

## **Technika Cyfrowa**

Sprawozdanie z ćwiczenia nr. 3

**Piotr Błaszczuk, Eryk Olejarz, Krzysztof Swędzioł, Filip Zieliński**

18.06.2024

# 1 Treść Zadania

Proszę zaproponować, zbudować i przetestować układ sterujący windą w przykładowym trzykondygnacyjnym budynku.

Winda posiada:

- wskaźnik ruchu windy
- wskaźnik kierunku ruchu windy
- trzy czujniki otwarcia drzwi, po jednym na każdej kondygnacji
- trzy przyciski przywołania windy, po jednym na każdej kondygnacji
- trzy przyciski wyboru piętra w kabinie windy.

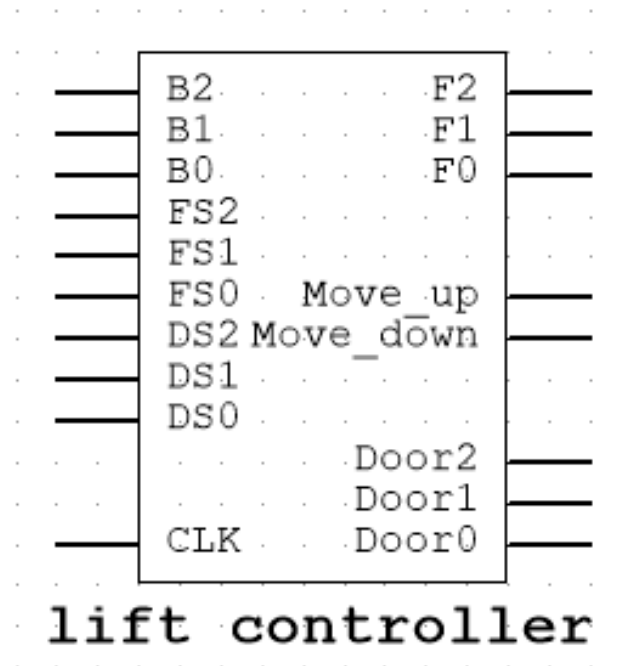
Winda powinna posiadać stale aktualizowany wskaźnik aktualnego piętra.

## 2 Ogólna idea rozwiązania

W celu realizacji zadania zaprojektowaliśmy 3 automaty – jeden do obsługi przycisku, drugi do obsługi drzwi i trzeci do obsługi ruchu windy. Rozwiązanie oparliśmy na przerzutnikach typu D. Do efektywnej minimalizacji Karnaugh wykorzystaliśmy stronę <https://www.charlie-coleman.com/experiments/kmap/>.

### 2.1 Pełny schemat planowanego układu

"Czarna Skrzynka" odpowiedzialna za działanie całości.



### 3 Rozwiązanie - Automaty

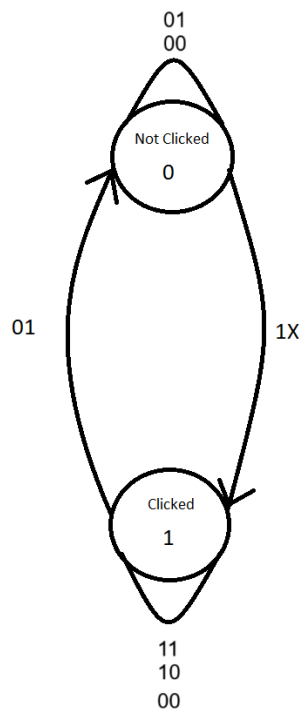
Poniżej przejdziemy przez logikę stojącą za zaimplementowanymi w rozwiązaniu układami.

#### 3.1 Automat odpowiedzialny za przycisk

Mechanizm przywoływania windy można opisać następująco: Gdy przycisk zostanie naciśnięty, system przechodzi do stanu aktywnego i oczekuje, aż winda dojedzie na piętro. W trakcie oczekiwania na dojazd windy, przycisk pozostaje aktywny. Po dotarciu windy na piętro i spełnieniu warunku czujnika, przycisk automatycznie zmienia stan na nieaktywny.

Opis bitów wejść:

- **Pierwszy bit:** Informuje, czy przycisk jest wciśnięty (1 jeśli tak, 0 jeśli nie).
- **Drugi bit:** Czujnik informujący o tym, czy winda znajduje się na piętrze (1), czy też nie (0).



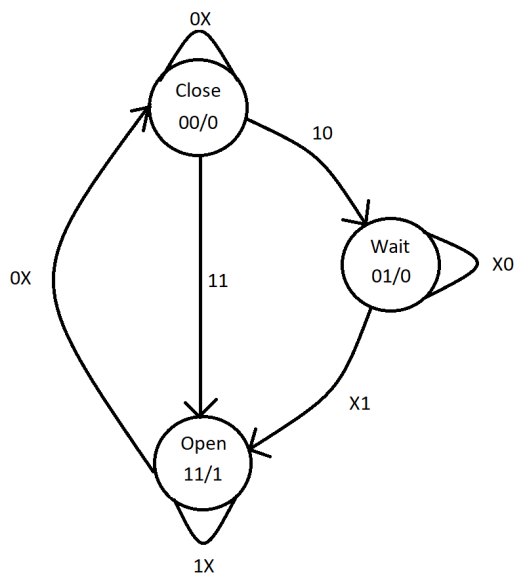
Rysunek 1: Schemat automatu odpowiedzialnego za przycisk

### 3.2 Automat odpowiedzialny za Drzwi

Winda operuje w następujący sposób: Winda sprawdza, czy została wezwana na dane piętro. Jeśli tak, przechodzi do stanu oczekiwania, aż dotrze na to piętro, zamiast zawieszać się gdzieś w połowie drogi. Gdy winda dojedzie na wybrane piętro, drzwi się otwierają natychmiastowo (bez powolnego otwierania).

Opis bitów wejść:

- **Pierwszy bit:** Sygnał przywołania z danego piętra. Informuje, czy przycisk został naciśnięty w celu wezwania windy na określone piętro.
- **Drugi bit:** Czujnik piętra. Informuje, czy winda znajduje się na piętrze, umożliwiając wejście do niej, czy jest między piętrami.



Rysunek 2: Schemat automatu odpowiedzialnego za otwieranie drzwi

### 3.3 Automat odpowiedzialny za mechanizm działania windy

Informacje o położeniu windy: Na wyjściu z każdego piętra znajduje się informacja o aktualnym położeniu windy. Trzy pierwsze bity ciągu XXXXX po znaku „/” reprezentują numer piętra, na którym aktualnie znajduje się winda. Dzięki temu możliwe jest wyświetlenie bieżącej lokalizacji windy. Dwa dalsze bity informują o kierunku ruchu windy (dół/góra) lub jej postój.

