

## Elektroniczny model niezawodności obiektu nieodnawialnego

**Streszczenie.** W artykule opisano koncepcję elektronicznego modelu niezawodności obiektu nieodnawialnego. Taki model może znaleźć zastosowanie w nauczaniu teorii niezawodności na uczelniach technicznych. Został przedstawiony model zbudowany z wykorzystaniem popularnego mikrokontrolera ATiny13A. Przeprowadzone zostały również badania modelu prototypowego.

**Abstract.** The concept of electronic reliability model of non-repairable item is described. This model can be used for educational purposes. The model circuit is based on popular ATiny13A microcontroller. The prototype studies were also performed. (**Electronic model of non-repairable item's reliability**).

**Słowa kluczowe:** teoria niezawodności, modelowanie niezawodności, obiekt nieodnawialny, model elektroniczny

**Keywords:** reliability theory, reliability modeling, non-repairable item, electronic model

### Wstęp

Niezawodność była, jest i będzie jednym z najważniejszych kryteriów, które powinny być rozpatrywane, zarówno podczas planowania rozwoju, jak i eksploatacji systemu elektroenergetycznego [6-9]. Zatem zagadnienia niezawodności systemu elektroenergetycznego, niezawodności zasilania odbiorców w energię elektryczną, bezpieczeństwa elektroenergetycznego powinny być, i są, elementami programów studiów wyższych w obszarze elektroenergetyki.

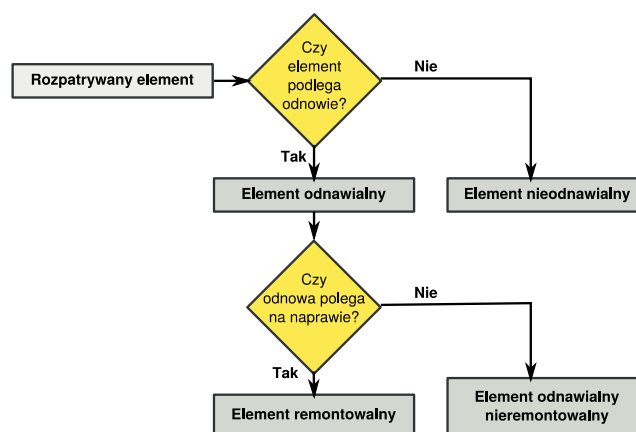
W nauczaniu teorii niezawodności na uczelniach technicznych pewnym problemem jest brak możliwości obserwacji przez studentów funkcjonowania elementów rzeczywistych. Wynika to z dwóch przyczyn. Po pierwsze częstość uszkodzeń elementów i obiektów jest zbyt niska, aby móc przeprowadzić te obserwacje podczas 1,5 godzinnych zajęć laboratoryjnych. Drugą przyczyną są znaczne koszty wiążące się z wymianą choćby najtańszych elementów, które ulegałyby uszkodzeniu. Dlatego, by móc przeprowadzić ze studentami badania awaryjności pewnych elementów trzeba zapewnić następujące warunki:

- czas poprawnej pracy badanego elementu musi być na tyle długi, aby człowiek mógł zaobserwować tę pracę, a z drugiej strony na tyle krótki, aby móc przeprowadzić wiele obserwacji podczas 90 min zajęć;
- zastąpienie uszkodzonego elementu nowym powinno być szybkie i nie wiązać się z ponoszeniem kosztów.

W niniejszym artykule zaproponowano zastosowanie do tego celu układu elektronicznego, który symulowałby działanie i niezawodność elementów (obiektów) rzeczywistych. Uszkodzenie obiektu w tym przypadku oznaczałoby jedynie zmianę stanu logicznego na wyjściu, a jego odnowę realizowałoby się przez podanie odpowiedniego sygnału na wejście lub też reset układu. Dodatkowo, aby zaprojektowany układ mógł znaleźć zastosowanie w zajęciach laboratoryjnych, musi on być tani i łatwo dostępny, aby móc zrealizować wiele podobnych stanowisk.

### Elementy nieodnawialne o wykładniczym rozkładzie czasów zdadności

Pojęciem pierwotnym w teorii niezawodności jest obiekt rozumiany w zależności od potrzeb jako niepodzielny element (bez uwzględnienia jego struktury wewnętrznej) lub zbiór elementów tworzących system [8]. Element jest to taki obiekt, który podczas analizy niezawodności jest traktowany jako niepodzielna całość. System zaś jest to grupa niezależnych (najczęściej) elementów, połączonych ze sobą w celu wykonywania określonego zadania i tworzących określoną strukturę niezawodnościową. Klasyfikacja elementów została przedstawiona na rys. 1.



Rys. 1. Klasyfikacja elementów

Element nieodnawialny jest w pełni scharakteryzowany przez rozkład czasu funkcjonowania  $\tau$  (bezawaryjnej pracy, zdadności).

