***Wojskowa Akademia Techniczna***

***im. Jarosława Dąbrowskiego***

Laboratorium z przedmiotu:

[Wprowadzenie](http://shaql.w.staszic.waw.pl/~shaql/wcy/viewforum.php?f=13) do Automatyki

Sprawozdanie z ćwiczenia laboratoryjnego nr 1:

**Implementacja maszyny stanowej na mikrokontrolerze**

Prowadzący:

mgr inż. Małgorzata Rudnicka - Schmidt

**Wykonał:** Radosław Relidzyński

**Grupa:** WCY20IY4S1

**Data laboratoriów**: 07.04.2021 r.

Spis treści

[A. Treść zadania 3](#_Toc100256549)

[B. Schemat podłączenia sterownika Arduino do urządzenia sterowanego 3](#_Toc100256550)

[C. Opracowanie ćwiczenia 3](#_Toc100256551)

[1. Definicja stanów maszyny stanowej 3](#_Toc100256552)

[2. Sposób kodowania stanów 3](#_Toc100256553)

[3. Diagram przejść stanów FSM 3](#_Toc100256554)

[4. Tabela przejść stanów 3](#_Toc100256555)

[5. Wyrażenie algebraiczne funkcji przejścia 3](#_Toc100256556)

[6. Tabela funkcji wyjścia 3](#_Toc100256557)

[7. Wyrażenie algebraiczne funkcji wyjścia 3](#_Toc100256558)

[D. Tabulogram programu (szkic) 3](#_Toc100256559)

[E. Wyniki sprawdzania poprawności programu 3](#_Toc100256560)

[F. Wnioski 3](#_Toc100256561)

# Treść zadania

Zaprojektować układ sterowania zapalaniem lampki za pomocą jednego przycisku. Kolejne

naciśnięcie przycisku powinno spowodować zmianę stanu lampki: jeśli była zgaszona - powinna

się zapalić, jeśli zapalona - zgasnąć.

# Schemat podłączenia sterownika Arduino do urządzenia sterowanego

Obraz zawierający tekst, sprzęt elektroniczny

Opis wygenerowany automatycznie

# Opracowanie ćwiczenia

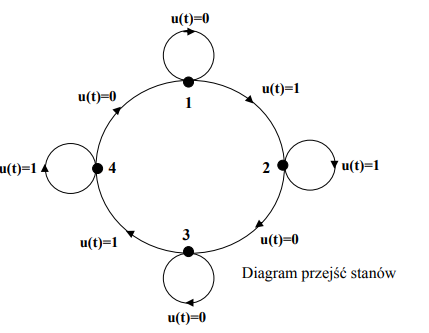
### Definicja stanów maszyny stanowej

|  |  |
| --- | --- |
| Stan | Stan Lampki |
| 1 | Nie świeci się |
| 2 | Świeci się |
| 3 | Świeci się |
| 4 | Nie świeci się |

### Sposób kodowania stanów

|  |  |  |
| --- | --- | --- |
| Stan\Kod | x1(t) | x2(t) |
| 1 | 0 | 0 |
| 2 | 0 | 1 |
| 3 | 1 | 0 |
| 4 | 1 | 1 |

### Diagram przejść stanów FSM



### Tabela przejść stanów

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| u(t) | x1(t) | x2(t) | x1(t+1) | x2(t+1) |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

### Wyrażenie algebraiczne funkcji przejścia

Tabela Karnought’a dla x1(t+1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| u\x1x2 | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |

Tabela Karnought’a dla x2(t+1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| u\x1x2 | 00 | 01 | 11 | 10 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

### Tabela funkcji wyjścia

Lampka początkowo jest w stanie 1 i jest wyłączona

|  |  |  |  |
| --- | --- | --- | --- |
| Stan | x1 | x2 | Q |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 1 |
| 3 | 1 | 0 | 1 |
| 4 | 1 | 1 | 0 |

Lampka jest zapalona dla stanów 2 oraz 3, a wyłączona dla stanów 1 oraz 4

### Wyrażenie algebraiczne funkcji wyjścia

# Tabulogram programu (szkic)

int ButtonPin =12;

int DiodaPin=5;

boolean Q1 = 0;

boolean M1 = 0;

boolean M2 = 0;

boolean M1p = 0;

boolean M2p = 0;

boolean I1 = 0;

void setup() {

  pinMode(ButtonPin, INPUT);

  pinMode(DiodaPin, OUTPUT);

  char \*hej="Setup passed";

  Serial.begin(9600);

}

void loop() {

  I1 = digitalRead(ButtonPin);

  M1p = !I1&!M1&M2 | M1&!M2 | I1&M1;

  M2p = I1;

  M1 = M1p;

  M2 = M2p;

  Q1 = !M1&M2 | M1&!M2;

  digitalWrite(DiodaPin,Q1);

  delay(100);

}

# Wyniki sprawdzania poprawności programu

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wybraniu opcji Szkic -> Weryfikuj/Kompiluj program kompiluje szkic, co potwierdza powyższy zrzut ekranu z komunikatem „Kompilacja zakończona”

# Wnioski

Szkic w pełni się kompiluje i spełnia swoje zadanie. Program działa zgodnie z układem sekwencyjnym zrealizowanym wstępnie na pierwszych ćwiczeniach laboratoryjnych. Wyłącznie wciśnięcia przycisku zmieniają wartość na wyjściu, opuszczenie przycisku zmienia stan układu, ale nie zmienia wartości na wyjściu.