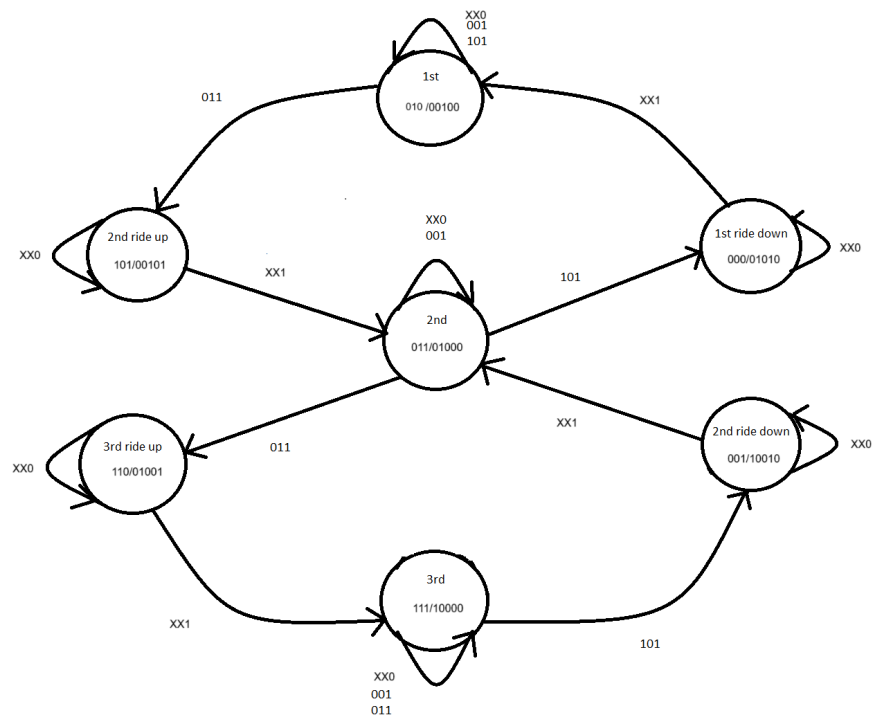
Opis bitów wejść:

- Pierwsze dwa bity: określają kierunek ruchu windy (dół/góra) lub jej postój.
- Trzeci bit: informuje, czy możliwe jest przejście do następnego stanu.

Brak możliwości przejścia do następnego stanu może wynikać z dwóch przyczyn: drzwi windy są niezamknięte lub winda nie znajduje się na piętrze (jest w ruchu).

Automaty sterujące są wzajemnie połączone i skomunikowane, co zapewnia poprawne i bezpieczne działanie całego systemu.

01X, XXX01 - Jazda w górę  
 10X, XXX10 - Jazda w dół  
 00X, XXX00 - postój  
 XX1 - można przejść do następnego stanu  
 XX0 - nie można przejść do następnego stanu

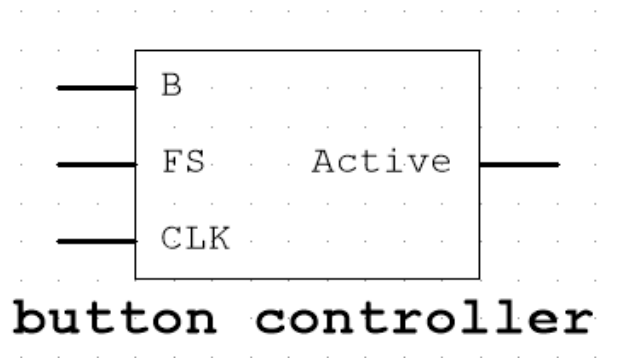


Rysunek 3: Schemat automatu odpowiedzialnego za otwieranie drzwi

## 4 Rozwiązanie - Podukłady

W tej części prezentujemy schematy podukładów i logikę stojącą z implementacją poszczególnych fragmentów w Multisimie. Rozwiązanie oparliśmy na przerzutnikach typu D.

### 4.1 Podukład nr 1 - Schemat przycisku



Rysunek 4: Schemat przycisku kontrolera windy

Tabela przejścia:

Q0 / B, FS	00	01	11	10
0	0	0	1	1
1	1	0	1	1

Tabela 1

Wyliczone wzory:

$$D0 = B + Q0 \cdot \neg FS$$

Tabela wyjścia:

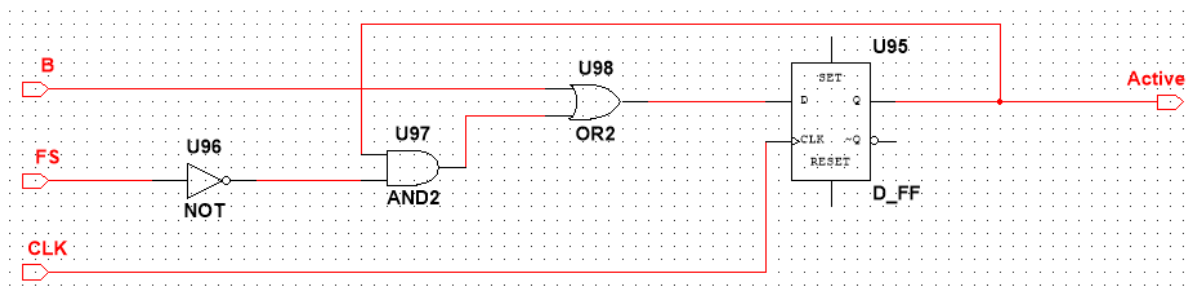
Q0	ACTIVE
0	0
1	1

Tabela 2

Wyliczone wzory:

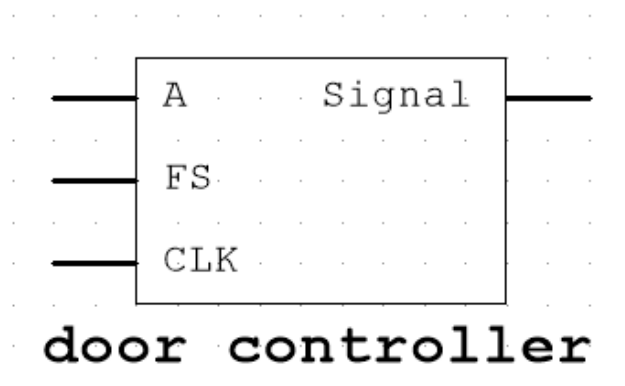
$$ACTIVE = Q0$$

Implementacja w Multisimie:



Rysunek 5: Implementacja w programie Multisim

## 4.2 Podukład nr 2 - Schemat odpowiedzialny za drzwi



Rysunek 6: Schemat kontrolera odpowiedzialnego za drzwi

Tabele przejścia:

Stan / Wejścia	00	01	11	10
00	00	00	11	01
01	01	11	11	01
11	00	00	11	11
10	X	X	X	X

Tabela 3: Tabela przejść dla kontrolera drzwi.

Tabele wyjścia:

Stan	Wyjście
00	0
01	0
11	1
10	X (1)

Tabela 4: Tabela wyjścia dla kontrolera drzwi.

Tablice:

<i>D0</i>		<i>B,FS</i>			
		<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>Q1,Q0</i>	<i>00</i>	0	0	1	1
	<i>01</i>	1	1	1	1
	<i>11</i>	0	0	1	1
	<i>10</i>	-	-	-	-

Rysunek 7: Tablice dla D0

Wzór:

$$D0 = B + \neg Q1 \cdot Q0$$

<i>D1</i>		<i>B,FS</i>			
		00	01	11	10
<i>Q1,Q0</i>	00	0	0	1	0
	01	0	1	1	0
	11	0	0	1	1
	10	-	-	-	-