Podstawowe charakterystyki (wskaźniki) niezawodności elementu nieodnawialnego to:

- funkcja zawodności (dystrybuanta rozkładu);

$$(1) \quad F(t) = P\{\tau < t\} = Q(t)$$

- funkcja niezawodności;

$$(2) \quad R(t) = P\{\tau \geq t\} = 1 - F(t)$$

- gęstość prawdopodobieństwa;

$$(3) \quad f(t) = \frac{dF(t)}{dt}$$

gdzie:  $F(t)$  – dystrybuanta zmiennej losowej;  $R(t)$  – prawdopodobieństwo funkcjonowania elementu (niezawodność elementu);  $f(t)$  – gęstość prawdopodobieństwa.

Względna gęstość prawdopodobieństwa zmiennej losowej  $\tau$  nazywa się intensywnością niesprawności awaryjnych (uszkodzeń) – zwana jest ona również funkcją ryzyka:

$$(4) \quad \lambda(t) = \frac{f(t)}{R(t)} = \frac{F'(t)}{1 - F(t)} = \frac{1}{R(t)} \cdot \frac{dR(t)}{dt}$$

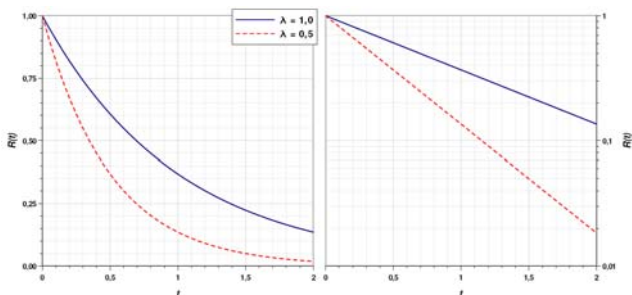
Zachodzi związek:

$$(5) \quad R(t) = R(0) \exp \left[ - \int_0^t \lambda(t) dt \right]$$

Najczęściej zakłada się, że w chwili rozpoczęcia eksploatacji  $t = 0$  element jest w stanie zdadności, czyli że  $R(0) = 1$ . Wtedy:

$$(6) \quad R(t) = \exp\left[-\int_0^t \lambda(t) dt\right]$$

Rozkład wykładniczy jest jednym z najczęściej stosowanych rozkładów w odniesieniu do elementów nieodnawialnych. Rozkład ten charakteryzuje się stałą wartością intensywności uszkodzeń  $\lambda(t) = \lambda = \text{const}$ . Oznacza to przyjęcie założenia, że prawdopodobieństwo uszkodzenia elementu w każdej chwili jest stałe, niezależnie od jego historii. Jest to zatem rozkład, który zakłada brak uszkodzeń wynikających ze starzenia się elementu. Rozkład wykładniczy jest stosowany w przypadku opisu elementów uszkadzających się w sposób losowy i nagły. Przykładami takich elementów mogą być elementy półprzewodnikowe, żarówki, czy też elementy uszkadzające się w wyniku przebicia izolacji. Wykres funkcji niezawodności dla elementów nieodnawialnych charakteryzujących się wykładniczym rozkładem czasu bezawaryjnej pracy został przedstawiony na rys. 2.



Rys. 2. Funkcja niezawodności  $R(t)$  elementu nieodnawialnego o wykładniczym rozkładzie czasu bezawaryjnej pracy (w skali liniowej po lewej i logarytmicznej po prawej)

Ze stałości intensywności uszkodzeń wynika, że można symulować pracę takiego elementu w sposób następujący. Zakłada się, że analizujemy pracę elementu w pewnych stałych i dostatecznie krótkich odstępach czasu. W każdej z chwil jest wybierana pewna liczba losowa  $r$  o rozkładzie równomiernym w przedziale  $(0, 1)$ . Jeżeli wylosowana liczba jest mniejsza od pewnego założonego prawdopodobieństwa  $p$  ( $r < p$ ), wówczas przyjmuje się, że element uległ uszkodzeniu i algorytm kończy się.

