

Wydział Fizyki, Matematyki i Informatyki
Politechnika Krakowska



INSTYTUT TELEINFORMATYKI

Systemy Wbudowane

Sprawozdanie z Laboratorium 2

Prowadzący:

dr hab. Zbysław Tabor
prof. dr hab. inż. Piotr
Malecki

Autor:

Kamil Wójcik

1 Wprowadzenie

Celem laboratorium było zapoznanie się z systemem przerwań w układach mikrokontrolerów, na przykładzie zestawu ZL15AVR wyposażonego w AVR ATmega32.

2 Zadanie

Celem wybranego zadania(2.2) było napisanie programu odliczającego sekundy na wyświetlaczu 7-segmentowym wykorzystując przerwania od Timera0.

2.1 Obsługa przerwań

Zgodnie z wymaganiem do odliczania czasu użyjemy przerwań od Timera0. Zastosowany zestaw uruchomieniowy posiada wbudowany oscylator o $f = 1MHz$. Zakładając najwyższą wartość preskalera $T = 1024$ oraz fakt że użyjemy przerwań od przepełnienia rejestru możemy określić że potrzebna liczba przerwań do odliczenia 1s będzie wynosić ≈ 3.81 . Aby rozwiązać ten problem wprowadzimy dodatkową zmienną która będzie określała liczbę przerwań.

$$C = \frac{f_{clk}}{f_{czas} \cdot RT}$$

gdzie:

C - licznik przerwań

f_{clk} - częstotliwość pracy oscylatora (1MHz)

f_{czas} - częstotliwość pożądana (1Hz)

R - ilość cykli timera do przerwania (255). W naszym przypadku przerwania ustawione są od przepełnienia a Timer0 jest timerem 8-bitowym

T - wartość preskalera(64)

Podstawiając do wzoru otrzymujemy: $C = 61,2745$. Łatwo można obliczyć że czas odliczany tym sposobem będzie wynosił $\approx 0.9955s$. Na takiej precyzji poprzestaniemy. Wymienię tylko sposoby jakimi można osiągnąć większą precyzję.

- Wprowadzenie kolejnej zmiennej pozwalającej dokładniej zliczać przerwania przy mniejszej wartości preskalera
- Zastosowanie zegara czasu rzeczywistego

W bloku obsługi przerwania pomniejszamy licznik przerwań. W przypadku gdy limit został osiągnięty wyświetlana cyfra zostaje zwiększona a w razie przepełnienia wyzerowana. Po zwiększeniu cyfry licznik przerwań jest resetowany.

2.2 Obsługa wyświetlacza

Do sterowania wyświetlaczem posłużymy się portem A. W tym celu ustawiamy wartość rejestru DDRA w stan wysoki(0xFF). W celu konwersji wartości cyfr do wartości sterującej zapalonymi segmentami wyświetlacza definiujemy tablice *digitValues* jej wartości ustawiamy w funkcji *init*. Przy takim założeniu piny segmentów powinny być podłączone w kolejności *gfedcba* począwszy od pierwszego pinu Portu A. Do ostatniego pinu portu A podłączamy pin wyboru cyfry wyświetlacza oraz ustawiamy wyjście w stan wysoki (0x80).

```
1  /*
2   * lab2.c
3   *
4   * Created: 2014-11-22 13:24:46
5   * Author: Kamil Wjcik
6   */
7
8  #include <avr/io.h>
9  #include <avr/interrupt.h>
10
11 #define F_CPU 1000000UL
12 #include <util/delay.h>
13
14 void init(void);
15 void showDigit(void);
16
17 uint8_t timeCounter = 0;
18 uint8_t timeCounterLimit = 61;
19 uint8_t digit = 0;
20
21 uint8_t digitValues[10];
22
```

```
23 ISR(INT0_vect){
24     timeCounter --; // decrease time counter
25
26     if (timeCounterLimit != 0){
27         digit ++; //increase second digit
28         digit = digit % 10; // reset digit if it reach 10
29         timeCounter = timeCounterLimit; // reset time counter
30     }
31 }
32
33 int main(void){
34     TCCR0 |= 0x03; // set prescaler value to 64
35     // by default timer is reset after overflow
36     TIMSK = 0x01; // set interrupts from overflow (TOIE0=1)
37     timeCounter = timeCounterLimit;
38
39     sei();
40
41     //use PORT A to control display
42     DDRA = 0xFF; // configure PORT A as output
43     init();
44     showDigit();
45
46     while(1){
47     }
48
49 }
50
51 void init(void){
52     digitValues[0] = 0x3f;
53     digitValues[1] = 0x06;
54     digitValues[2] = 0x5b;
55     digitValues[3] = 0x4f;
56     digitValues[4] = 0x66;
57     digitValues[5] = 0x6d;
58     digitValues[6] = 0x7d;
59     digitValues[7] = 0x07;
60     digitValues[8] = 0x7f;
61     digitValues[9] = 0x6f;
62
63 }
64 void showDigit(void){
65     uint8_t dataValue = digitValues[digit];
66
67     PORTA = 0x80 | dataValue;
```

68 }

Listing 1: Program zadanie 2.2

Literatura

- [1] Instrukcja do ćwiczeń laboratoryjnych
- [2] Nota katalogowa mikrokontrolera