

Technika Cyfrowa.
Ćwiczenie 1.

Maciej Pieta

Piotr Koproń
Rafał Piwowar

Jakub Woś

Marzec 2023

1 Zadanie 1a

1.1 Treść zadania

Bazując wyłącznie na dwuwejściowych bramkach logicznych NAND, proszę od podstaw zaprojektować, zbudować i przetestować układ realizujący funkcję logiczną:

$$Y = \overline{A} \text{ xor } (B + C) \quad (1)$$

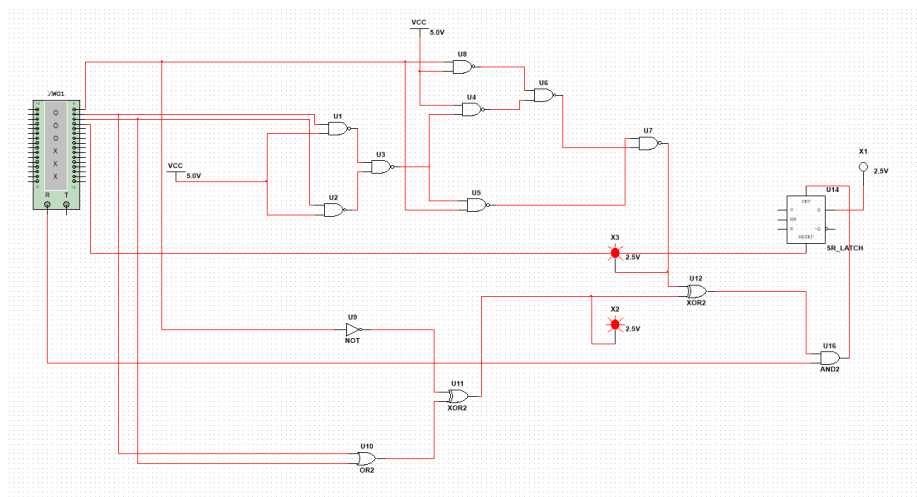
1.2 Rozwiązanie teoretyczne

Dokonujemy następujących przekształceń, korzystając z definicji xor, praw de Morgana oraz prawa podwójnej negacji.

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

Otrzymaliśmy równoważny układ bezpośrednio zapisujący się jako zbiór bramek NAND, przy obserwacji, że $x \text{ NAND } 1 = \overline{x}$.

1.3 Implementacja układu 1a w programie Multisim



Rysunek 1: Górna część układu to układ faktyczny, dolna część - układ testujący. Jeżeli w którymkolwiek momencie wykryta zostanie rozbieżność, przerzutnik po prawej "zapamięta" ten fakt.

Moduł XWG1 został zaprogramowany aby sprawdzał wszystkie możliwe kombinacje wartości A,B,C.

1.4 Wnioski

Należy zauważyć, że bramka NAND jest wystarczająca do utworzenia pełnego systemu logicznego. Tj, dowolną skończoną funkcję logiczną można fizycznie zaimplementować za pomocą skończonej ilości bramek NAND.

2 Zadanie 1b

2.1 Treść zadania

Rozważmy pomieszczenie w którym znajdują się: drzwi wejściowe i 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.

W oparciu o dowolne bramki logiczne, przełączniki i wskaźniki LED, proszę zaprojektować, zminimalizować, zbudować i przetestować układ realizujący funkcję opisanego wyżej systemu alarmowego. Rolę czujników mogą tutaj pełnić dowolne (dostępne w Multisimie) źródła sygnału cyfrowego.

2.2 Rozwiązanie teoretyczne

Łączenie czujników Zauważmy, że z perspektywy alarmu, dowolne naruszenie czujników jest traktowane identycznie - nie interesuje nas to, czy otwarte są drzwi czy okna, w obu przypadkach można się dostać do pomieszczenia. Możemy więc utworzyć sygnał łączny czujników, dalej oznaczony jako A .

Zapamiętywanie stanu Skorzystamy z trzech przerzutników "zapamiętujących" odpowiednio:

- Stan syreny alarmu, reprezentowanej przez LED.
- Stan uzbrojenia alarmu: 0 - rozbrojony, 1 - uzbrojony.
- Stan lampki błędu.

Przyciski Przycisk uzbrojenia uzbroi alarm, lub, jeżeli zachodzi sygnał A, ustawi lampkę błędu. Przycisk rozbrojenia rozbraja alarm i wyłącza syrenę jeżeli jest włączona. Dodatkowo w celu usunięcia stanu nielegalnego, zablokujemy przycisk rozbrojenia jeżeli równocześnie wciśnięty jest przycisk uzbrojenia.

Implementacja logiczna stanów przejścia Niech U oznacza stan uzbrojenia, R - rozbrojenia, E - błędu, S - syreny alarmowej. Niech B oznacza sygnał przycisku uzbrojenia, C - sygnał przycisku rozbrojenia. Wtedy otrzymujemy następujące przejścia:

$$RB\bar{A} \Rightarrow U$$

$$RBA \Rightarrow E$$

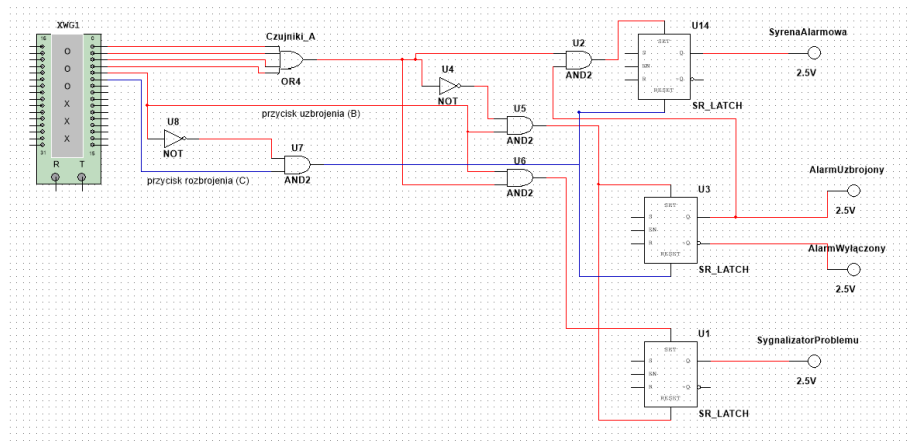
$$EB\bar{A} \Rightarrow U$$

$$UA \Rightarrow S$$

$$UC \Rightarrow R$$

$$SC \Rightarrow R$$

2.3 Implementacja układu 1b w programie Multisim



Rysunek 2: Sygnał U5 opowiada skutecznemu uzbrojeniu, U6 - nieskutecznemu. Sygnał U2 odpowiada za aktywację syreny alarmowej.

2.4 Wnioski

Prostota a praktyka Korzystając tylko z podstawowej wiedzy na temat bramek logicznych oraz przerzutników, jesteśmy w stanie zaprojektować układ stosowany w życiu codziennym. Co więcej, mamy niemalże gwarancję że układ nasz będzie szybszy niż jakiegokolwiek implementacje oparte o programowanie "klasycznego" procesora.

3 przerzutniki W celu zapamiętania możliwych stanów wykorzystaliśmy 3 przerzutniki, zamiast matematycznie niezbędnych dwóch. Zostało to podyktowane założeniem, że każdy przerzutnik składa się z dwóch bramek logicznych. Odtworzenie zachowania na dwóch przerzutnikach wymagałoby użycia większej ilości bramek.