### Koncepcja elektronicznego modelu niezawodności obiektu nieodnawialnego

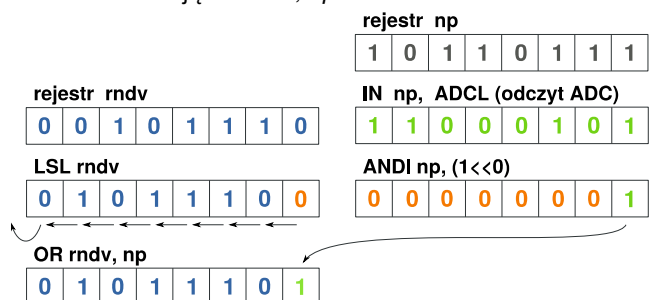
Opisana w poprzednim rozdziale właściwość rozkładu wykładniczego została wykorzystana do programowej symulacji pracy i uszkodzenia elementu nieodnawialnego. Zastosowany program sprawdza, w pewnym stałym odstępie czasu, czy 8-bitowa liczba losowa przyjmuje określoną wartość. Jeżeli testowana liczba ma charakter losowy i przyjmowane wartości są jednakowo prawdopodobne, to prawdopodobieństwo wybrania konkretnej wartości wynosi  $1/256$ . Jeżeli ten warunek jest spełniony, wówczas element przechodzi do stanu niezdatności. Jeżeli nie, wówczas losowanie i porównywanie jest powtarzane cyklicznie. Przejście do stanu niezdatności elementu jest sygnalizowane poprzez zmianę stanu logicznego wyjścia mikrokontrolera z wysokiego na niski. Jednocześnie zatrzymywany jest proces porównywania i element do czasu resetu pozostaje w stanie niezdatności. Symulowaną intensywność uszkodzeń można dobierać zmieniając liczbę cykli, po których jest wykonywane porównywanie lub zmieniając częstotliwość taktowania mikrokontrolera.

### Generowanie liczb pseudolosowych na potrzeby modelu

Dla poprawnej pracy opisanego algorytmu istotne jest

uzyskanie ciągu liczb losowych. Metody generowania liczb losowych można podzielić na dwie grupy. Pierwsza z tych grup to generatory fizyczne, które wykorzystują przebiegające losowo zjawiska fizyczne i zamieniają ich mierzalne parametry na ciąg liczb. Drugi sposób to wykorzystanie algorytmów matematycznych do tworzenia ciągu liczb nazywanych pseudolosowymi. O ile generatory fizyczne tworzą niepowtarzalny ciąg liczb losowych, o tyle generatory matematyczne dla tych samych danych źródłowych, nazywanych zarodnikami bądź nasionami (ang. *seed*), tworzą w rezultacie powtarzalny ciąg liczb.

W proponowanej aplikacji został użyty generator algorytmiczny, który zarodniki otrzymuje z generatora fizycznego. W roli generatora fizycznego zastosowano odczyt szumu złącza p-n tranzystora za pomocą wbudowanego w mikrokontroler 10-bitowego przetwornika analogowo-cyfrowego. Jest to modyfikacja rozwiązania zaproponowanego w [3]. Do generowania liczby losowej jest używany jedynie najmłodszy bit z wejścia przetwornika. Zasada tworzenia liczby losowej została przedstawiona na rys. 3. Do rejestru pomocniczego  $np$  jest odczytywany młodszy bajt z przetwornika (instrukcją  $IN\ np, ADCL$ ). W kolejnym kroku są zerowane wszystkie bity w tym bajcie, oprócz najmłodszego (instrukcją  $ANDI\ np, (1 << 0)$ ). Rejestr przechowujący wartość losową  $rndv$  jest przesuwany w lewo (instrukcją  $LSL\ rndv$ ). Następnie oba rejestry są scalane instrukcją  $OR\ rndv, np$ .



Rys. 3. Procedura tworzenia liczby losowej za pomocą wartości odczytanej z przetwornika ADC

Tak uzyskiwana liczba, po 8 odczytach przetwornika ADC, jest w pełni losowa, jeżeli na jego wejściu podawany jest szum. Okazuje się jednak, że zaproponowane rozwiązanie sprzętowe nie generuje wystarczającego szumu. Jednym z rozwiązań tego problemu mogłoby być dołączenie sprzętowego generatora szumu. To rozwiązanie komplikowałoby układ. Dlatego zaproponowane zostało inne rozwiązanie. Wykorzystany został liniowy generator kongruencyjny liczb pseudolosowych (ang. *pseudorandom number linear congruential generator* – LCG), który pozwala na wyznaczenie sekwencji liczb naturalnych pseudolosowych na podstawie zależności [4, 5]:

$$(7) \quad x_i \equiv (ax_{i-1} + c) \bmod m$$

gdzie:  $x_i$  –  $i$ -ta liczba pseudolosowa,  $x_{i-1}$  – poprzednia liczba pseudolosowa,  $a$  – mnożnik (ang. *multiplier*),  $c$  – przyrost (ang. *increment*),  $m$  – moduł (ang. *modulus*).

Liczy  $a$ ,  $c$  i  $m$  są pewnymi, odpowiednio dobranymi stałymi. Moduł  $m$  określa zakres generowanych liczb pseudolosowych, gdyż  $0 \leq x_i \leq m - 1$ . Stała  $a$  musi być w miarę duża, ale mniejsza niż  $m$ . Stała  $c$  musi być mniejsza od  $m$ , ale może również być równa 0. Aby uzyskać generowany ciąg o maksymalnym okresie (równym  $m$ ) należy kierować się następującymi zasadami:

- liczby  $m$  i  $c$  muszą być względnie pierwsze;
- $a - 1$  musi być podzielne przez każdy z czynników pierwszych liczby  $m$ .