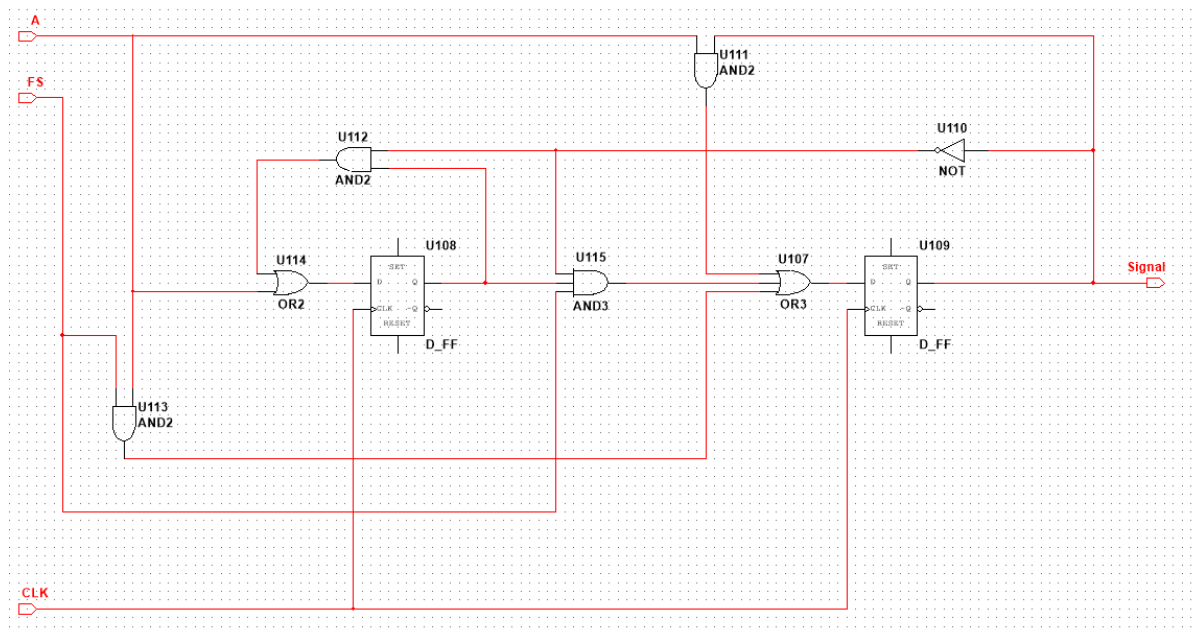
Rysunek 8: Tablice dla D1

$$D1 = BFS + \neg Q1Q0FS + Q1B$$

Wzór na wyjście odczytujemy bezpośrednio z tabeli wyjścia.

$$Signal = Q1$$

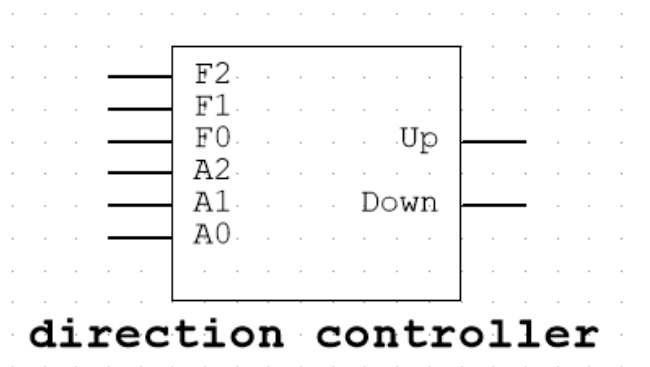
Implementacja w Multisimie:



Rysunek 9: Implementacja w programie Multisim



### 4.3 Podukład nr 3 - kontroler kierunku ruchu windy



Rysunek 10: schemat kontrolera odpowiedzialnego za kierunek poruszania się windy

Tabela prawdy dla kontrolera:

F2	F1	F0	A2	A1	A0	DOWN	UP
0	0	0	0	0	0	x	x
0	0	0	0	0	1	x	x
0	0	0	0	1	1	x	x
0	0	0	0	1	0	x	x
0	0	0	1	1	0	x	x
0	0	0	1	1	1	x	x
0	0	0	1	0	1	x	x
0	0	0	1	0	0	x	x
0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0
0	0	1	1	1	1	0	0
0	0	1	1	1	0	0	1
0	0	1	0	1	0	0	1
0	0	1	0	1	1	0	0
0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0
0	1	1	0	0	0	x	x
0	1	1	0	0	1	x	x
0	1	1	0	1	1	x	x
0	1	1	0	1	0	x	x
0	1	1	1	1	0	x	x
0	1	1	1	1	1	x	x
0	1	1	1	0	1	x	x
0	1	1	1	0	0	x	x
0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0
0	1	0	1	1	1	0	0
0	1	0	1	1	0	0	0
0	1	0	0	1	0	0	0
0	1	0	0	1	1	0	0
0	1	0	0	0	1	0	0
0	1	0	0	0	0	1	0
0	1	0	0	0	0	0	0

Tabela 5: Tabela prawdy dla kontrolera kierunku ruchu windy (część 1)

<b>F2</b>	<b>F1</b>	<b>F0</b>	<b>A2</b>	<b>A1</b>	<b>A0</b>	<b>DOWN</b>	<b>UP</b>
1	1	0	0	0	0	x	x
1	1	0	0	0	1	x	x
1	1	0	0	1	1	x	x
1	1	0	0	1	0	x	x
1	1	0	1	1	0	x	x
1	1	0	1	1	1	x	x
1	1	0	1	0	1	x	x
1	1	0	1	0	0	x	x
1	1	1	1	0	0	x	x
1	1	1	1	0	1	x	x
1	1	1	1	1	1	x	x
1	1	1	1	1	0	x	x
1	1	1	0	1	0	x	x
1	1	1	0	1	1	x	x
1	1	1	0	0	1	x	x
1	1	1	0	0	0	x	x
1	0	1	0	0	0	x	x
1	0	1	0	0	1	x	x
1	0	1	0	1	1	x	x
1	0	1	0	1	0	x	x
1	0	1	1	1	0	x	x
1	0	1	1	1	1	x	x
1	0	1	1	0	1	x	x
1	0	1	1	0	0	x	x
1	0	0	1	0	0	0	0
1	0	0	1	0	1	0	0
1	0	0	1	1	1	0	0
1	0	0	1	1	0	0	0
1	0	0	0	1	0	1	0
1	0	0	0	1	1	1	0
1	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0

Tabela 6: Tabela prawdy dla kontrolera kierunku ruchu windy (część 2)

Tabela dla DOWN:

GO_DOWN		B2,B1,B0							
		000	001	011	010	110	111	101	100
F2,F1,F0	000	-	-	-	-	-	-	-	-
	001	0	0	0	0	0	0	0	0
	011	-	-	-	-	-	-	-	-
	010	0	1	0	0	0	0	1	0
	110	-	-	-	-	-	-	-	-
	111	-	-	-	-	-	-	-	-
	101	-	-	-	-	-	-	-	-
	100	0	1	1	1	0	0	0	0

Rysunek 11: Tabela Karnaugh dla DOWN

Wzór:

$$\text{DOWN} = \neg F2 \cdot \neg F0 \cdot \neg A1 \cdot A0 + \neg F0 \cdot \neg A2 \cdot \neg A1 \cdot A0 + \neg F1 \cdot \neg F0 \cdot \neg A2 \cdot A1$$

