEN	<input type="checkbox"/> WDTON	<input type="checkbox"/> unused
<input type="checkbox"/> unused	<input type="checkbox"/> BODLEVEL	<input type="checkbox"/> RSTDISBL	<input type="checkbox"/> unused

Memory range

Start address:

End address:

Address range: 00000000 - 0001FFF

Calibration byte(s)

Write Calibration byte(s) at 0001FFB - 0001FFF

Smart
 Pin Check

OK **CANCEL**

Zadanie specjalne !!!

Na podstawie znajomości obsługi klawiatury oraz sterowania multipleksowego wyświetlaczem LED, napisać program zegara wyświetlającego czas w postaci (godziny i minuty) na czterech wyświetlaczach LED