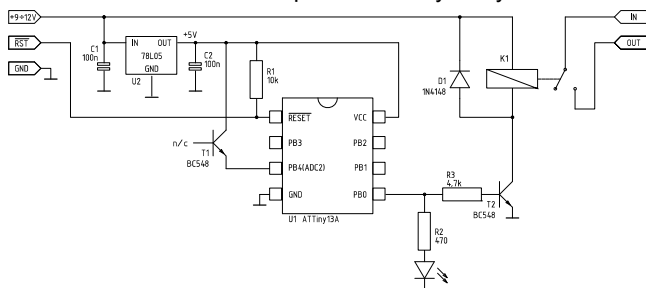
W programie przyjęto następujące wartości stałych:

- $m = 2^{16} = 65536$ , dzięki czemu generowana jest liczba pseudolosowa 16-bitowa;
- $a = 25173_{(10)} = 0110\ 0010\ 0101\ 0101_{(2)} = 6255_{(16)}$ ;
- $c = 13849_{(10)} = 0011\ 0110\ 0001\ 1001_{(2)} = 3611_{(16)}$ .

Tak dobrane liczby pozwalają na uzyskanie najdłuższego możliwego ciągu liczb pseudolosowych o długości  $2^{16} = 65536$ . Ponieważ młodsze bity w tak uzyskiwanych liczbach pseudolosowych nie są wystarczająco losowe, wobec tego jako wartość losowa jest przyjmowany starszy bajt, przechowywany w rejestrze *mdh*. Pewną wadą generatorów pseudolosowych jest to, że aby wygenerować ciąg liczb losowych, należy podać losową wartość początkową, zwaną ziarnem lub zarodkiem (ang. *seed*). Jeśli nie jest on losowy, wówczas cały ciąg liczb pochodzący z generatora jest powtarzalny. W proponowanym rozwiązaniu ziarno jest modyfikowane za pomocą opisanego wcześniej generatora fizycznego wykorzystując szum z wejścia przetwornika analogowo-cyfrowego. Generator hybrydowy, który jest połączeniem fizycznego generatora ziarna z programowym generatorem liczb pseudolosowych okazał się wystarczający do realizacji zadania stawianego układowi.

### Układ elektroniczny modelu

Schemat układu został przedstawiony na rys. 4.



Rys. 4. Schemat układu elektronicznego

Sercem projektowanego układu jest mikrokontroler **U1** Atmel ATTiny13A [1]. Jest to jeden z najtańszych (kosztuje około 3 zł) i najprostszych mikrokontrolerów, a jednocześnie jest powszechnie dostępny. Cechami predysponującymi go do tej aplikacji są: wbudowany wewnętrzny oscylator, dzięki czemu nie ma potrzeby wykorzystywania zewnętrznych rezonatorów oraz wbudowany przetwornik analogowo-cyfrowy, który jest wykorzystywany do generowania liczb pseudolosowych. Kontroler jest wyposażony w 6 programowalnych wejść-wyjść (oznaczonych PB0-PB5), z których jedno jest wykorzystywane jako reset. Wejście PB4 jest używane jako wejście przetwornika ADC, zaś na wyjściu PB0 pojawia się sygnał odpowiadający stanowi niezawodnościowemu modelowanego obiektu – stan „1” oznacza zdatność, zaś „0” niezdatność. Do wejścia ADC jest podłączony tranzystor **T1**, z niepodłączoną bazą, którego zadaniem jest generowanie szumu. Na wejście RESET, poprzez rezystor **R1**, jest podane napięcie źródła zasilania. Resetowanie układu, odpowiadające ponownemu uruchomieniu, może być realizowane zewnętrznie przez zwarcie wejścia **RST** z masą układu **GND**. Mikrokontroler jest zasilany napięciem stałym +5 V, które jest uzyskiwane za pomocą stabilizatora **U2** 78L05. Kondensatory **C1** i **C2** stabilizują pracę tego układu. Wyjście modelu stanowi przełącznik **K1** (o napięciu dostosowanym do napięcia zasilania), który jest uruchamiany kluczem tranzystorowym z wykorzystaniem tranzystora **T2**. Dioda **D1** pełni funkcję zabezpieczającą. Dodatkowo przewiduje się zastosowanie diody **LED** sygnalizującej stan zdatności modelu, z dodatkowym rezystorem ograniczającym jej prąd **R2**. Przedstawione na schemacie elementy **D1**, **K1**, **T1**, **T2**

mogą być bez problemów zastąpione innymi o podobnych parametrach.

