

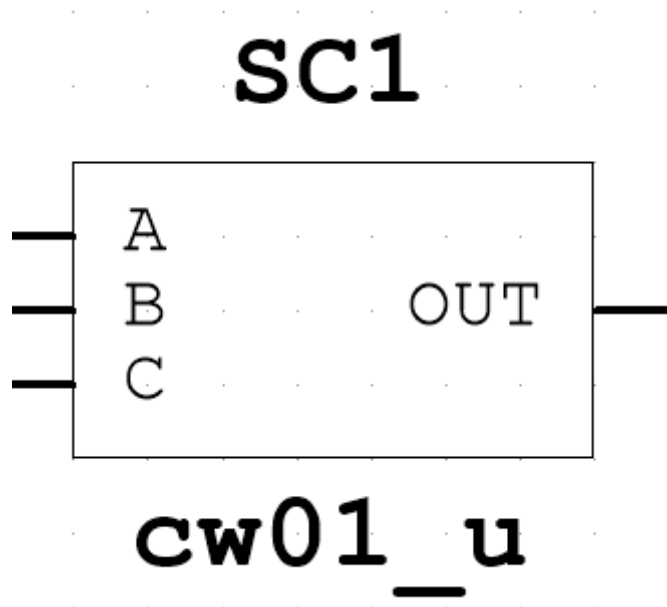
Sprawozdanie 1

Krzysztof Gryboś, Piotr Kądziela, Adrian Madej, Filip Piskorski

1. Zadanie 1a

Bazując wyłącznie na dwuwejściowych bramkach logicznych NAND, proszę od podstaw zaprojektować, zbudować i przetestować układ realizujący funkcję logiczną: $Y = \overline{A} \text{ xor } (B + C)$

1.1. Rozwiązanie teoretyczne



Rysunek 0. Koncepcja układu

Przekształcenia zastosowane w celu wykorzystania dwuwejściowych bramek NAND:

$$\begin{aligned} Y &= \overline{A} \text{ xor } (B + C) = \overline{A} \text{ xor } (\overline{\overline{B} + \overline{C}}) = \overline{A} \text{ xor } (\overline{\overline{B} * \overline{C}}) = \overline{A} * \overline{\overline{\overline{B} * \overline{C}}} + \overline{A} * \overline{\overline{B} * \overline{C}} = \\ &\overline{A} * \overline{B} * \overline{C} + A * \overline{\overline{B} * \overline{C}} = \overline{\overline{\overline{A} * \overline{B} * \overline{C}} + \overline{A * \overline{B} * \overline{C}}} = \\ &\overline{\overline{\overline{A} * \overline{B} * \overline{C}} * \overline{A * \overline{B} * \overline{C}}} = \overline{\overline{\overline{A} * \overline{B} * \overline{C}} * \overline{A * \overline{B} * \overline{C}}} \end{aligned}$$

Przekształcenia wykonano w oparciu o prawa de Morgana oraz prawo podwójnej negacji.

1.2. Tabela prawdy

A	B	C	$B + C$	\overline{A}	Y
0	0	0	0	1	1
0	0	1	1	1	0
0	1	1	1	1	0
0	1	0	1	1	0
1	0	0	0	0	0
1	0	1	1	0	1
1	1	1	1	0	1
1	1	0	1	0	1

Tabela 1. Tabela prawdy

1.3 Tabela Karnaugh

$A \backslash B C$	00	01	11	10
0	1	0	0	0
1	0	1	1	1

Tabela 2.1 Tabela Karnaugh

$A \backslash B C$	00	01	11	10
0	1	0	0	0
1	0	1	1	1

Tabela 2.2 Tabela Karnaugh

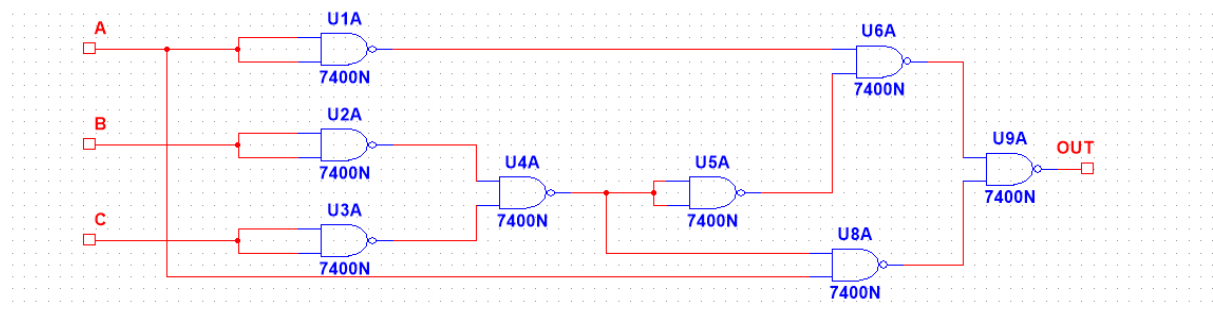
Otrzymujemy:

$$Y = \bar{A} * \bar{B} * \bar{C} + A * C + A * B = \bar{A} * \bar{B} * \bar{C} + \overline{\overline{A * C} + \overline{A * B}} = \bar{A} * \bar{B} * \bar{C} + \overline{\overline{A * C} * \overline{A * B}} = \bar{A} * \bar{B} * \bar{C} + \overline{\overline{A} * \bar{C} * \bar{A} * \bar{B}} = \bar{A} * \bar{B} * \bar{C} + \overline{\overline{A} * \bar{B} * \bar{C}} = \bar{A} * \bar{B} * \bar{C} + A * \overline{\bar{B} * \bar{C}} = \dots$$

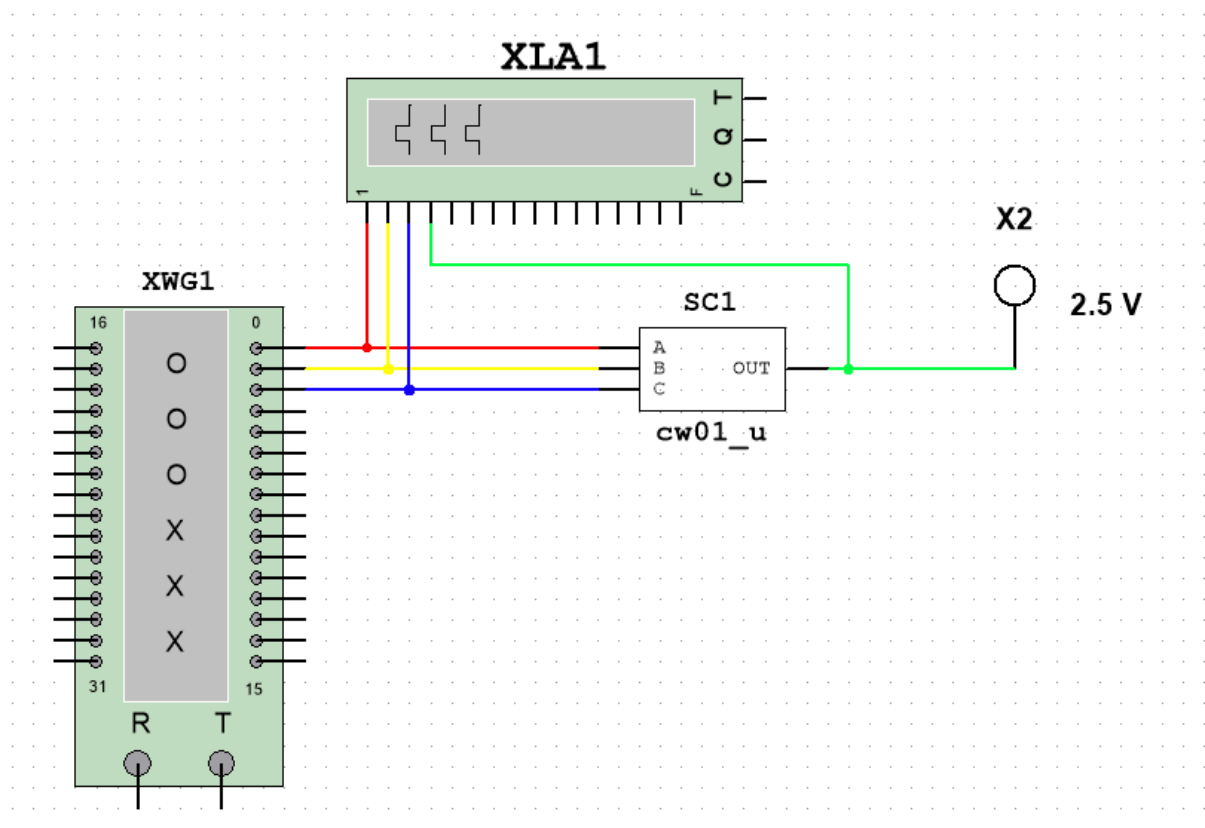
Doszlismy do miejsca z punktu 1.1 linijka 2 zatem nie bedziemy dalej tego wprowadzac.

1.4 Implementacja w programie Multisim

Uklad zostal skonstruowany przy uzyciu osmiu bramek NAND, generatora slow, analizatora logicznego oraz diody LED.