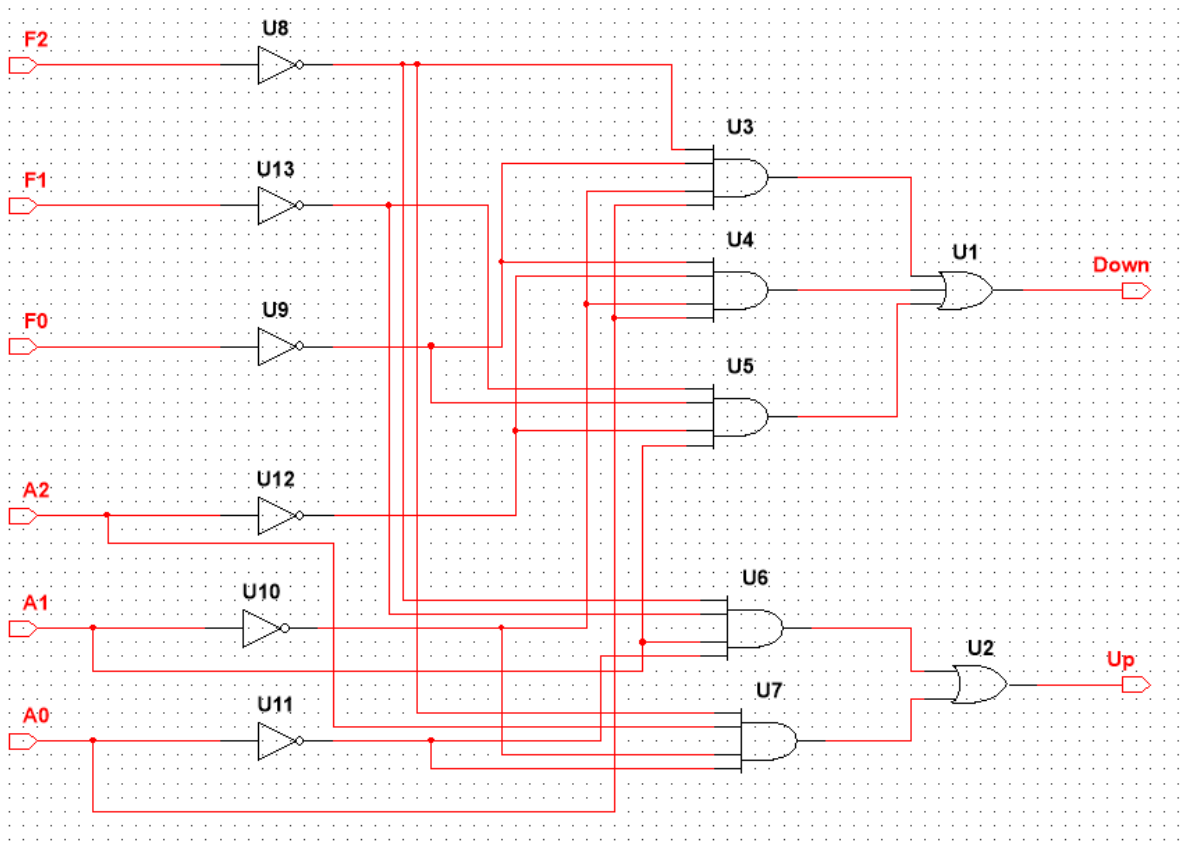
Tabela dla UP:

GO_UP		B2,B1,B0							
		000	001	011	010	110	111	101	100
F2,F1,F0	000	-	-	-	-	-	-	-	-
	001	0	0	0	1	1	0	0	1
	011	-	-	-	-	-	-	-	-
	010	0	0	0	0	0	0	0	1
	110	-	-	-	-	-	-	-	-
	111	-	-	-	-	-	-	-	-
	101	-	-	-	-	-	-	-	-
	100	0	0	0	0	0	0	0	0

Rysunek 12: Tabela Karnaugh dla UP

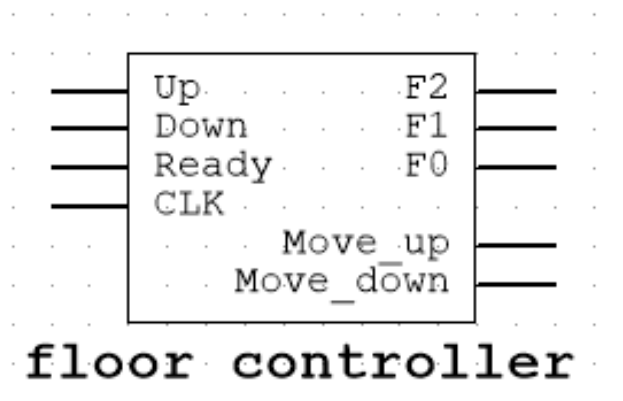
Wzór:

$$\text{UP} = \neg F2 \cdot \neg F1 \cdot A1 \cdot \neg A0 + \neg F2 \cdot A2 \cdot \neg A1 \cdot \neg A0$$



Rysunek 13: Implementacja schematu w multisimie

#### 4.4 Podukład nr 4 - kontroler pięter



Rysunek 14: Ogólny schemat kontrolera pięter

Tabela przejść:

Stan / Wejścia	000	001	011	010	110	111	101	100
000	000	010	010	000	X	X	010	000
001	001	011	011	001	X	X	011	001
011	011	011	110	011	X	X	000	011
010	010	010	101	010	X	X	010	010
110	110	111	111	110	X	X	111	110
111	111	111	111	111	X	X	001	111
101	101	011	011	101	X	X	011	101
100	X	X	X	X	X	X	X	X

Tabela 7: Tabela przejść dla mechanizmu windy.

Tabela wyjść:

Stan	Wyjście
000	01010
001	10010
011	01000
010	00100
110	01001
111	10000
101	10010
100	XXXXX

Tabela 8: Tabela wyjść dla mechanizmu windy.

Tabela dla D0:

D0	DOWN,UP,READY							
	000 001 011 010 110 111 101 100							
Q2,Q1,Q0 000	0	0	0	0	-	-	0	0
001	1	1	1	1	-	-	1	1
011	1	1	0	1	-	-	0	1
010	0	0	1	0	-	-	-	0
110	0	1	1	0	-	-	1	0
111	1	1	-	1	-	-	1	1
101	1	1	1	1	-	-	1	1
100	-	-	-	-	-	-	-	-

Rysunek 15: Tabela dla D0

Wzór:

$$D0 = \neg Q_1 \cdot Q_0 + Q_1 \cdot \neg Q_0 \cdot UP \cdot READY + Q_0 \cdot \neg DOWN \cdot \neg UP + Q_0 \cdot \neg READY + Q_2 \cdot READY$$

Tabela dla D1:

<i>D1</i>		<i>DOWN,UP,READY</i>							
		<i>000 001 011 010 110 111 101 100</i>							
<i>Q2,Q1,Q0</i>	<i>000</i>	0	1	1	0	-	-	1	0
	<i>001</i>	0	1	1	0	-	-	1	0
	<i>011</i>	1	1	1	1	-	-	0	1
	<i>010</i>	1	1	0	1	-	-	-	1
	<i>110</i>	1	1	1	1	-	-	1	1
	<i>111</i>	1	1	-	1	-	-	0	1
	<i>101</i>	0	1	1	0	-	-	0	0
	<i>100</i>	-	-	-	-	-	-	-	-

Rysunek 16: Tabela dla D1

Wzór:

$$D1 = \neg Q_2 \cdot \neg Q_1 \cdot \text{READY} + Q_1 \cdot \neg \text{READY} + \neg Q_0 \cdot \neg \text{UP} \cdot \text{READY} + Q_0 \cdot \neg \text{DOWN} \cdot \text{READY} + Q_2 \cdot Q_1 \cdot \neg \text{DOWN}$$

Tabela dla D2:

<i>D2</i>		<i>DOWN,UP,READY</i>							
		<i>000 001 011 010 110 111 101 100</i>							
<i>Q2,Q1,Q0</i>	<i>000</i>	0	0	0	0	-	-	0	0
	<i>001</i>	0	0	0	0	-	-	0	0
	<i>011</i>	0	0	1	0	-	-	0	0
	<i>010</i>	0	0	1	0	-	-	-	0
	<i>110</i>	1	1	1	1	-	-	1	1
	<i>111</i>	1	1	-	1	-	-	0	1
	<i>101</i>	1	0	0	1	-	-	0	1
	<i>100</i>	-	-	-	-	-	-	-	-