### Kod programu

Sam model niezawodnościowy jest realizowany programowo. Program obsługujący mikrokontroler został napisany w języku Assembler AVR [1, 2]. Jego początek stanowi nagłówek, w którym zostały zdefiniowane używane rejestry, stałe oraz instrukcje obsługi przerwań mikrokontrolera.

```
.include "tn13adef.inc"
.def mp=R16 ;
.def np=R17 ; rejestry robocze
.def op=R18 ;
.def tpm=R19 ; rejestr licznika
.def mdh=R20 ; rejestry gen. liczb
.def mdl=R21 ; pseudo-losowych
.def mndv=R22 ; rejestr z wartoscia losowa
.def tc2=R23 ; rejestr licznika pom.
.equ tconst=24 ; stała czasowa
.equ tcpom=13 ; pom. stała czasowa
.equ log=0xA7 ; stała do porownania
.equ c=0x3619 ; stała c dla CLG
.org $0000 ; obsługa przerwan
rjmp MAIN ; Reset Handler
[...]
```

Kolejną część programu stanowi procedura obsługi przerwania przetwornika analogowo-cyfrowego. W tej procedurze jest realizowana główna część programu: tworzenie liczby pseudolosowej z wykorzystaniem przetwornika i algorytmu liniowego generatora kongruencyjnego (LCG) oraz wykorzystanie tej wartości do symulacji uszkodzenia.

```
ADC_INT: ; obsługa przerwania ADC
PUSH mp ; PUSH np
IN np, ADCL ; IN mp, ADCH; odczyt ADC
IN op, SREG ; PUSH op
ANDI np, (1<<0) ; odczyt ADC i uzycie
LSL mndv ; do wejścia generatora
OR mndv, np ; liczb pseudolosowych
DEC tpm ; licznik (wydluzenie
BRBC 1,EAID ; o stała czasowa)
LDI tpm, tconst ; generator pseudo-losowy LCG
MOV np, mdl ; MOV mp, mndh
; mnozenie rejestrów mndh:mdl przez a
LSL np
ROL mp
LSL np
ROL mp
ADD mdl, np
ADC mndh, mp
[...]
```

```
LSL mp
ADD mndh, mp
LSL mp
ADD mndh, mp
; dodanie stałej c
LDI mp, low(c) ; LDI np, low(c)
ADD mdl, np ; ADC mndh, mp
LDI mp, 0 ; dodanie ciągu z wejścia ADC
MOV np, mndv
ADD mdl, np
ADC mndh, mp

DEC tc2 ; licznik pomocniczy
BRBC 1,EAID
LDI tc2, tcpom
MOV mndv, mndh ; wartosc losowa
; porownaj z pewna stała
CPI mndv, log
BRBC 1,EAID
; ustaw 0 na wyjściu i wylacz przerwania
IN mp, PORTB
```

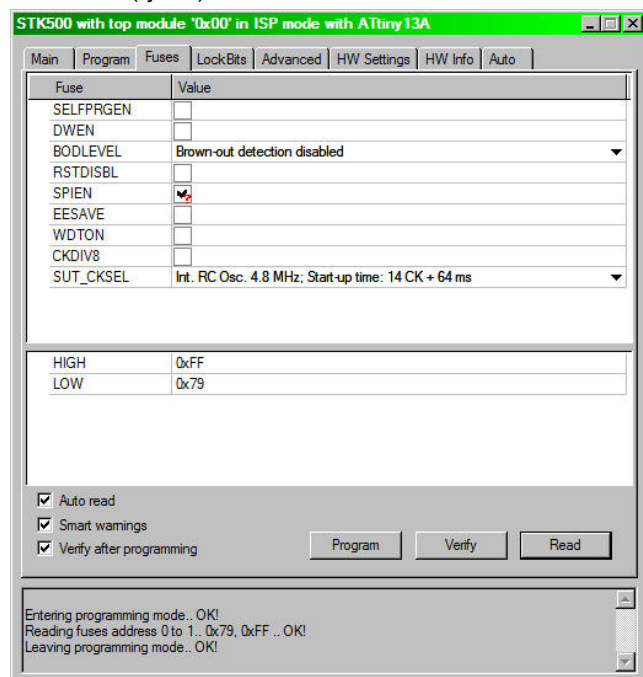
```

ANDI mp, ($FF-(1<<PB0))
OUT PORTB, mp
CLI
EAID:
POP op                OUT SREG, op
POP np                POP mp
RETI ; koniec przerwania ADC

```

Ostatnią częścią programu jest procedura główna, gdzie są ustawiane parametry oraz jest realizowana główna pętla programu. Procesor przez większość czasu pracy znajduje się w trybie bezczynności (uśpienia) – *Idle*. W tym trybie są wyłączone wszystkie funkcje mikrokontrolera z wyjątkiem: aktywnych liczników (timerów), watchdoga i systemu przerwań. Obudzenie procesora następuje w wyniku wystąpienia przerwania, przepelnienia licznika lub wyzerowania watchdoga. W prezentowanym programie układ jest wybudzany przez przerwanie przetwornika analogowo-cyfrowego, po którego zakończeniu procesor powraca do trybu uśpienia poleceniem *SLEEP*.