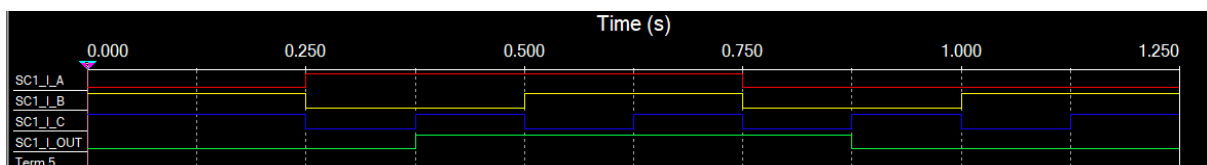


Schemat 1. Układ cw01_u w programie Multisim



Schemat 2. Układ testujący

Analizator logiczny przedstawia zmiany wejść kolejno A, B, C, Y



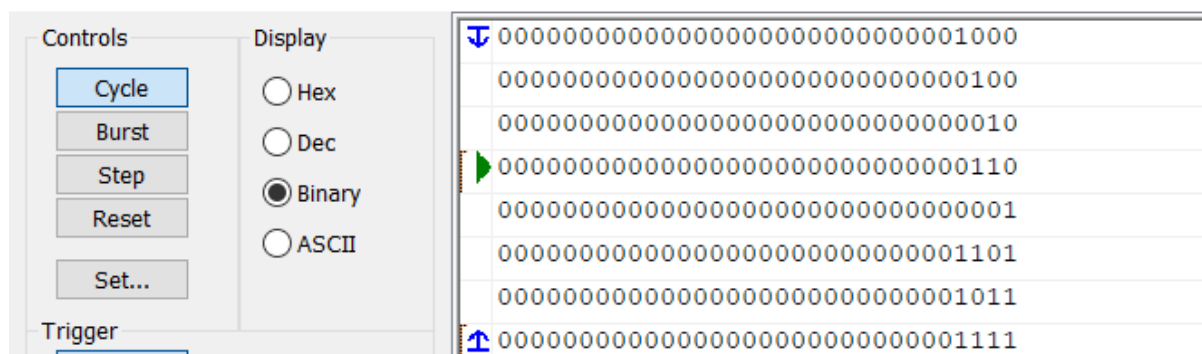
Rysunek 1. Analizator logiczny

Jak widać układ działa poprawnie.

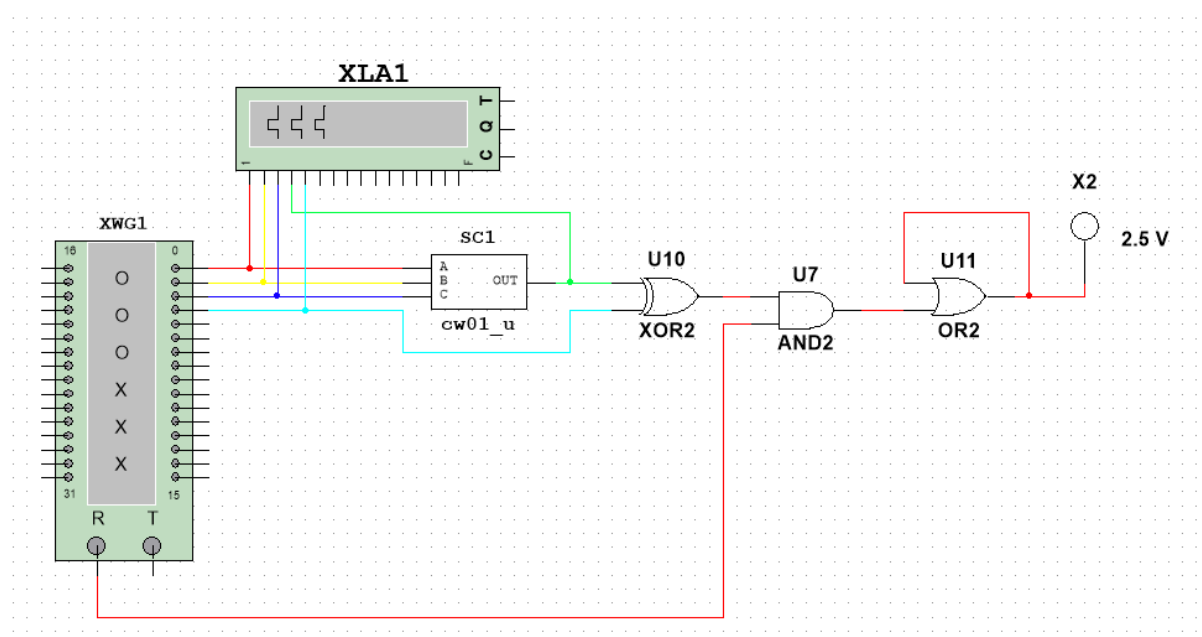
1.5 Testowanie

Układ testujący powstał w oparciu o bramkę XOR, AND oraz diodę. W generatorze słów na czwartym bicie zapisujemy oczekiwaną przez nas wartość. Dioda zapala się w przypadku niespójności sygnałów.

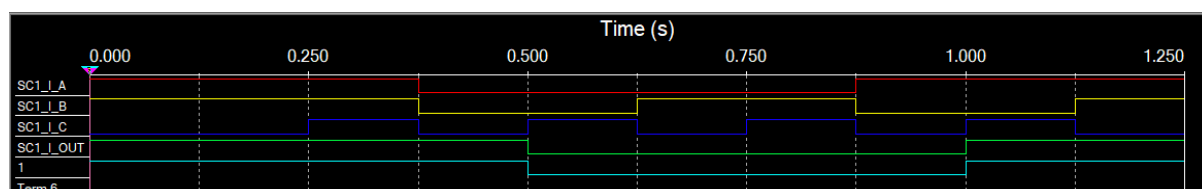
Word generator-XWG1



Rysunek 2. Generator słów



Schemat 3. Układ testujący 2



Rysunek. 3. Analizator logiczny 2

Jak widać wykresy o kolorze cyjanowym i zielonym mają identyczne kształty. Dioda led także się nie zapala, co dowodzi poprawności schematu.

1.6 Wnioski

1. Stosowanie praw i przekształceń logicznych pozwala zminimalizować ilość funkcji logicznych, co skutkuje mniejszą liczbą elementów elektronicznych potrzebnych do zbudowania układu realizującego daną funkcję.
2. Wykorzystując odpowiednie prawa i przekształcenia logiczne jesteśmy w stanie zrealizować każdą inną bramkę logiczną przy pomocy NAND.
3. Bramki NAND wykorzystuje się częściej w praktyce niż bramki AND i OR gdyż nie mają negacji i nie wymagają dodatkowego tranzystora zatem są tańsze w produkcji.
4. Podany układ znalazłby zastosowanie w systemach oświetlenia. A - zaprzeczenie poprzedniego stanu. B, C to przyciski służące do sterowania oświetleniem z różnych miejsc, np z wewnątrz i zewnątrz. Jeśli poprzedni stan był 0 i klikniemy któryś z przełączników to zapalimy światło, w przeciwnym przypadku je zgasimy
5. Kolejnym z przykładów byłaby obsługa telewizora gdzie, A to zaprzeczenie poprzedniego stanu telewizora, B - wyłącznik/włącznik telewizora na telewizorze, C - włącznik/wyłącznik telewizora na pilocie. Telewizor wówczas można wyłączyć za pomocą pilota, jak i przycisku znajdującego się na telewizorze.
6. Układ można także zastosować w mechanice pojazdowej. A to zaprzeczenie poprzedniego stanu, B - czujnik osoby na miejscu pasażera, C - czujnik zapięcia pasów. Jeśli osoba usiądzie na miejscu pasażera zapali się kontrolka, że musi zapiąć pasy, jeśli je zapnie to ona zgaśnie.
7. Układ z wykorzystaniem XOR ma wiele zastosowań w przypadku układów, które muszą pamiętać poprzedni stan układu.

2. Zadanie 1b

Rozważmy pomieszczenie w którym znajdują się: drzwi wejściowe

dwa okna (wszystko wyposażone w czujniki stanu zamknięcia).

Poza tym znajduje się tam: czujnik ruchu, syrena alarmowa (może być reprezentowana wskaźnikiem LED), dwa przyciski: uzbrojenia i rozbrojenia alarmu, dwa wskaźniki LED: alarm uzbrojony i alarm wyłączony, LEDowy czerwony sygnalizator problemu załączenia alarmu.

Alarm można uzbroić dedykowanym przyciskiem tylko wtedy, gdy w pomieszczeniu nie wykryto ruchu, a drzwi i okna są skutecznie zamknięte. Wówczas powinna zaświecić się kontrolka uzbrojenia alarmu. Jeśli warunki te nie są spełnione, zaświeca się czerwony sygnalizator problemu, a alarm pozostaje rozbrojony, co ciągle wówczas sygnalizuje stosowny wskaźnik LED.

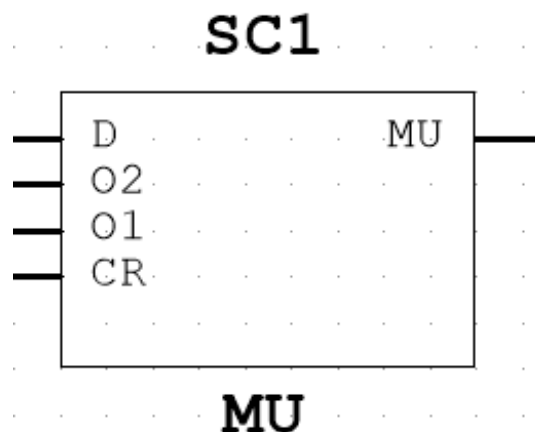
Poprawne uzbrojenie alarmu powoduje zgaszenie się wskaźnika rozbrojenia alarmu i sygnalizatora problemu (jeśli jest zaświecony) oraz powoduje zaświecenie się wskaźnika uzbrojenia alarmu.

Alarm uruchamia się, gdy system alarmowy jest uzbrojony i wykryty jest ruch lub sygnalizowane jest otwarcie: drzwi lub któregoś z okien.

2.1 Wyprowadzenie układu

2.1.1 Układ Można Uzbroić

Układ MU informuje czy alarm można uzbroić. Przyjmuje na wejściu informację od czujników i zwraca informację czy wykryto problem.