Rysunek 17: Tabela dla D2

Wzór:

$$D2 = Q_1 \cdot \text{UP} \cdot \text{READY} + Q_2 \cdot \neg \text{READY} + Q_2 \cdot Q_1 \cdot \neg \text{DOWN} + Q_1 \cdot \neg Q_0 \cdot \text{DOWN} \cdot \text{READY}$$

Tabela dla F0:

<i>F0</i>		<i>Q1, Q0</i>			
		<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>Q2</i>	<i>0</i>	0	0	0	1
	<i>1</i>	-	1	0	0

Rysunek 18: Tabela dla F0

Wzór:

$$F_0 = \neg Q_2 \cdot Q_1 \cdot \neg Q_0 + Q_2 \cdot \neg Q_1$$

Tabela dla F1:

<i>F1</i>		<i>Q1, Q0</i>			
		<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>Q2</i>	<i>0</i>	1	0	1	0
	<i>1</i>	-	0	0	1

Rysunek 19: Tabela dla F1

Wzór:

$$F_1 = \neg Q_1 \cdot \neg Q_0 + \neg Q_2 \cdot Q_1 \cdot Q_0 + Q_2 \cdot \neg Q_0$$

Tabela dla F2:

<i>F2</i>		<i>Q1,Q0</i>			
		00	01	11	10
<i>Q2</i>	0	0	1	0	0
	1	-	0	1	0

Rysunek 20: Tabela dla F0

Wzór:

$$F_2 = \neg Q_2 \cdot \neg Q_1 \cdot Q_0 + Q_2 \cdot Q_1 \cdot Q_0$$

Tabela dla MOVE DOWN:

<i>GO_DOWN</i>		<i>Q1,Q0</i>			
		00	01	11	10
<i>Q2</i>	0	1	1	0	0
	1	-	0	0	0

Rysunek 21: Tabela dla GO DOWN

Wzór:

$$MOVE\_DOWN = \neg Q_2 \cdot \neg Q_1$$

Tabela dla MOVE UP:

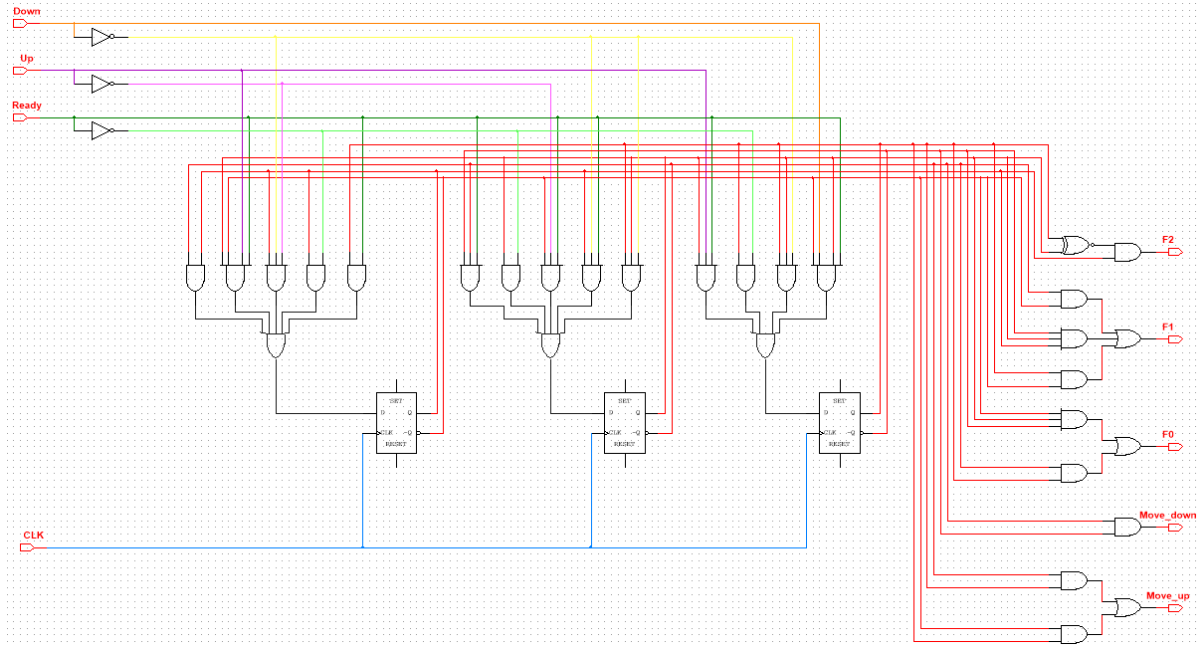
<i>GO_UP</i>		<i>Q1,Q0</i>			
		00	01	11	10
<i>Q2</i>	0	0	0	0	0
	1	-	1	0	1

Rysunek 22: Tabela dla GO UP



Wzór:

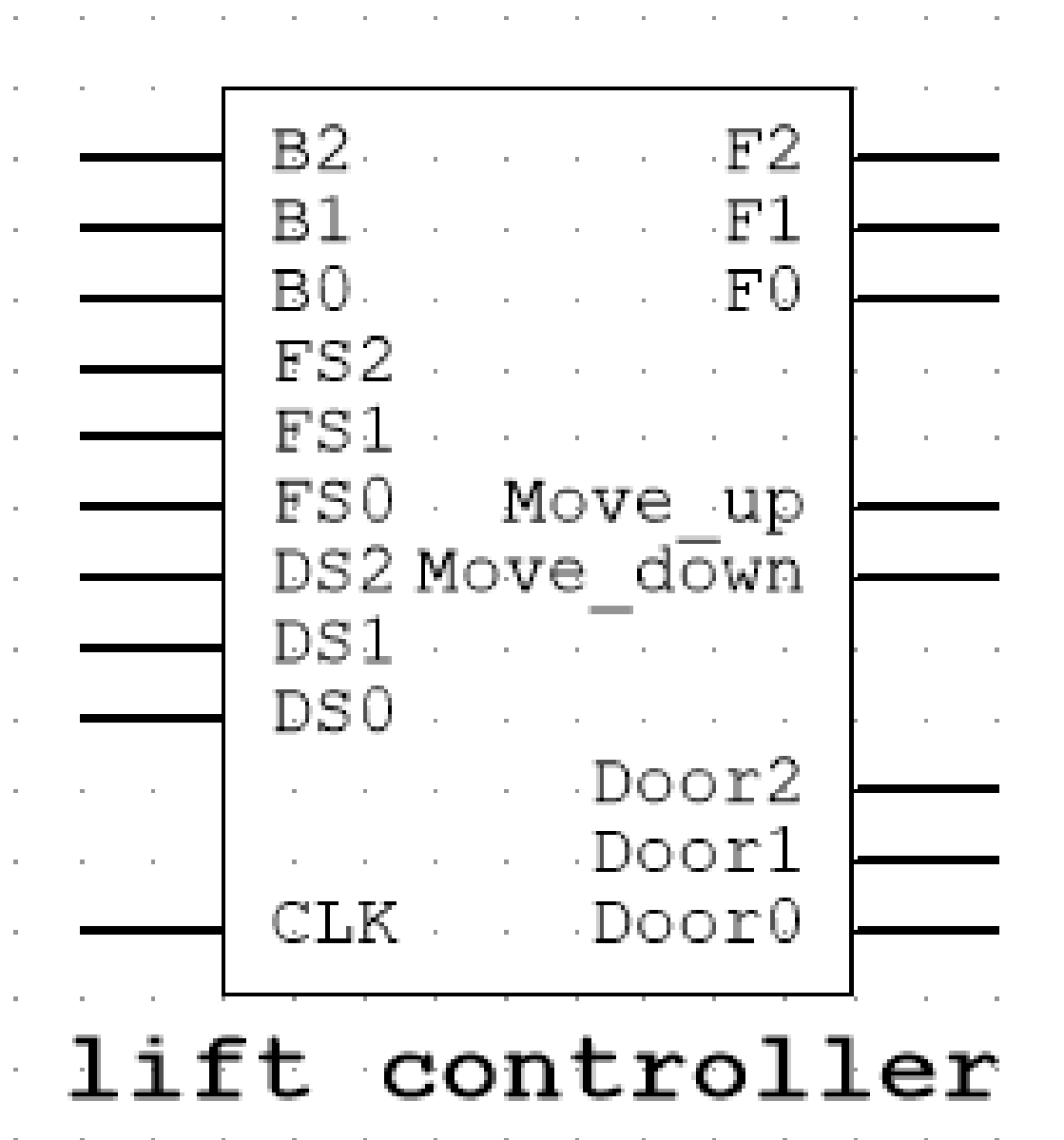
$$MOVE\_UP = Q_2 \cdot \neg Q_1 + Q_2 \cdot \neg Q_0$$