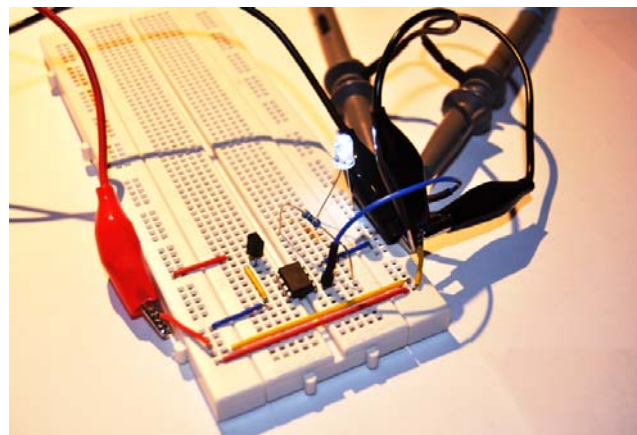
Po skompilowaniu program zajmuje 204 bajty, co stanowi 19,9% wielkości pamięci programu mikrokontrolera ATtiny13A. Duża rezerwa pozwala na łatwe rozbudowanie kodu o dodatkowe funkcje. Kontroler jest taktowany oscylatorem wewnętrznym RC z częstotliwością 4,8 MHz (bez wewnętrznego dzielnika częstotliwości). W tym celu są ustawione następujące wartości bitów „fuse”: HIGH 0xFF, LOW 0x79 (rys. 5).



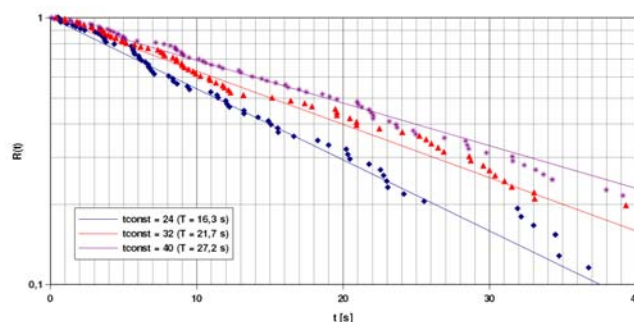
Rys. 5. Ustawienia bitów „fuse” w oknie programatora środowiska AVR Studio 4.0

### Badanie układu prototypowego

Do badania zachowania proponowanego modelu został użyty układ uproszczony, zrealizowany na płytce prototypowej (rys. 6), w którym nie ma przekątnika wyjściowego oraz układu stabilizatora napięcia, zaś zasilanie jest podawane bezpośrednio z zasilacza stabilizowanego 5 V DC. Wyjściowy pin PB0 został podłączony do miernika czasu. Zostały przebadane trzy konfiguracje układu, dla stałych czasowych *tconst* równych: 24, 32 oraz 40. Odpowiada to kolejno następującym rzeczywistym stałym czasowym rozkładu (uzyskanym empirycznie): 16,6 s, 22,2 s oraz 27,8 s. Dla każdej ze stałych czasowych wykonano pomiary czasu funkcjonowania obiektu. Wyniki zostały przedstawione na rys. 7.



Rys. 6. Badany układ prototypowy



Rys. 7. Uzyskane funkcje niezawodności badanego elementu dla różnych wartości stałych czasowych *tconst* oraz teoretyczne rozkłady wykładnicze im odpowiadające

Aby zweryfikować, czy uzyskane czasy przebywania układu w stanie zdatności mają zakładany rozkład wykładniczy można wykonać test zgodności  $\chi^2$  Pearsona [10]. Przyjmujemy hipotezę zerową  $H_0$ , że czas przebywania w stanie zdatności ma charakter wykładniczy ze stałą czasową  $T$ . Wówczas prawdopodobieństwo uszkodzenia  $Q$  pomiędzy chwilą  $t_1$  a  $t_2$  wyniesie:

$$(8) \quad Q(t_1, t_2) = p(t_1 \geq t_u > t_2) = R(t_1) - R(t_2) = e^{-t_1/T} - e^{-t_2/T}$$

gdzie:  $t_u$  – chwila uszkodzenia,  $R(t)$  – funkcja niezawodności elementu.