Rysunek 4. Koncepcja podobwołu MU

Oznaczenia:

- D - Drzwi wejściowe
- O2 - Okno 2
- O1 - Okno 1
- CR - Czujnik ruchu

Stan logiczny 0 na czujnikach D, O1, O2 interpretujemy jako włamanie, natomiast dla czujnika ruchu interpretujemy 1 jako wykrycie ruchu.

Na podstawie powyższych założeń tworzymy tabelę

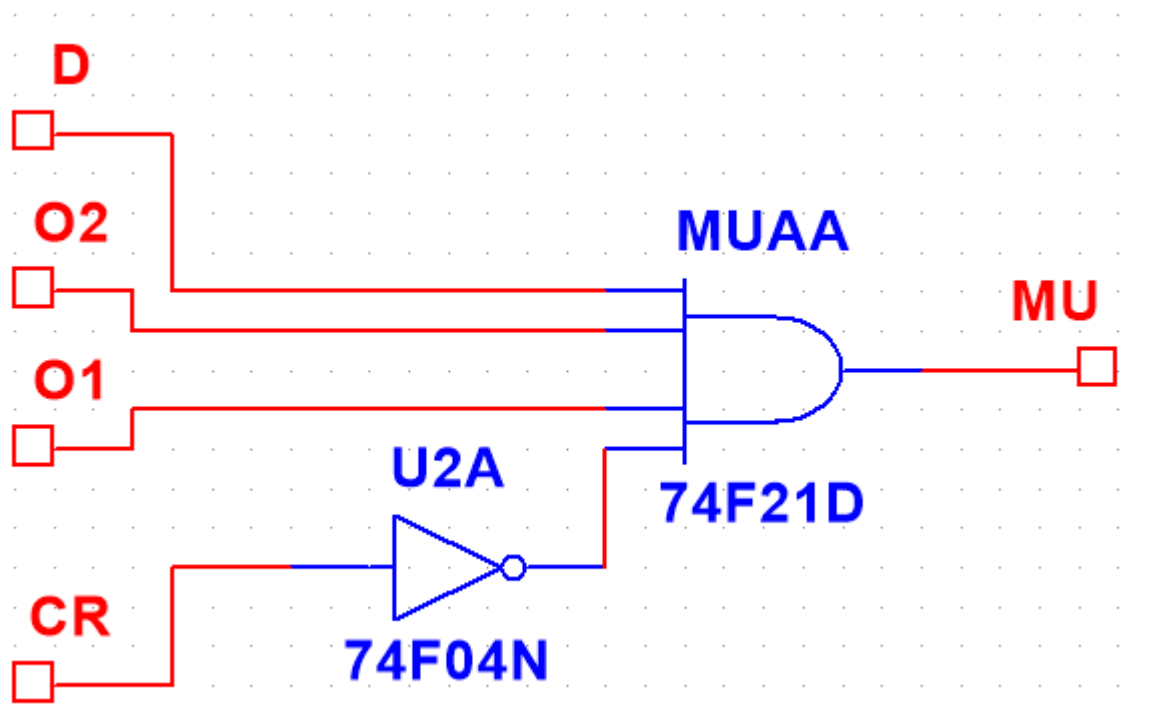
D	O2	O1	CR	MU
0	0	0	0	0
0	0	0	1	0
0	0	1	1	0
0	0	1	0	0
0	1	0	0	0
0	1	0	1	0
0	1	1	1	0
0	1	1	0	0
1	0	0	0	0
1	0	0	1	0
1	0	1	1	0
1	0	1	0	0
1	1	0	0	0
1	1	0	1	0
1	1	1	1	0
1	1	1	0	1

Tabela 3. Tabela prawdy na układzie MU

Układ pozostaje prawdziwy w przypadku następującej koniunkcji:

$$MU = D * O2 * O1 * \overline{CR}$$

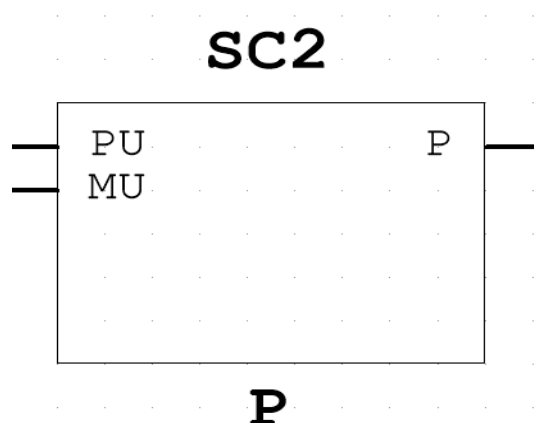
Zatem podobwód MU wygląda następująco



Schemat 4. Podobwód MU

2.1.2 Układ Problem

Układ P informuje czy wystąpił problem przy uzbrajaniu alarmu, na wejściu przyjmuje on informację, czy przycisk uzbrojenia jest włączony oraz czy można uzbroić alarm.



Rysunek 5. Koncepcja podobwodu P

Oznaczenia:

- PU - Przycisk uzbrojenia
- MU - Można uzbroić alarm
- P0 - stan wcześniejszy

- P - Problem w uzbrojeniu alarmu

Tworzymy tabelę dla układu P

MU	PU	P0	P
0	0	0	0
0	0	1	1
0	1	1	1
0	1	0	1
1	1	0	0
1	1	1	0
1	0	1	1
1	0	0	0

Tabela 4. Tabela prawdy dla układu P

Konstruujemy tabelę Karnaugh na podstawie tabeli 4

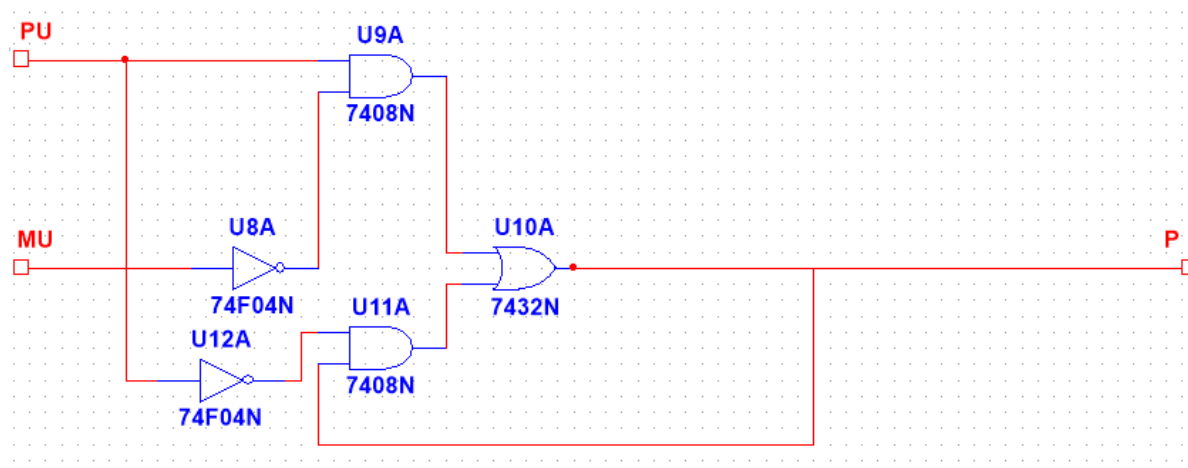
MU PU \ P0	0	1
00	0	1
01	1	1
11	0	0
10	0	1

Tabela 5. Tabela Karnaugh układu P

Otrzymujemy:

$$P = \overline{MU} * PU + P0 * \overline{PU}$$

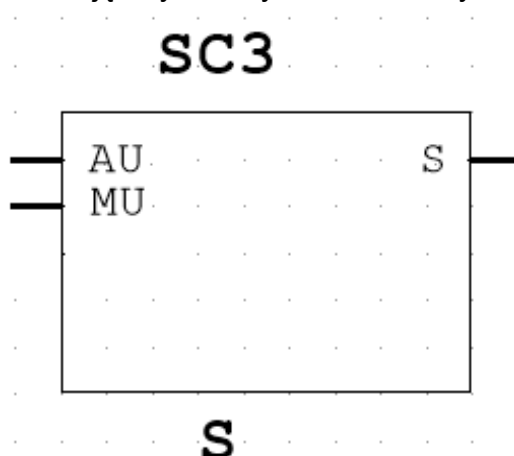
Na podstawie powyższego wzoru sporządzamy układ



Schemat 5. Podobwód P

2.1.3 Układ Syrena

Układ S przyjmuje informację czy zostało wykryte włamanie oraz czy alarm jest uzbrojony. Zwraca on informację czy należy uruchomić syrenę



Rysunek 6. Koncepcja podobwodu S

Oznaczenia:

- AU - Alarm uzbrojony
- MU - Włamanie (Można uzbroić)
- S0 - stan wcześniejszy
- S - Syrena