Rysunek 23: Implementacja w multisimie

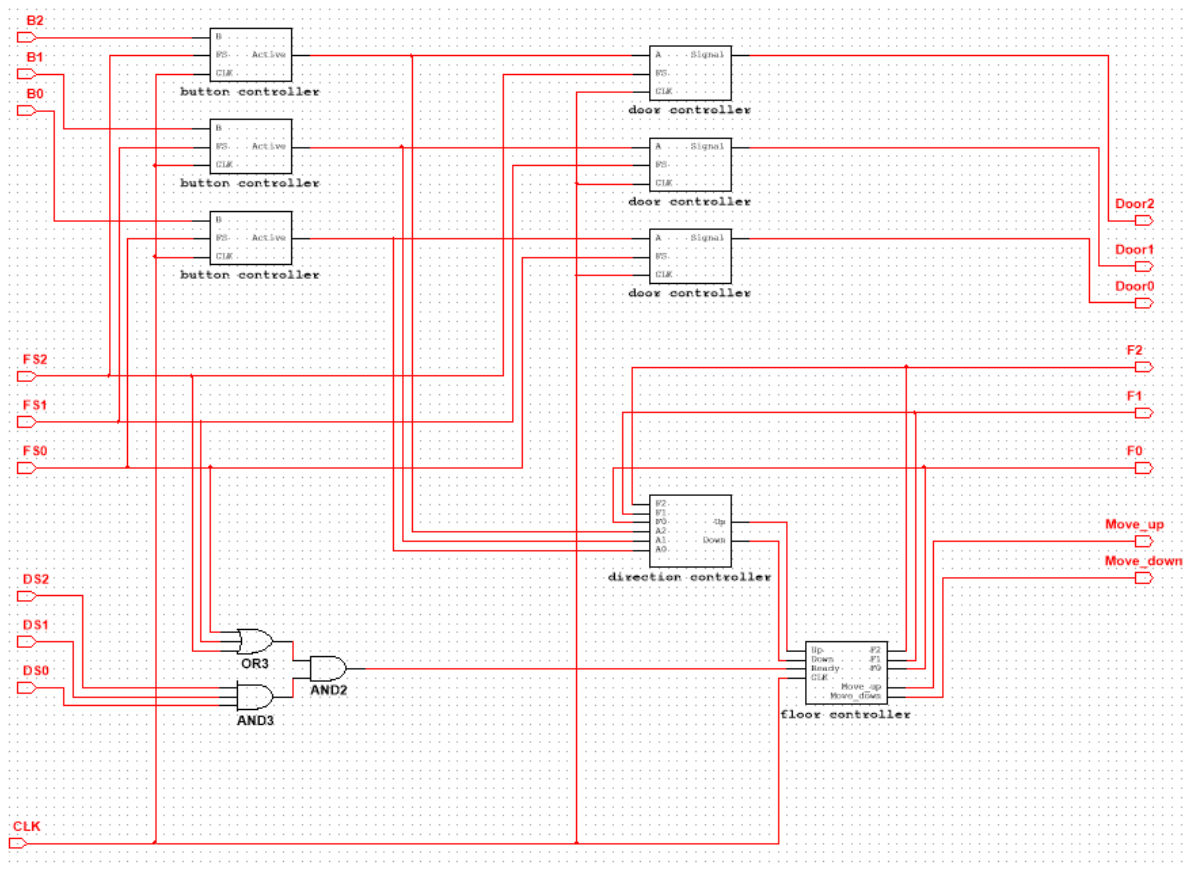
## 5 Cały Układ

Schemat całego układu windy:



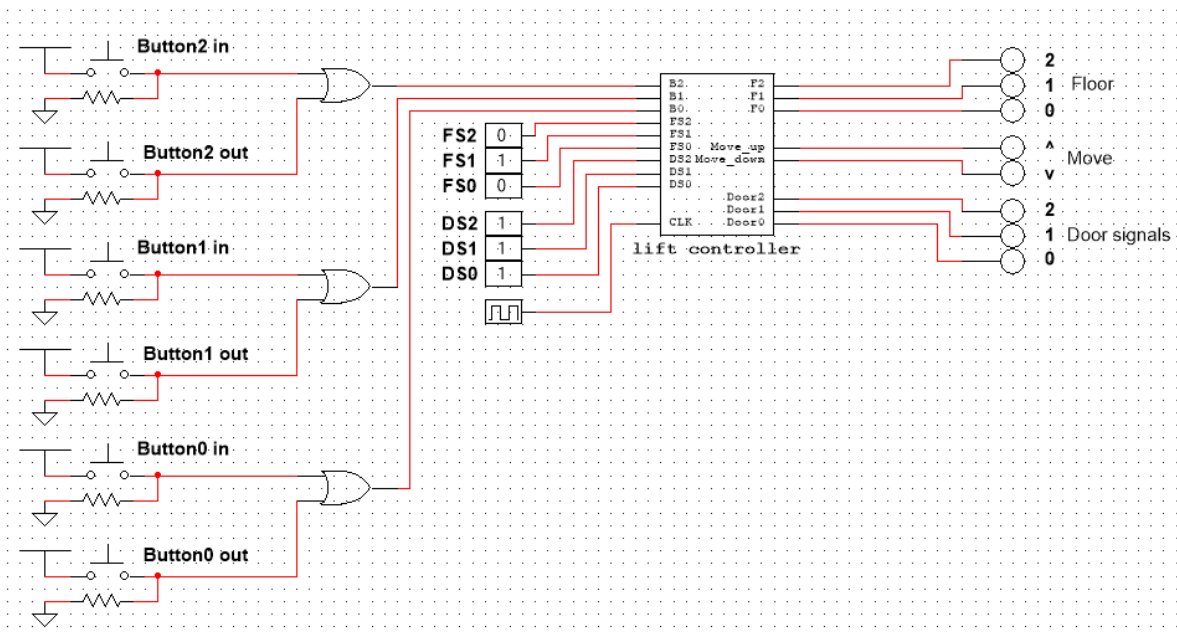
Rysunek 24: Schemat pełnego układu

Implementacja w Multisimie:



Rysunek 25: Pełen układ w Multisimie

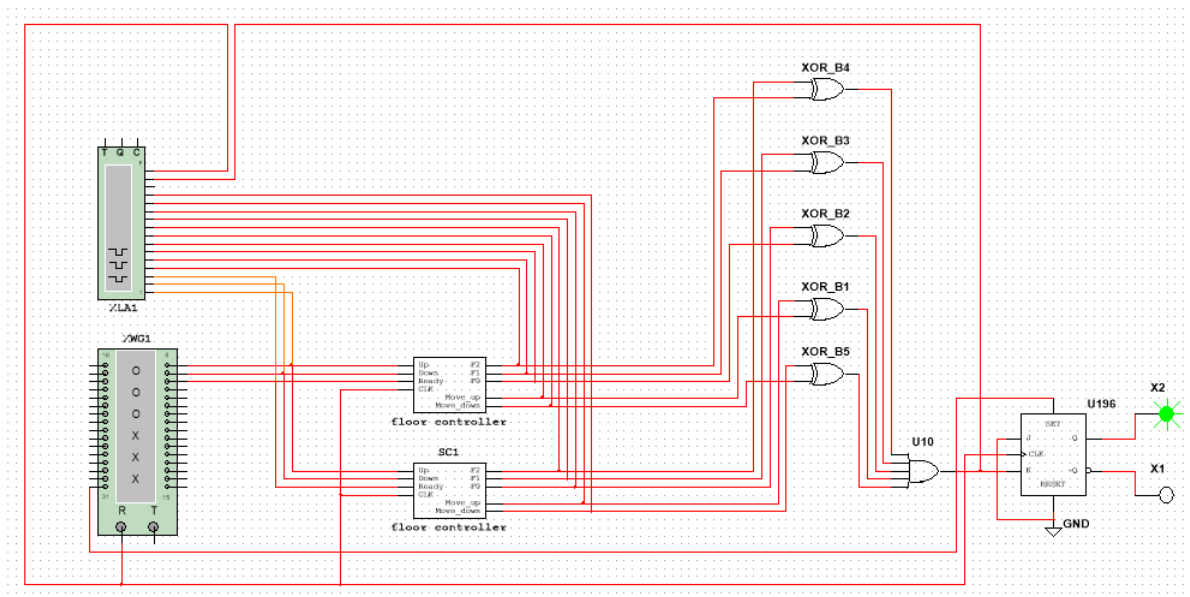
Pełen układ wraz z podłączonymi przyciskami;



Rysunek 26: Pełen układ z przyciskami w Multisimie

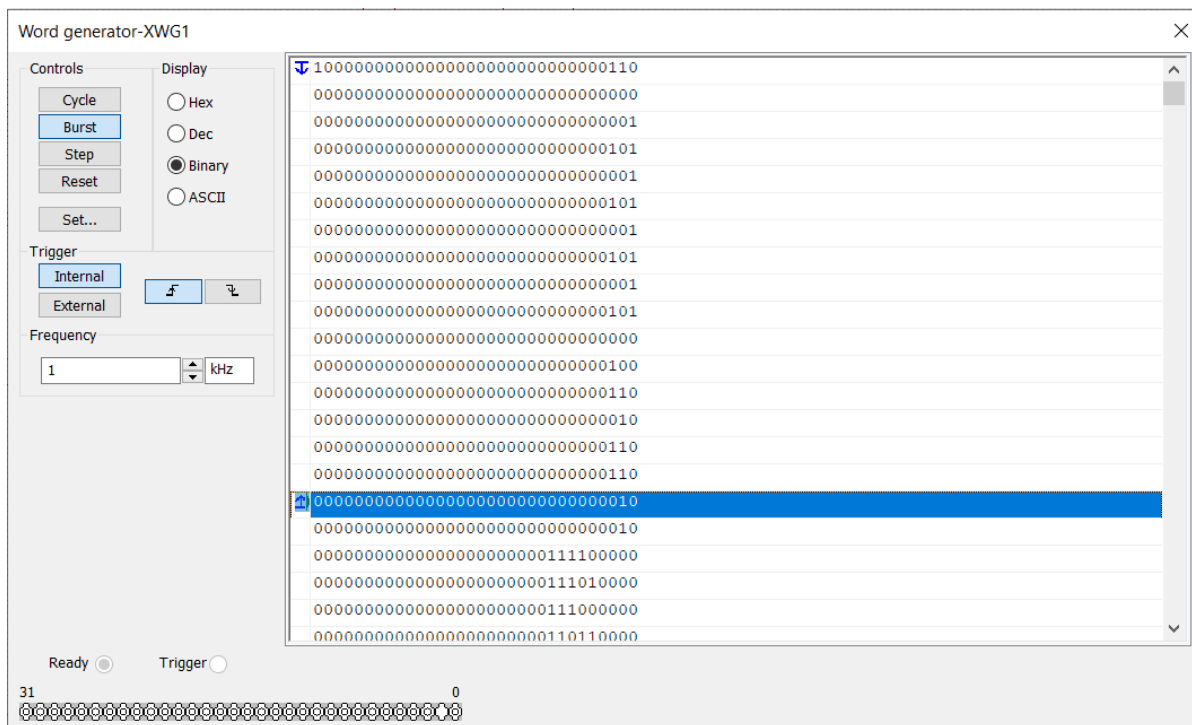
## 6 Układ testujący

Jeżeli posiadamy jeden sprawdzony prototyp, którego działania jesteśmy pewni, możemy testować kolejne układy porównując działanie z już wcześniej sprawdzonym urządzeniem.



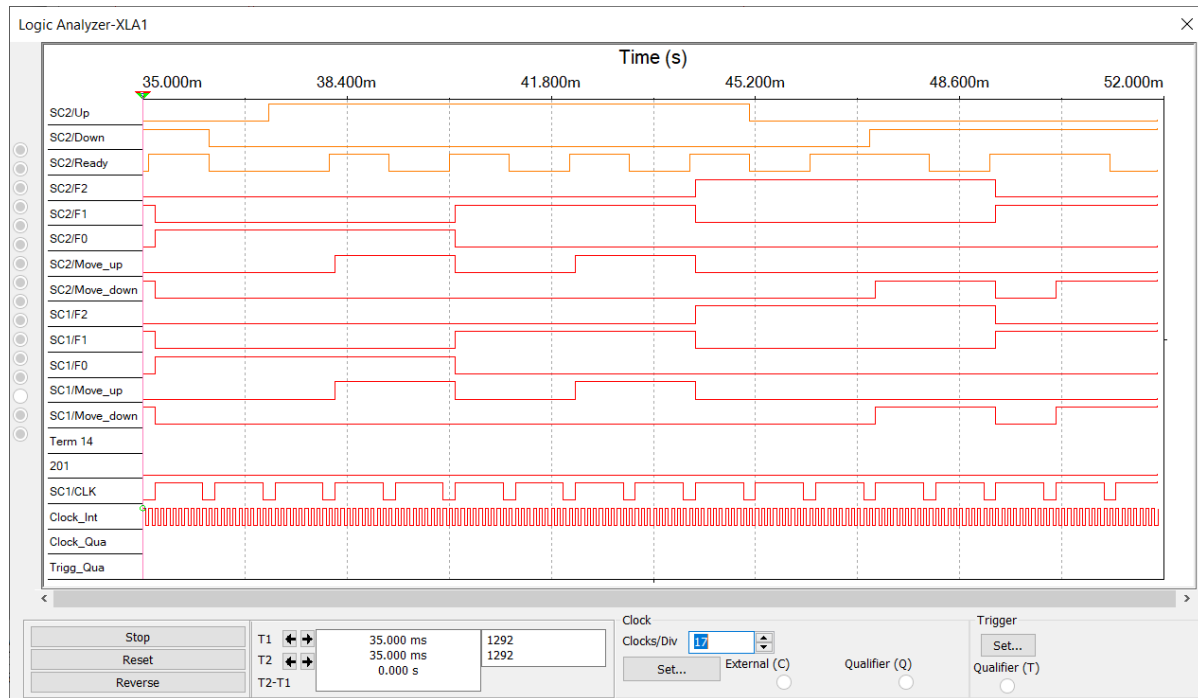
Rysunek 27: Implementacja w Multisimie układu Testującego

Ustawienia generatora słów:



Rysunek 28: Generator słów

Logic Analyzer:



Rysunek 29: Logic Analyzer

Ustawienia zegara:

The screenshot shows the 'Clock Setup' dialog box. It contains the following settings:

- Clock source:** ☐ External, ☒ Internal
- Clock rate:** 10 kHz
- Clock qualifier:** x
- Sampling setting:**
  - Pre-trigger samples: 10
  - Post-trigger samples: 1000
  - Threshold volt. (V): 2.5 V

The 'OK' button is highlighted with a blue border.