Uzyskane wyniki  $n$  pomiarów czasu przebywania w stanie zdatności przedstawiono w postaci szeregu rozdzielczego o  $k$  przedziałach. Dla poprawności testu minimalna liczebność przedziału nie powinna być mniejsza niż 5. Oczekiwana teoretyczna liczebność przedziału o początku  $t_p$  i końcu  $t_k$  jest równa  $np_i = nQ(t_p, t_k)$ . Do weryfikacji hipotezy  $H_0$  można wykorzystać statystykę:

$$(9) \quad \chi^2 = \sum_{i=1}^k \frac{(n_i - n_i p_i)^2}{n_i p_i}$$

gdzie:  $n_i$  – liczba zaobserwowanych wartości z danego przedziału  $i$ ;  $p_i = Q(t_p, t_k)$  – prawdopodobieństwo tego, że czas przyjmie wartości z zakresu przedziału  $i$  o początku  $t_p$  i końcu  $t_k$ , jeżeli badany rozkład ma charakter wykładniczy.

Statystyka testu ma rozkład  $\chi^2$  o  $d = k - r - 1$  stopniach swobody, gdzie  $r$  to liczba szacowanych parametrów, od których zależy rozkład cechy w populacji. Jeżeli  $\chi^2 < \chi^2_{\alpha}$ , to nie ma podstaw do odrzucenia hipotezy zerowej  $H_0$ , w przeciwnym przypadku istnieją podstawy do odrzucenia hipotezy zerowej.  $\chi^2_{\alpha}$  jest krytyczną wartością statystyki odczytywaną z tablic lub wyliczaną dla danego poziomu istotności  $\alpha$  oraz danej liczby stopni swobody  $d$ . Najczęściej

przyjmowaną wartością poziomu istotności jest 0,05; co oznacza, że prawdopodobieństwo popełnienia błędu przy odrzuceniu prawdziwej hipotezy zerowej wynosi 5%.

Jeżeli dane zostały przedstawione w postaci szeregu rozdzielczego o  $k=7$  przedziałach, zaś analizowany rozkład (wykładniczy) ma jeden parametr szacowany  $T$ , wówczas  $r=1$ , zaś rozkład  $\chi^2$  ma 5 stopni swobody ( $d=7-1-1=5$ ). Przy przyjętym poziomie istotności  $\alpha=0,05$  oraz  $d=5$  stopniach swobody krytyczna wartość statystyki  $\chi^2_{\alpha} = 11,07$ . Dla uzyskanych wyników pomiarów (tabela 1-3) wartości statystyk  $\chi^2$  wynoszą odpowiednio: 4,40; 3,61 i 4,57. W każdym z analizowanych przypadków  $\chi^2 < \chi^2_{\alpha}$ , nie ma zatem podstaw do odrzucenia hipotezy zerowej  $H_0$ , że uzyskane czasy przebywania układu w stanie zdadności mają charakter wykładniczy. Można przyjąć, że badany układ spełnia w wystarczającym stopniu założenia projektowe i odwzorowuje element nieodnawialny o wykładniczym rozkładzie czasu zdadności.

Tabela 1. Test  $\chi^2$  dla stałej czasowej  $t_{const} = 24$  ( $T = 16,3$  s)

| i                          | $t_u$ [s]          | $n_i$     | $np_i$   | $(n_i - np_i)^2 / np_i$ |
|----------------------------|--------------------|-----------|----------|-------------------------|
| 1                          | $t_u < 5$          | 17        | 20,59    | 0,63                    |
| 2                          | $5 \leq t_u < 10$  | 20        | 15,16    | 1,55                    |
| 3                          | $10 \leq t_u < 15$ | 10        | 11,15    | 0,12                    |
| 4                          | $15 \leq t_u < 20$ | 6         | 8,21     | 0,59                    |
| 5                          | $20 \leq t_u < 25$ | 9         | 6,04     | 1,45                    |
| 6                          | $25 \leq t_u < 35$ | 7         | 7,72     | 0,07                    |
| 7                          | $t_u \geq 35$      | 9         | 9,07     | 0,00                    |
| <b><math>\Sigma</math></b> |                    | <b>78</b> | <b>-</b> | <b>4,40</b>             |

Tabela 2. Test  $\chi^2$  dla stałej czasowej  $t_{const} = 32$  ( $T = 21,7$  s)

| i                          | $t_u$ [s]          | $n_i$     | $np_i$   | $(n_i - np_i)^2 / np_i$ |
|----------------------------|--------------------|-----------|----------|-------------------------|
| 1                          | $t_u < 5$          | 17        | 17,66    | 0,02                    |
| 2                          | $5 \leq t_u < 10$  | 17        | 14,04    | 0,62                    |
| 3                          | $10 \leq t_u < 15$ | 9         | 11,15    | 0,41                    |
| 4                          | $15 \leq t_u < 20$ | 7         | 8,86     | 0,39                    |
| 5                          | $20 \leq t_u < 25$ | 5         | 7,04     | 0,59                    |
| 6                          | $25 \leq t_u < 35$ | 14        | 10,04    | 1,56                    |
| 7                          | $t_u \geq 35$      | 17        | 16,85    | 0,00                    |
| <b><math>\Sigma</math></b> |                    | <b>86</b> | <b>-</b> | <b>3,61</b>             |