Sporządzamy tabelę dla możliwych przypadków

AU	MU	S0	S
0	0	0	0

0	0	1	0
0	1	1	0
0	1	0	0
1	1	0	0
1	1	1	1
1	0	1	1
1	0	0	1

Tabela 6. Tabela prawdy układu S

Na podstawie powyższej tabeli sporządzamy tabelę Karnaugh

AU MU \ S0	0	1
00	0	0
01	0	0
11	0	1
10	1	1

Tabela 7.1 Tabela Karnaugh układu S

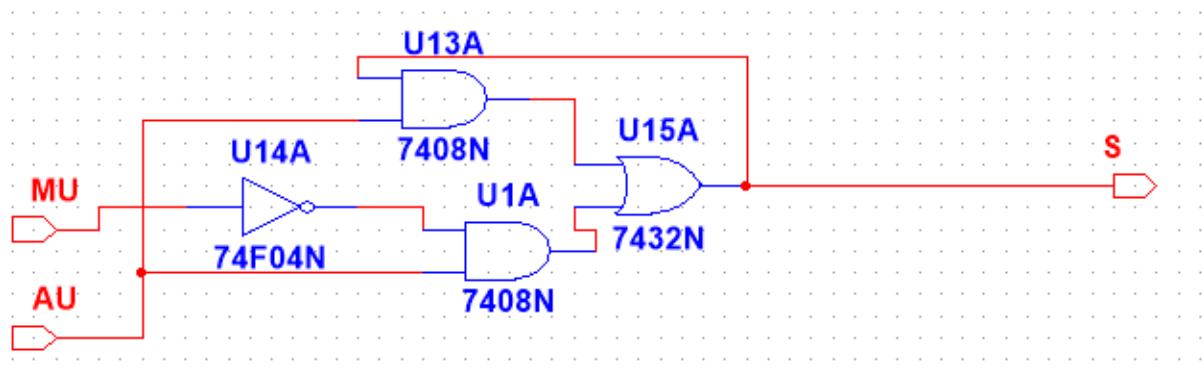
AU MU \ S0	0	1
00	0	0
01	0	0
11	0	1
10	1	1

Tabela 7.2 Tabela Karnaugh układu S

Otrzymujemy:

$$S = AU * \overline{MU} + S0 * AU$$

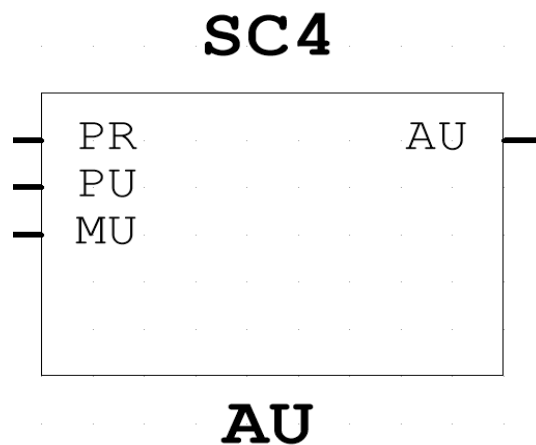
Co daje nam układ:



Schemat 5. Podobwód S

2.1.4 Układ Alarm Uzbrojony

Układ na wejściu przyjmuje informację czy przycisk można uzbroić oraz informację na temat stanu przycisku uzbrojenia i rozbrojenia.



Rysunek 6. Koncepcja podobwodu AU

Oznaczenia:

- PR - Przycisk rozbrojony
- PU - Przycisk uzbrojony
- MU - Można uzbroić
- AU0 - stan wcześniejszy
- AU - Alarm Uzbrojony

Tworzymy tabelę dla układu AU

MU	PU	PR	AU0	AU
0	0	0	0	0
0	0	0	1	1
0	0	1	1	0
0	0	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	0	1	1
0	1	0	0	0
1	1	0	0	1
1	1	0	1	1
1	1	1	1	0
1	1	1	0	0
1	0	1	0	0
1	0	1	1	0
1	0	0	1	1
1	0	0	0	0

Tabela 8. Tabela prawdy układu AU

Na podstawie powyższej tabeli tworzymy tabelę Karnaugh

MU PU \ PR AU0	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	1	1	0	0

10	0	1	0	0
----	---	---	---	---

Tabela 9.1. Tabela Karnaugh układu AU

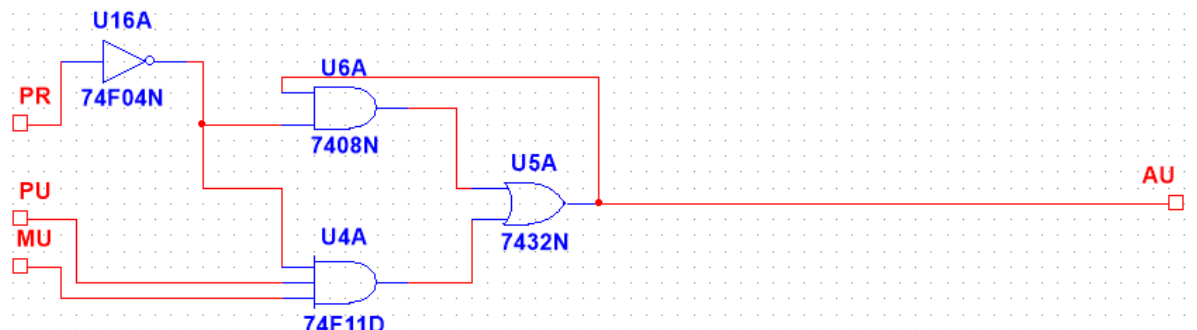
MU PU\ PR AU0	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	1	1	0	0
10	0	1	0	0

Tabela 9.2. Tabela Karnaugh układu AU

Otrzymujemy wzór:

$$AU = MU * PU * \overline{PR} + \overline{PR} * AU$$

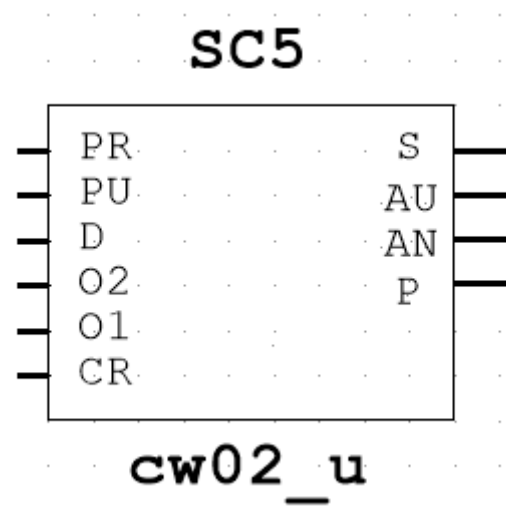
Na podstawie wzoru budujemy układ



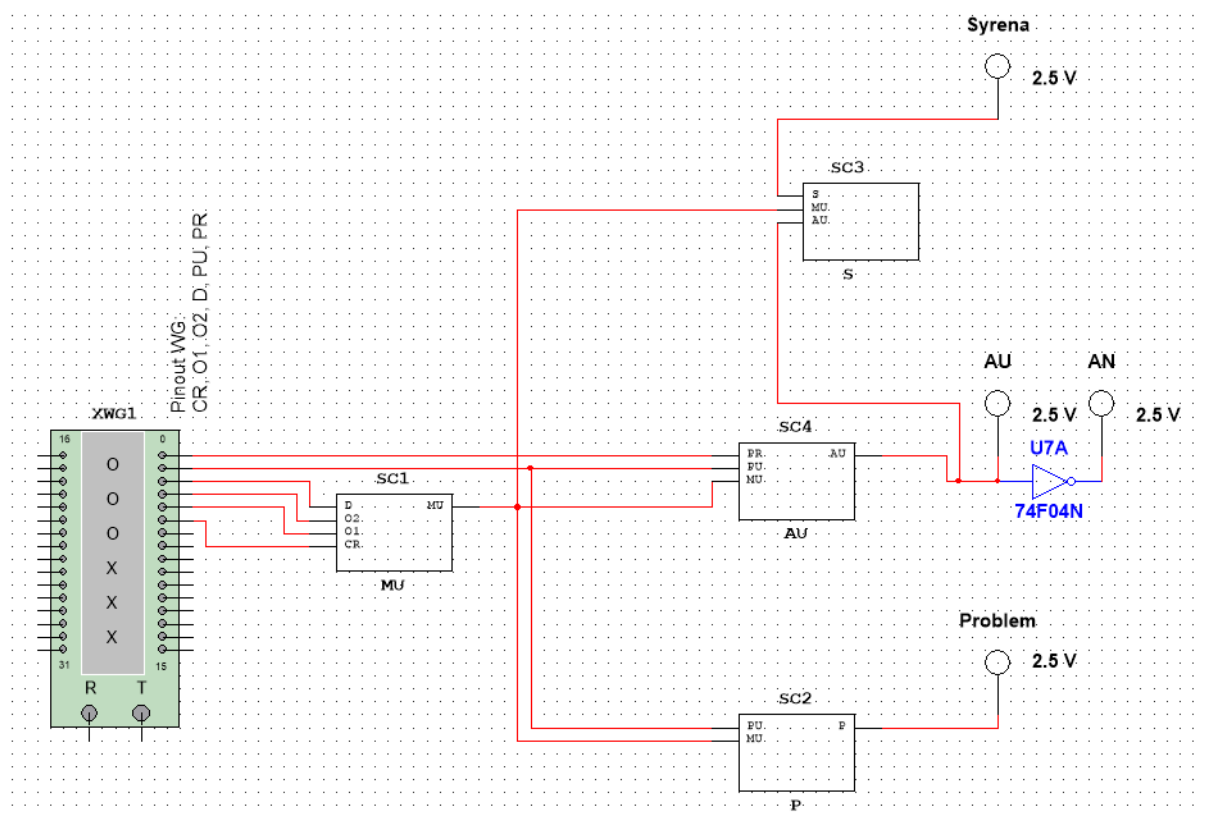
Schemat 7. Podobwód AU

2.2 Układ

Na podstawie podobwodów z rozdziału 2.1 otrzymaliśmy następujący układ



Rysunek 7. Koncepcja układu



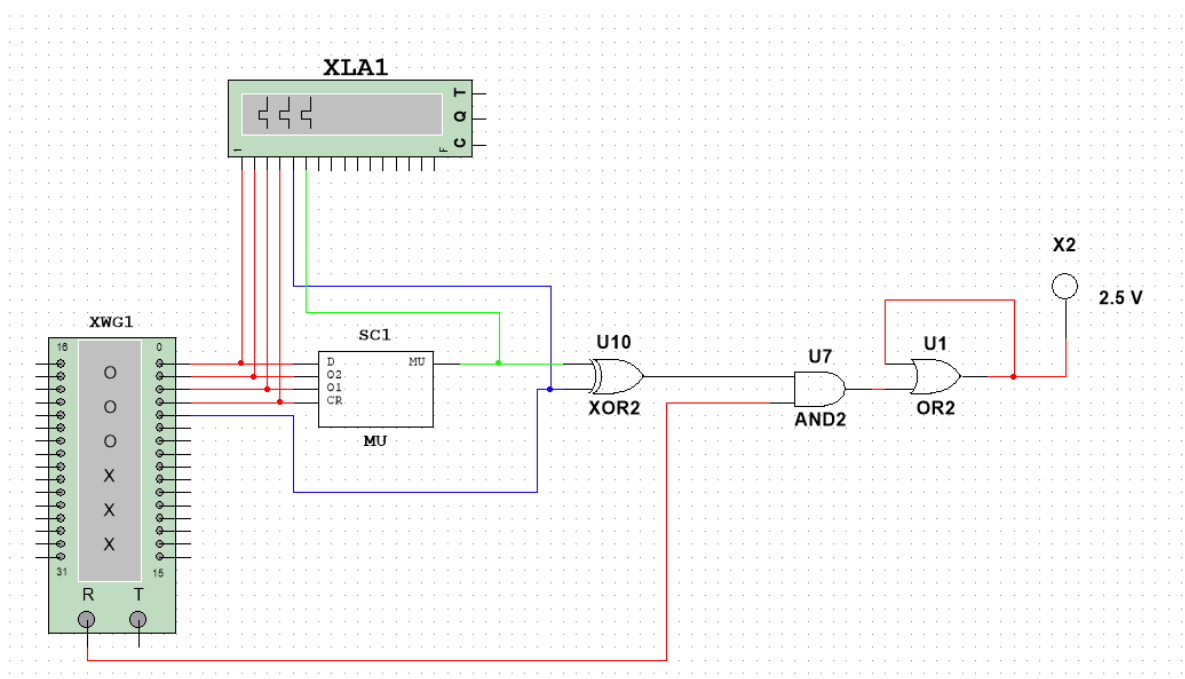
Schemat 7. Układ kompletny

2.3 Testowanie układu

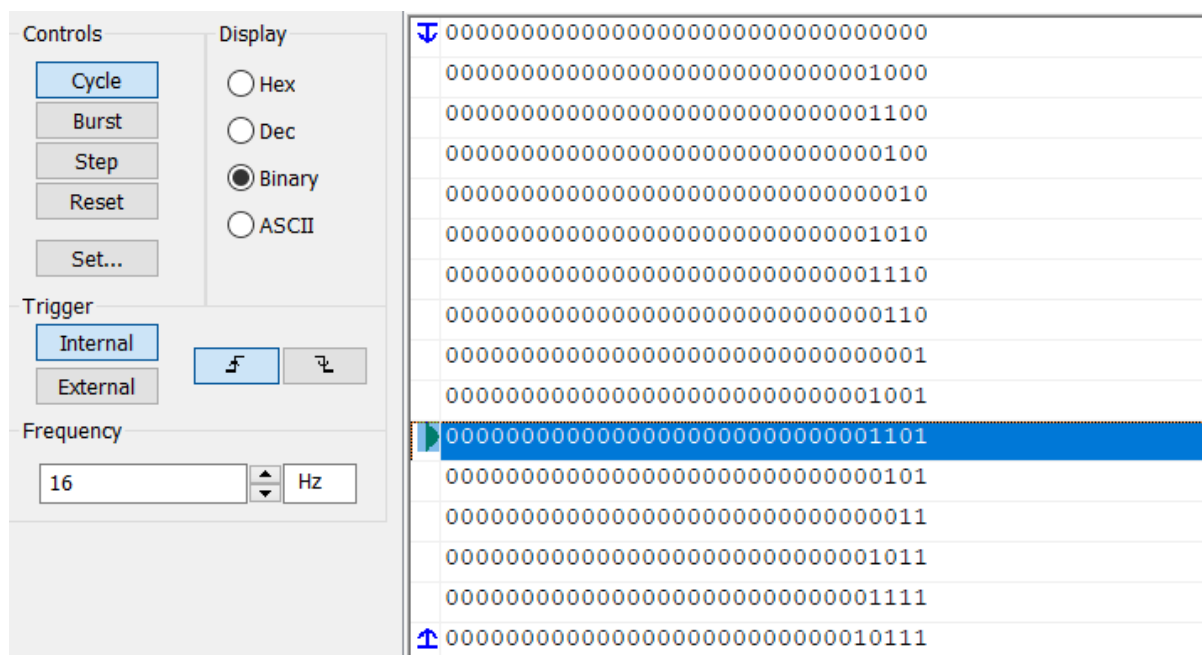
Testy zostaną przeprowadzone dla każdego podobwołu w oparciu o generator słów, analizator logiczny, bramki XOR i AND oraz diodę która zapali się w momencie niezgody.

2.3.1 Test układu MU

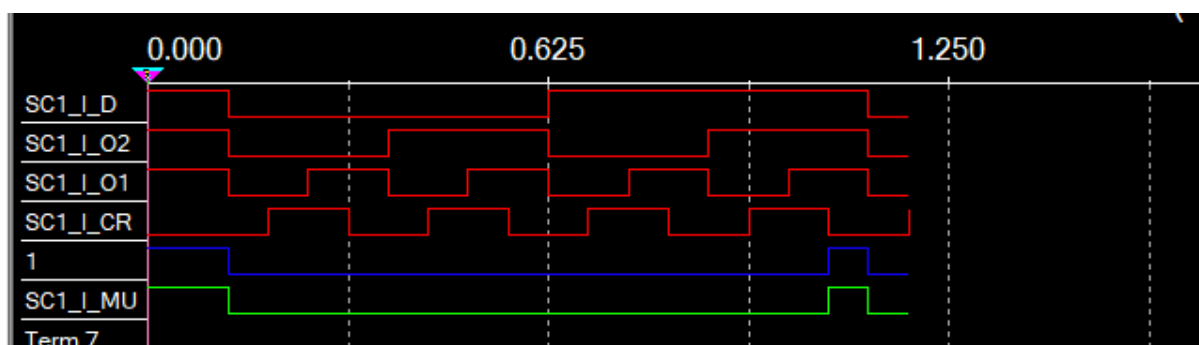
Testujemy układ MU, kolorem niebieskim oznaczyliśmy żądane wyjście, a zielonym otrzymane.



Schemat 8. Układ testujący MU



Rysunek 7. Generator słów dla testu MU

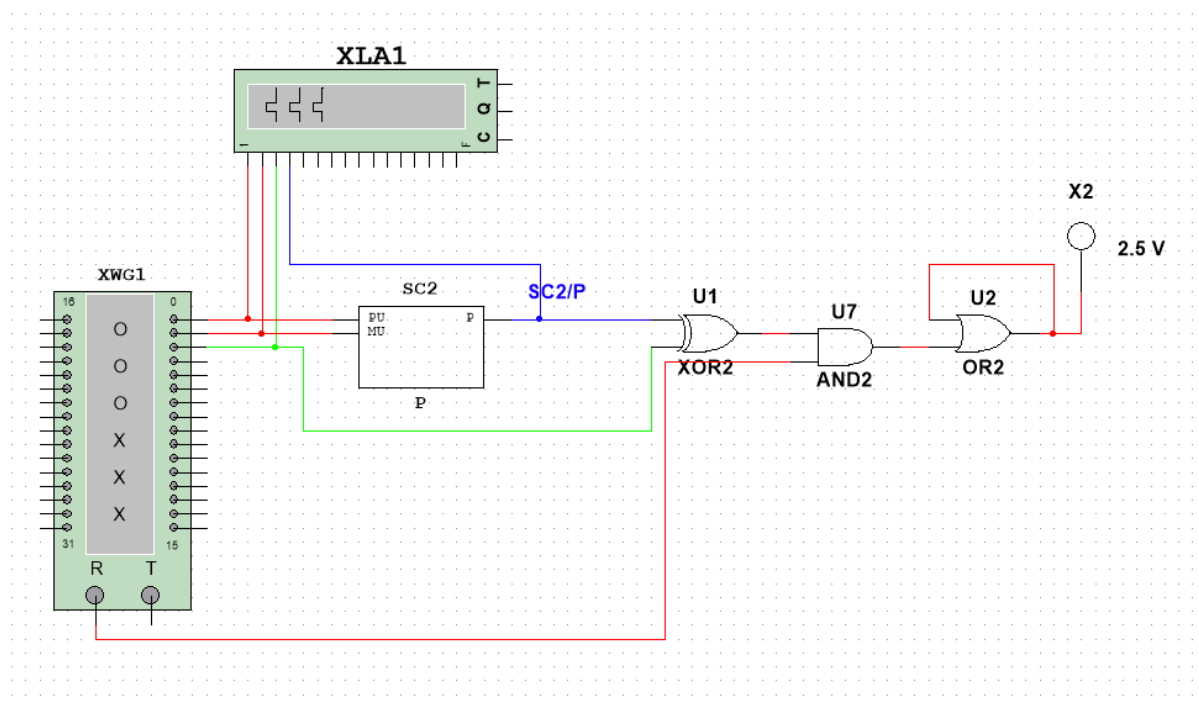


Rysunek 8. Generator słów dla testu MU

Dioda nigdy się nie zapala a wykresy niebieski i zielony mają taki sam kształt co potwierdza, że układ działa prawidłowo.

