|  |  |  |  |
| --- | --- | --- | --- |
| Systemy wbudowane Laboratorium | | | |
| **Grupa:**  C | **Temat:**  Zastosowanie licznika-czasomierza. | | |
| **Data:**  22.04.2021r. | **Wykonała:**  Eliza Mrówczyńska | | |
| **Godzina:**  12:40 | **II rok**  **Informatyk Stosowana** | **Ocena i uwagi prowadzącego:** | **Prowadzący:**  dr hab. Witold Kozłowski |

## Opis zadania

Celem ćwiczenia jest zastosowanie licznika-czasomierza Timer() do generowania stałych odcinków czasu, oraz wygenerowanie danego sygnału PWM przy użyciu Timer1.

Każdy mikrokontroler jest wyposażony w co najmniej jeden timer, który można zazwyczaj skonfigurować do pracy w trybach licznika, czasomierza czy generatora PWM.

Licznik Timer0 jest 8-bitowy, a więc może zliczyć 256 impulsów. Timer1 jest licznikiem 16-bitowym, więc może zliczyć 65 536 impulsów.

## Instrukcje potrzebne do konfiguracji i sterowania licznikiem

**Config** Timer0 Konfiguracja pracy licznika

**Start** Sterowanie licznikiem

**Stop** Sterowanie licznikiem

**Counter0 = wart\_poczatkowa** Do licznika Timer0 zostaje wpisana wartość początkowa

wart\_poczatkowa

**ON INTERRUPT** Obsługa przerwań

**ENABLE** Włączenie zgłaszania przerwań

**DISABLE** Wyłączenie zgłaszania przerwań

## Odmierzanie 1s – Timer0

Kod programu, który ma odmierzać 1s:

**$regfile** = "m8def.dat" Informuje kompilator o pliku dyrektyw mikroprocesora

**$crystal** = 8000000 Informuje kompilator o częstotliwości oscylatora taktującego

mikrokontroler

**Config** Pinb.0 = **Output** Ustawiamy PB0 jako wyjście

**Config** Timer0 = **TIMER**, Prescale = 256 Konfiguracja Timer0 jako timera z podziałem prescalera przez

256

**On** Timer0 Odmierz\_1s Przerwanie od przepełnienia Timer0 o etykiecie Odmierz\_1s

**Dim** Licz\_8ms **As Byte** Zmienna pomocnicza, która zlicza odcinki czasu równe 8ms

**Enable Interrupts** Włączenie globalnego systemu przerwań

**Enable** Timer0 Włączenie przerwania od przepełnienia Timer0

**Load** Timer0 = 250 Wpisanie wartości początkowej do Timer0

**Do** Nieskończona pętla

**Loop**

**End** Koniec programu

Odmierz\_1s: Początek podprogramu, uruchamiany jest, gdy wystąpi

przerwanie przepełnienia Timer0

**Load** Timer0 =250 Wpisanie wartości początkowej 6 do Timer0

**Incr** Licz\_8ms Zwiększenie o 1 zmiennej pomocniczej Licz\_8ms

**If** Licz\_8ms = 125 **Then** Jeśli wartość zmiennej pomocniczej wynosi 125 (125\*8ms = 1s)

to odliczono 1s

Licz\_8ms = 0 Zerowanie zmiennej pomocniczej

**Toggle** Portb.0 Zmień na przeciwny stan linii PB0

**End If** Koniec warunku

**Return** Powrót z przerwania

Uruchomienie programu dla układu z wewnętrznym rezonatorem RC skutkuje ukazaniem się na oscyloskopie czasu **968ms** zamiast odmierzanej 1s. Różnica wydaje się niewielka jednak w skali roku jest to prawie 12 dni różnicy! Aby wyeliminować ten błąd powinniśmy użyć zewnętrznego rezonatora kwarcowego, wtedy odmierzana będzie dokładnie **1s**.

## Generowanie sygnału PWM

Do generowania sygnału PWM potrzebujemy obliczyć częstotliwość, co robimy za pomocą jednego ze wzorów:

|  |  |  |
| --- | --- | --- |
| **Rozdzielczość PWM** | **Wartość max. licznika** | **Częstotliwość** |
| 8 | 255 | Fc/Prescaler/510 |
| 9 | 511 | Fc/Prescaler/1022 |
| 10 | 1023 | Fc/Prescaler/2046 |

Wypełnienie obliczamy ze wzoru:

|  |  |  |
| --- | --- | --- |
| **Rozdzielczość PWM** | **Wartość max. licznika** | **Wypełnienie** |
| 8 | 255 | 255 – 100%  x - 10% |
| 9 | 511 | 510 – 100%  x - 10% |
| 10 | 1023 | 1023 – 100%  x - 10% |

Dane, dla których będzie generowany sygnał PWM:

f = 61,27Hz

wypełnienie 50%

**Obliczenia:**

prescale = 256

pwm = 8

fk = 8000000/256/510 = 61,27Hz

Wypełnienie:

255 - 100%

x - 50%

100%x = 12 750%

x = 127,5

#### Kod generujący dany sygnał PWM

**$regfile** = "m8def.dat" Informuje kompilator o pliku dyrektyw mikroprocesora

**$crystal** = 8000000 Informuje kompilator o częstotliwości oscylatora taktującego

mikrokontroler

**Config** Pinb.1 = **output** Linia PB1 jako wyjście

**Config Timer1** = PWM, PWM = 8, Compare A PWM = **Clear Up**, Compare B PWM = **Disconnect**, Prescale = 256

Konfiguracja Timer1 jako generatora sygnału PWM

**PWM1a** = 127.5 Wpisanie do zmiennej PWM1a wartości określającej wypełnienie

sygnału na wyjściu

**End** Koniec programu

## Podsumowanie

Każdy mikrokontroler wyposażony jest w co najmniej jeden timer, który można skonfigurować do pracy w różnych trybach:

* Licznika,
* Czasomierza,
* PWM.

Zaletą timerów jest fakt, iż mogą pracować niezależnie od innych bloków funkcjonalnych mikrokontrolera.

Aby mieć pewność, że timer działa poprawnie należy podłączyć zewnętrzny rezonator kwarcowy, gdyż jego dokładność jest większa niż wewnętrznego rezonatora RC. Na przykładzie odmierzania czasu 1s widzieliśmy, że używając wewnętrznego rezonatora RC różnica czasowa wynosi prawie **12 dni**!