Tabela 3. Test  $\chi^2$  dla stałej czasowej  $t_{const} = 40$  ( $T = 27,2$  s)

| i                          | $t_u$ [s]          | $n_i$     | $np_i$   | $(n_i - np_i)^2 / np_i$ |
|----------------------------|--------------------|-----------|----------|-------------------------|
| 1                          | $t_u < 8$          | 20        | 23,71    | 0,58                    |
| 2                          | $8 \leq t_u < 16$  | 21        | 17,67    | 0,63                    |
| 3                          | $16 \leq t_u < 24$ | 17        | 13,16    | 1,12                    |
| 4                          | $24 \leq t_u < 32$ | 9         | 9,81     | 0,07                    |
| 5                          | $32 \leq t_u < 40$ | 7         | 7,30     | 0,01                    |
| 6                          | $40 \leq t_u < 48$ | 8         | 5,44     | 1,20                    |
| 7                          | $t_u \geq 48$      | 11        | 14,78    | 0,97                    |
| <b><math>\Sigma</math></b> |                    | <b>93</b> | <b>-</b> | <b>4,57</b>             |

### Podsumowanie

Nauczanie teorii niezawodności nie jest łatwym zadaniem. Przedstawiane pojęcia wydają się dla studentów zbyt abstrakcyjne. W czasie prowadzonych zajęć

laboratoryjnych trudno jest zrealizować badanie funkcji niezawodności obiektów rzeczywistych, głównie ze względu na długi czas niezbędnej obserwacji. Pewną pomocą może być symulacja komputerowa. Lepiej jednak, jeśli student podczas zajęć może samodzielnie wykonać obserwacje i pomiary dla rzeczywistych obiektów i struktur. W tym celu można zastosować przedstawiony model elektroniczny. Do jego zalet należą prostota i niski koszt budowy, możliwość wielokrotnego wykorzystania oraz możliwość dostosowania parametrów układu do potrzeb zajęć laboratoryjnych. Proponowane urządzenia mogą być łączone między sobą, co pozwoli studentom obserwować nie tylko pojedyncze elementy, ale również całe struktury niezawodnościowe. Jeżeli dokona się zmian w programie można również symulować pracę elementów odnawialnych lub też o innym rozkładzie charakteryzującym ich niezawodność.

### LITERATURA

- [1] Atmel Corporation, ATtiny13A Datasheet, <http://www.atmel.com/Images/doc8126.pdf>
- [2] Atmel Corporation, AVR Instruction Set Manual, <http://www.atmel.com/Images/doc0856.pdf>
- [3] Dondu J., Koncepcja generatora liczb losowych z wykorzystaniem przetwornika ADC ze strony: <http://mikrokontrolery.blogspot.com/2011/04/generator-liczb-losowych.html>
- [4] Gentle J.E., Random Number Generation and Monte Carlo Methods. New York-Berlin-Heidelberg-Hong Kong-London-Milan-Paris-Tokyo: Springer-Verlag, 2005
- [5] Niemro W. Symulacje stochastyczne i metody Monte Carlo. Warszawa: Wyd. Uniwersytetu Warszawskiego, 2013
- [6] Paska J., Generation system reliability and its assessment. Archiwum Energetyki 1999; 1-2: 27-50
- [7] Paska J., Momot A., Bargiel J., Goc W. Application of TRELSS and Implementation of Value Based Transmission Reliability Approach at Polish Power Grid Company. 6<sup>th</sup> International Conference on „Electrical Power Quality and Utilisation – EPQU'2001”. Kraków, 19-21 Sept. 2001: 487-494
- [8] Paska J., Niezawodność systemów elektroenergetycznych. Warszawa: Oficyna Wydawnicza Politechniki Warszawskiej, 2005
- [9] Paska J., Chosen aspects of electric power system reliability optimization (Wybrane aspekty optymalizacji niezawodności systemu elektroenergetycznego). Eksploatacja i Niezawodność – Maintenance and Reliability 2013; No 2 (15): 202-208
- [10] Pawłowski Z., Statystyka matematyczna. Warszawa: PWN, 1976

**Autorzy:** mgr inż. Piotr Marchel, Politechnika Warszawska, Wydział Elektryczny, Instytut Elektroenergetyki, Zakład Elektrowni i Gospodarki Elektroenergetycznej, ul. Koszykowa 75, 00-662 Warszawa, e-mail: [pm.m@interia.pl](mailto:pm.m@interia.pl); prof. dr hab. inż. Józef Paska, Politechnika Warszawska, Wydział Elektryczny, Instytut Elektroenergetyki, Zakład Elektrowni i Gospodarki Elektroenergetycznej, ul. Koszykowa 75, 00-662 Warszawa, e-mail: [Jozef.Paska@ien.pw.edu.pl](mailto:Jozef.Paska@ien.pw.edu.pl)