2.3.2 Test układu P

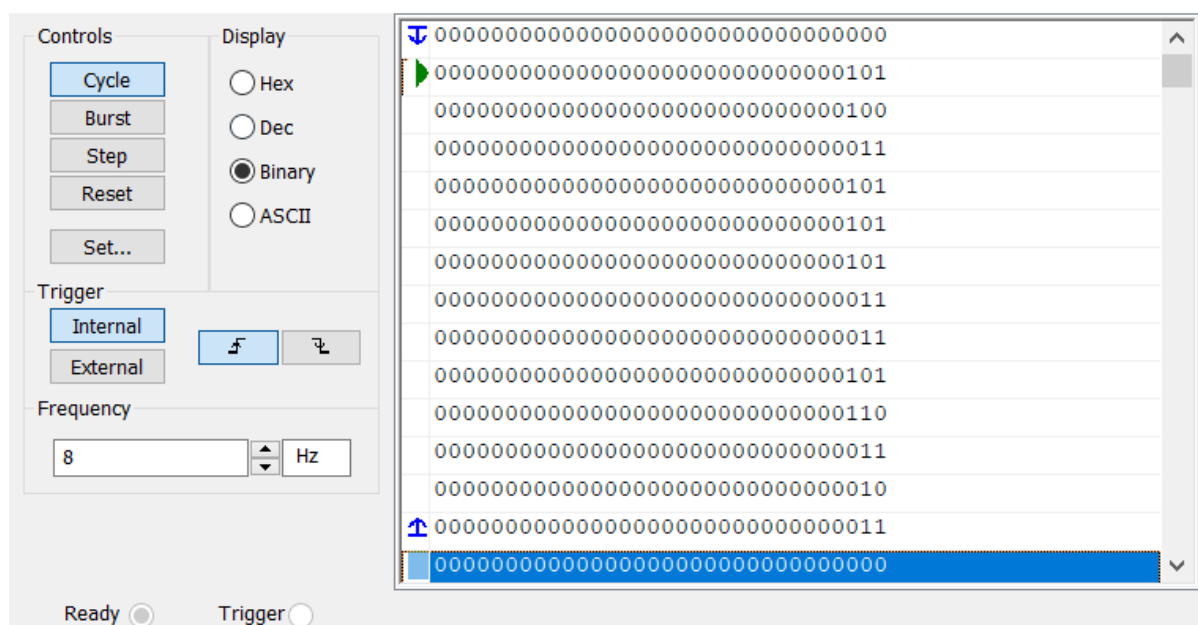
Testujemy układ P



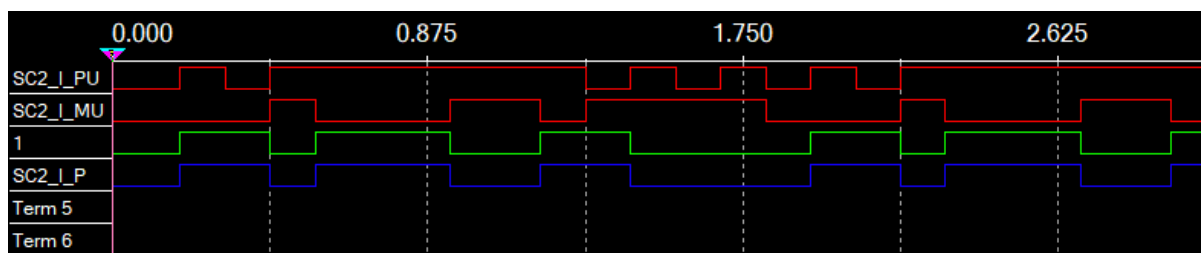
Schemat 8. Układ testujący P

W testowaniu uwzględniono poprzedni stan układu. Niezależnie od poprzedniego stanu dla wejść $PU = 1$, $MU = 1$, $P = 0$ oraz $PU = 1$, $MU = 0$, wówczas $P = 1$, co służyło do stymulacji poprzedniego stanu układu.

Word generator-XWG1



Rysunek 9. Generator słów dla testu P

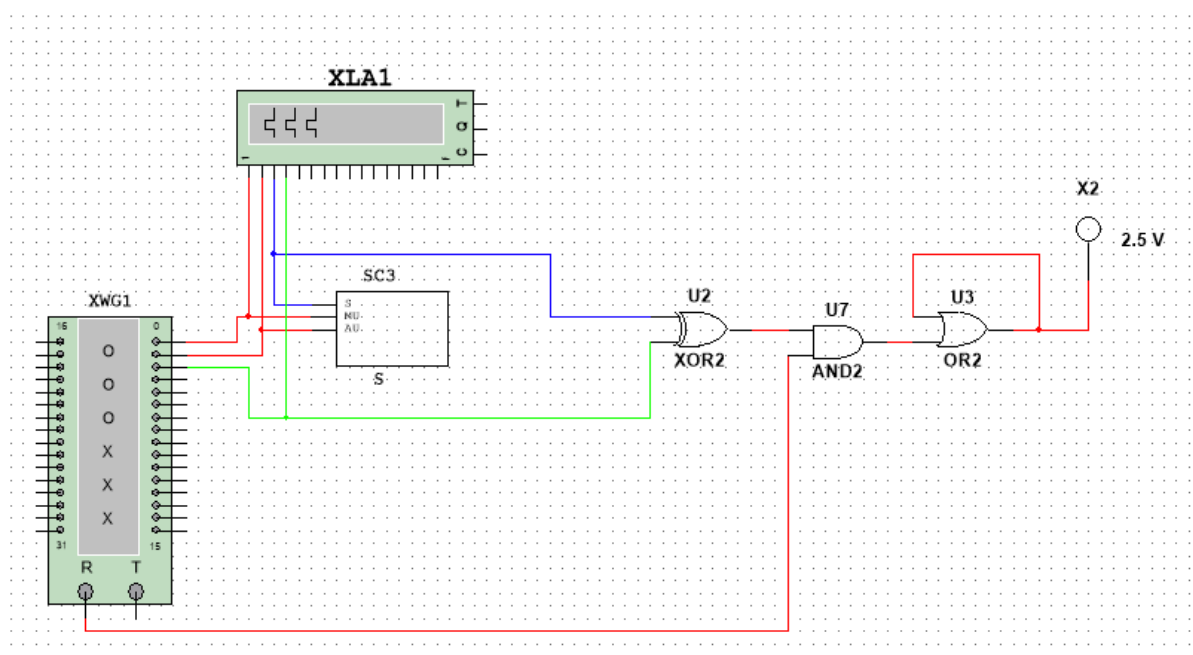


Rysunek 10 Analizator Logiczny dla testu

Jak widać wyniki żądane pokrywają się z wynikami rzeczywistymi co świadczy o poprawności układu

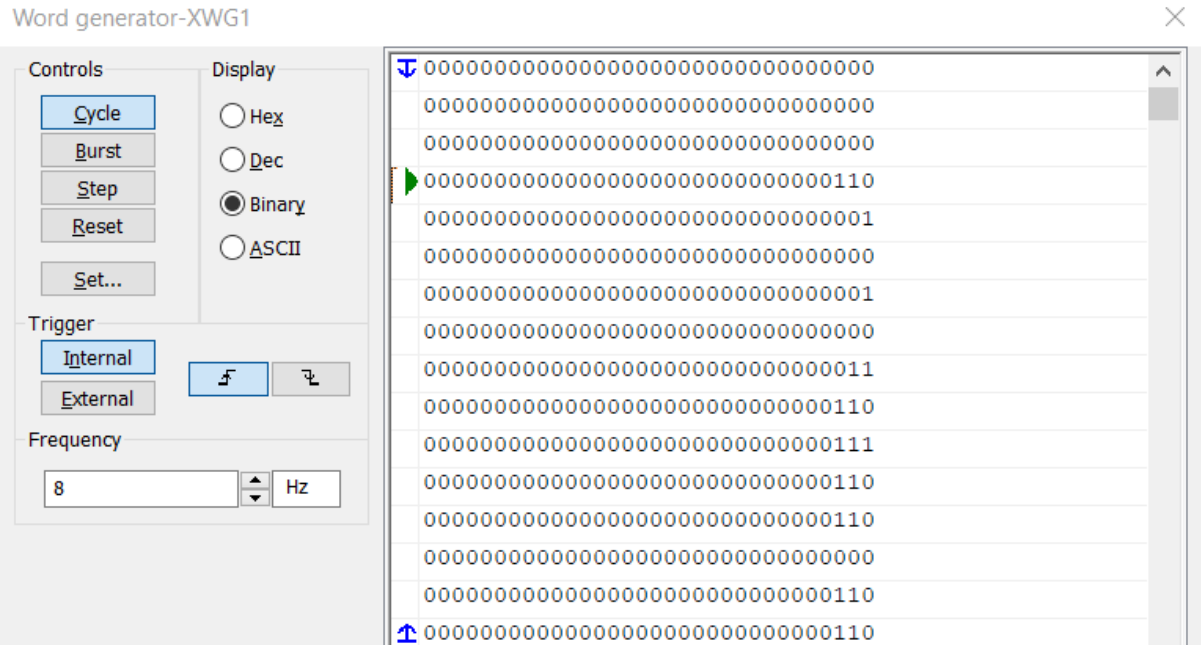
2.3.3 Test układu S

Schemat testujący układ S wygląda następująco

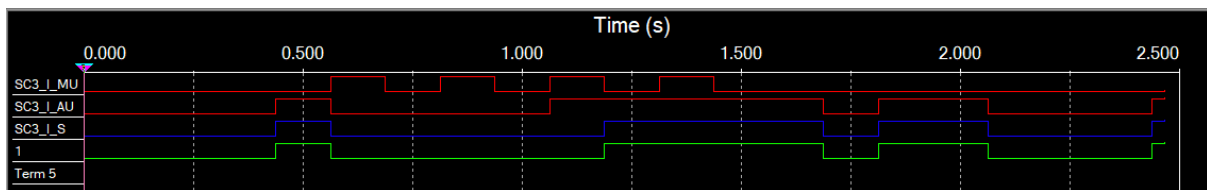


Schemat 8. Układ testujący S

Podobnie jak poprzednio stymulowaliśmy poprzedni stan podając odpowiednie wartości logiczne dla $AU = 0$, $MU = 0$, $S = 0$ oraz dla $AU = 1$, $MU = 0$, $S = 1$