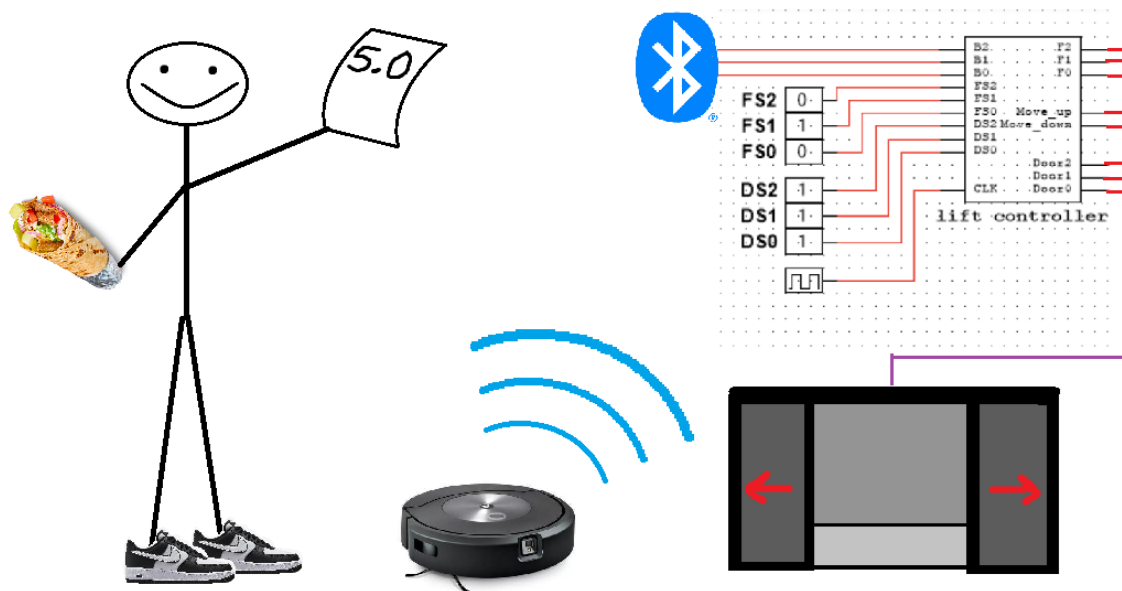
Rysunek 30: Clock setup

## 7 Zastosowania

Wpadliśmy na dwa pomysły, gdzie w praktyce można by wykorzystać nasz układ:

### 7.1 Mała winda dla Roomb'y w D17

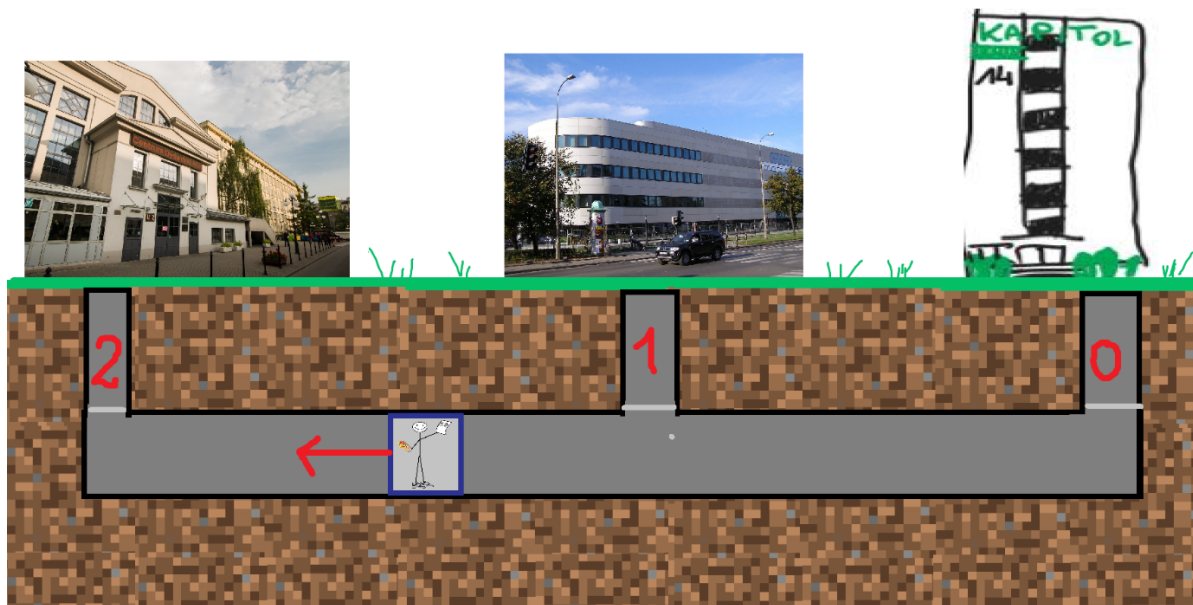
Mała winda, taka w sam raz dla Roomb'y, która cyklicznie w czasie trwania zajęć jeździ po piętrach 1, 2 oraz 3 w D17 (nie na parterze, bo ucieknie przez drzwi). Roomba czyści piętro, i przez bluetooth aktywuje odpowiednie „przyciski” windy, aby dostać się kolejne piętro i je wyczyścić. Mała winda, tylko do użytku dla romby będzie bardziej ekonomiczna w użyciu, i może być przystosowana tylko do niej (aby łatwo się jej wjeżdżało).



Rysunek 31: Prezentacja pomysłu, obok windy przeciętny student dla skali

## 7.2 (PPWA) Podziemna Pozioma Winda AGH

Podziemna winda, która, jak sama nazwa wskazuje, porusza się pod ziemią AGH, i porusza się w poziomie między DS-14 Kapitol, D17 oraz Aulą U-2. Bardzo to ułatwi życie studentom Informatyki, którzy zaoszczędzą na czasie, aby szybko przyjść na zajęcia, albo np. zdążyć na egzamin z matematyki dyskretnej w U-2.



Rysunek 32: Prototyp Podziemnej Poziomej Windy AGH

## 8 Wnioski

- Istnieje wiele sposobów tworzenia automatu, mającego służyć jako winda. Nie jest prosto określić, która z metod jest najlepsza.
- Nasze założenie, że drzwi się otwierają i zamykają natychmiastowo, jest nierealistyczne. Powinniśmy stworzyć jakiś dodatkowy mechanizm, który by zajmował się procesem otwierania drzwi.
- W obecnej chwili, winda priorytezuje niższe piętra. Sprawia to, że może dojść do patologicznej sytuacji, w której winda nigdy nie dotrze na 3 piętro.