Rysunek 11. Generator słów dla testu S

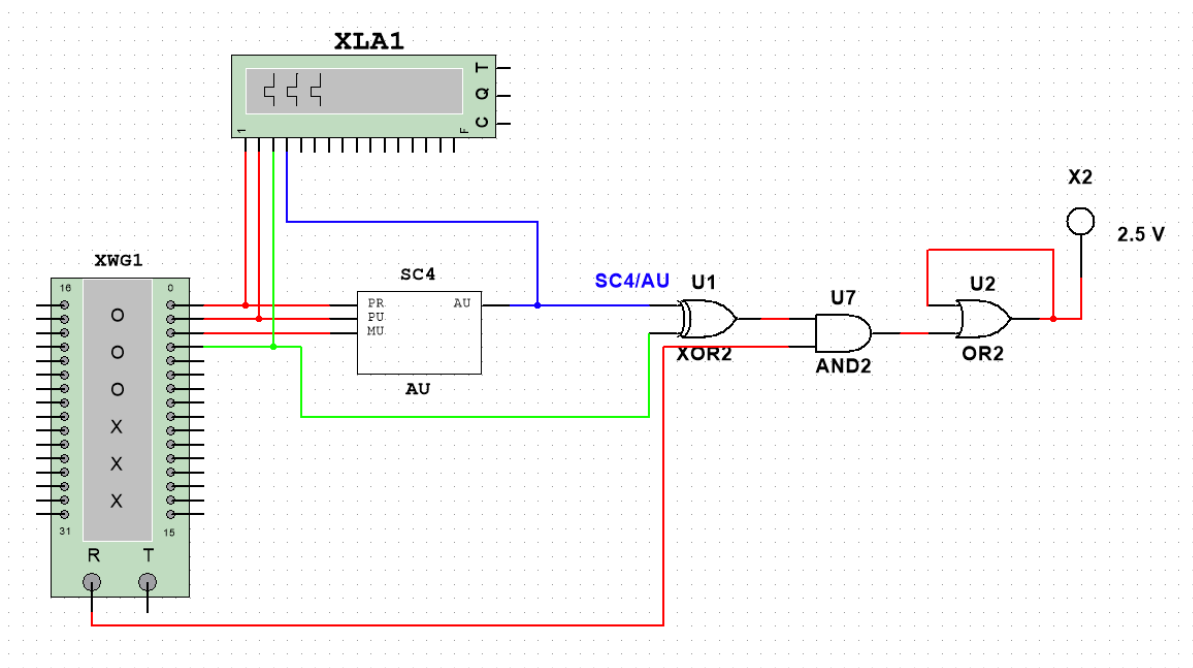


Rysunek 12. Analizator logiczny dla testu S

Jak widzimy ponownie odpowiednie wykresy mają identyczne kształty, dioda się nie świeci co oznacza iż układ działa poprawnie.

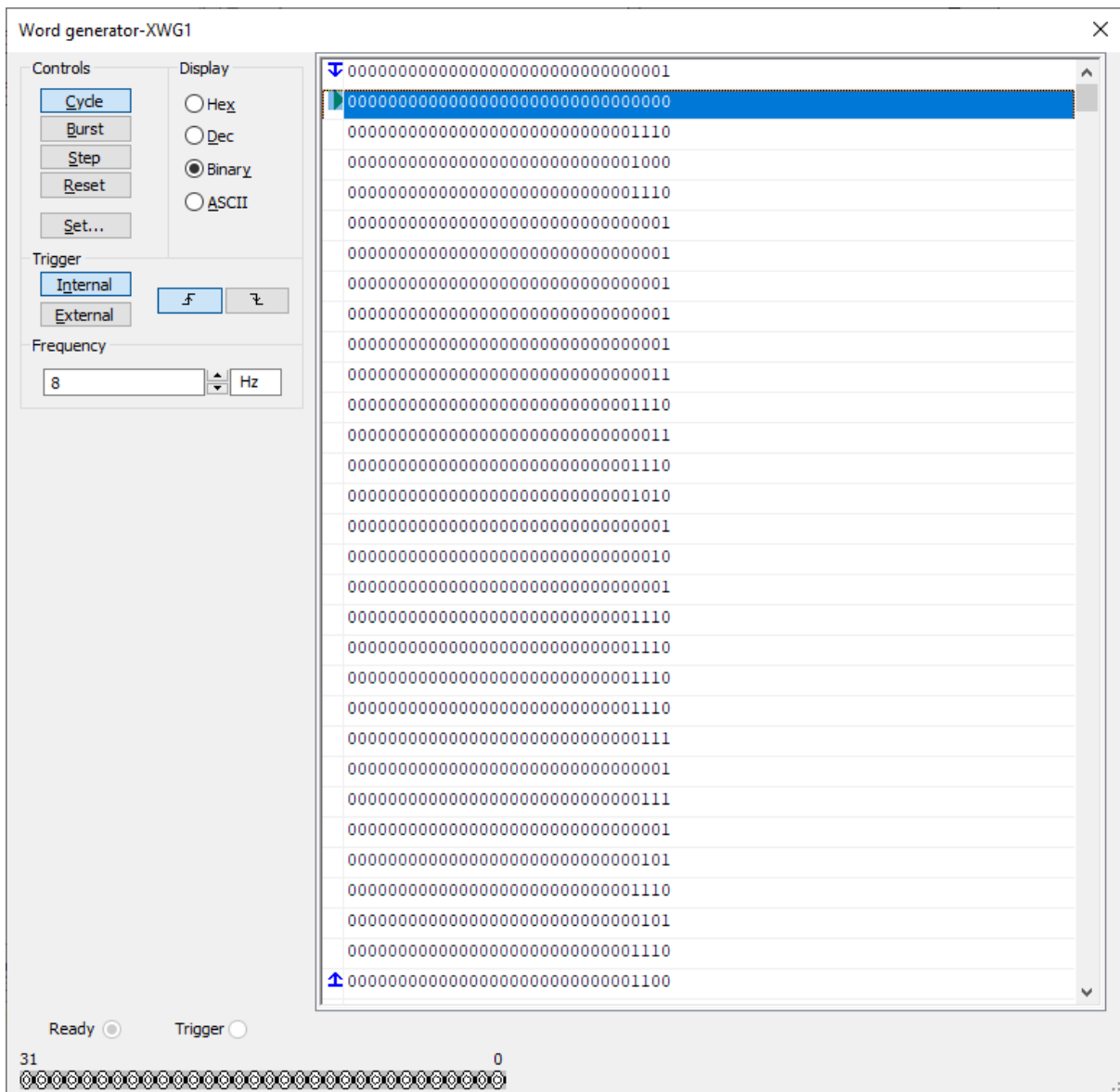
2.3.4 Test układu AU

Schemat układu AU wygląda następująco

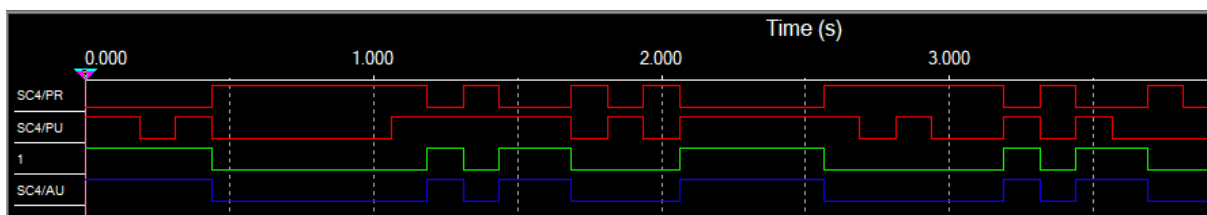


Schemat 9. Układ testujący AU

Poprzedni stan układu równy 0 uzyskujemy poprzez ustawienie $MU=0$, $PU=0$, $PR=0$, a 1 poprzez ustawienie $MU=1$, $PU=1$, $PR=1$.



Rysunek 13. Generator słów dla testu AU



Rysunek 14. Analizator logiczny dla testu AU

Działanie układu zgadza się dla każdego wejścia.

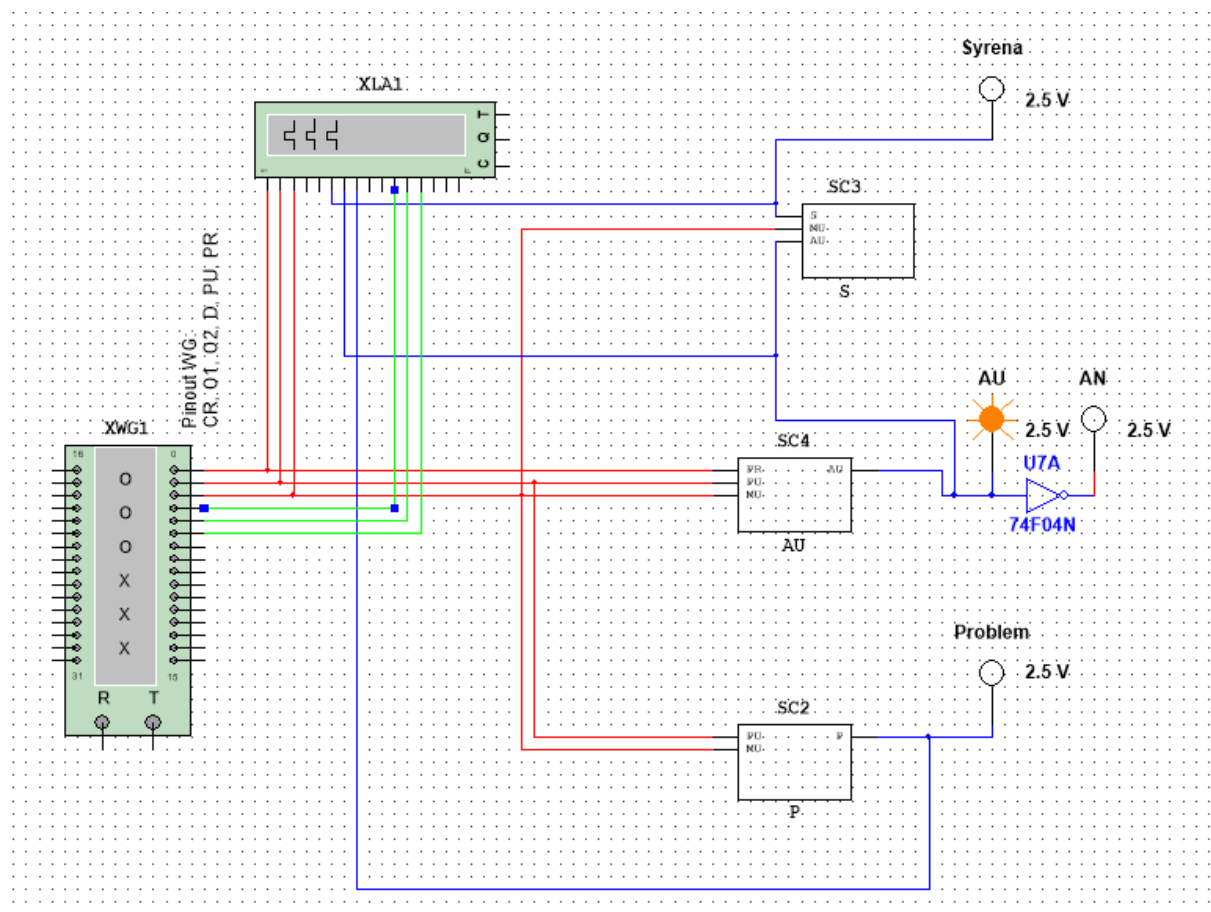
2.4 Test układu

Z racji tego iż przetestowaliśmy podobwody osobno, sprawdzimy jedynie poprawność ich połączenia ze sobą oraz wskazania odpowiednich diod. Podukład AU zastąpimy osobnym wejściem aby zmniejszyć niepotrzebną liczbę rozważanych przypadków. Sam test będzie opierał się na symulacji, którą obrazuje poniższa tabela.

MU	PU	PR	P	AU	S
1	1	0	0	1	0
0	0	0	0	1	1
1	1	0	0	1	1
1	0	1	0	0	0
0	0	1	0	0	0
0	1	0	1	0	0
0	0	0	1	0	0

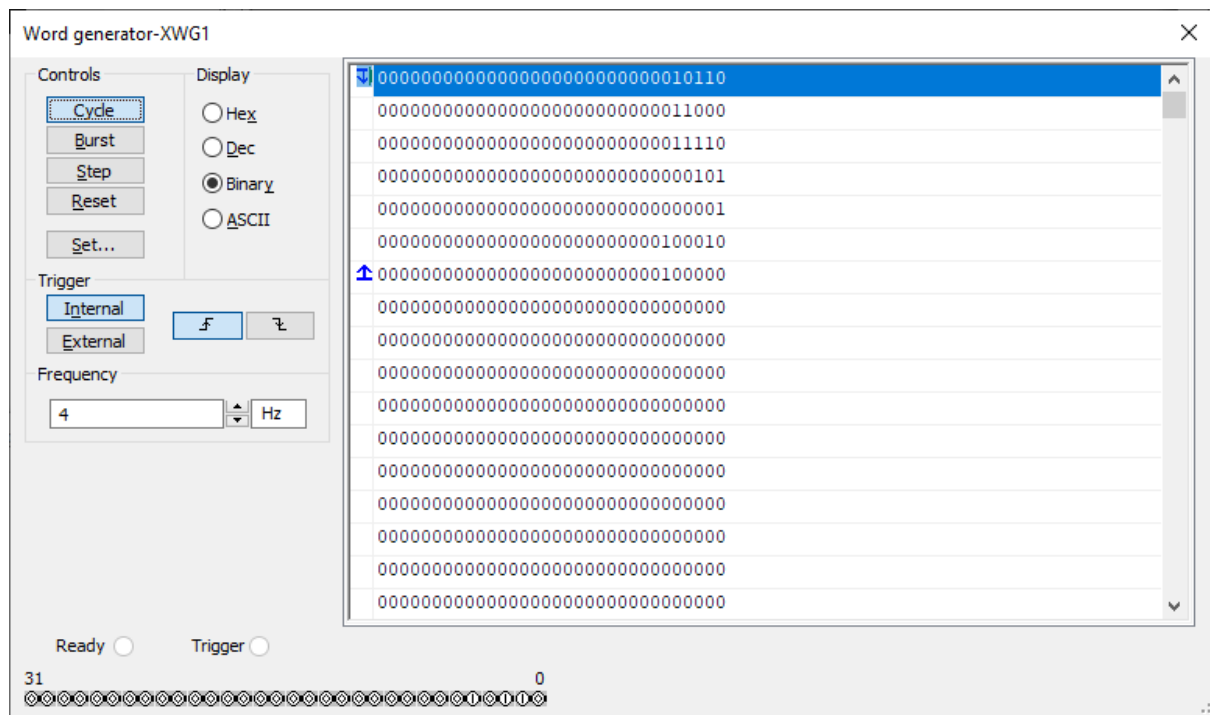
Pierwsze 3 kolumny to wejścia, natomiast 3 ostatnie obrazują żądane wyjścia
Najpierw uzbrajamy alarm, potem następuje włamanie, następnie sprawdzamy czy syrena dalej jest włączona po włamaniu, później dezaktywujemy ją wyłączając uzbrojenie, a potem sprawdzamy czy jesteśmy w stanie włączyć alarm przy włamaniu.

Schemat testujący cały układ wygląda następująco



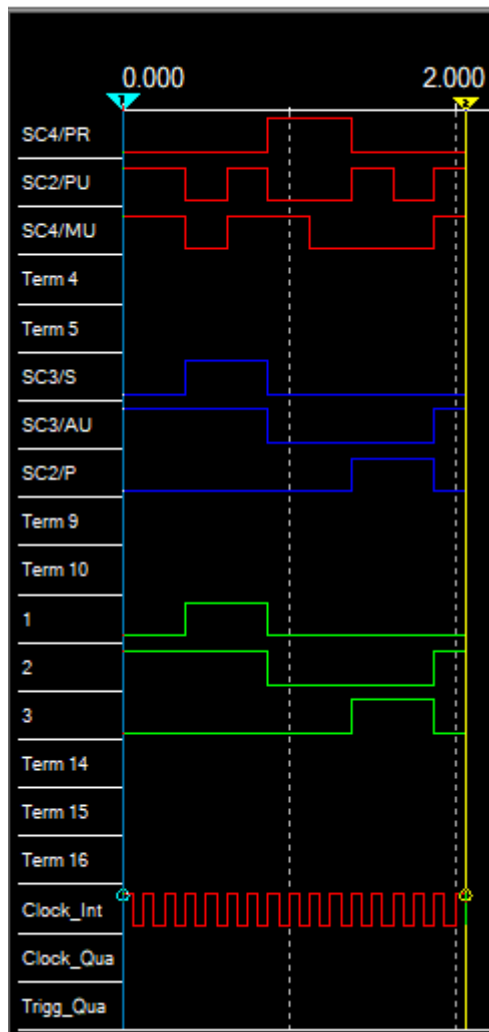
Schemat 10. Układ testujący cały schemat

Generator słów do układu



Rysunek 15. Analizator logiczny dla testu całościowego

Poszczególne stany odzwierciedla analizator logiczny



Rysunek 16. Analizator logiczny dla testu całościowego

Jak widzimy wykresy mają identyczne kształty co dowodzi poprawności układu.

2.5 Wnioski

1. Dzięki zastosowaniu praw logiki jesteśmy w stanie uprościć równania, co skutkuje mniejszym zużyciem bramek logicznych co w rezultacie prowadzi do zmniejszenia kosztów produkcji.
2. Generator słów znacznie ułatwia przeprowadzenie testów utworzonych schematów.
3. Dzielenie układu na podukłady wyraźnie poprawia czytelność schematu oraz ułatwia jego testowanie.
4. Zastosowanie przełącznika jako zamiennika dwóch przycisków znacznie uprościłoby schemat.

5. Tabele prawdy znacznie upraszczają prezentację działania układu.
Dodatkowo są znacznym ułatwieniem przy pisaniu testów.
6. Układ znalazłby zastosowanie jako system alarmowy w przypadku włamania do sklepu, gdzie w momencie włączenia syreny mógłby od razu informować policję o włamaniu.
7. System sprawdziłby się także jako alarm przeciwpożarowy. Mógłby on wówczas zamiast wejść D, O1, O2, CR przyjmować czujniki dymu. Można by wówczas rozważyć uruchomienie kurtyny wodnej wraz z syreną oraz powiadomienie odpowiednich służb o zaistniałej sytuacji.