**Wprowadzenie do automatyki**

Sprawozdanie z laboratorium nr 3

Temat zajęć: „Projekt układu sterowania z wykorzystaniem PLC i mikrokontrolerów”

Data laboratorium: 16.04.2024

Wykonawca: Kamil Borkowski 83374

Grupa: WCY22IY1S1

Prowadzący zajęcia: mgr inż. Małgorzata Rudnicka

**Treść zadania:**

Zaprojektować układ sterowania silnikiem windy.

Winda porusza się między trzema kondygnacjami.

Zadany jest stan początkowy (zachowanie windy w momencie uruchomienia programu na sterowniku i mikrokontrolerze).

Uwaga: przed uruchomieniem programu na sterowniku winda zostaje ustawiona

na 3 kondygnacji.

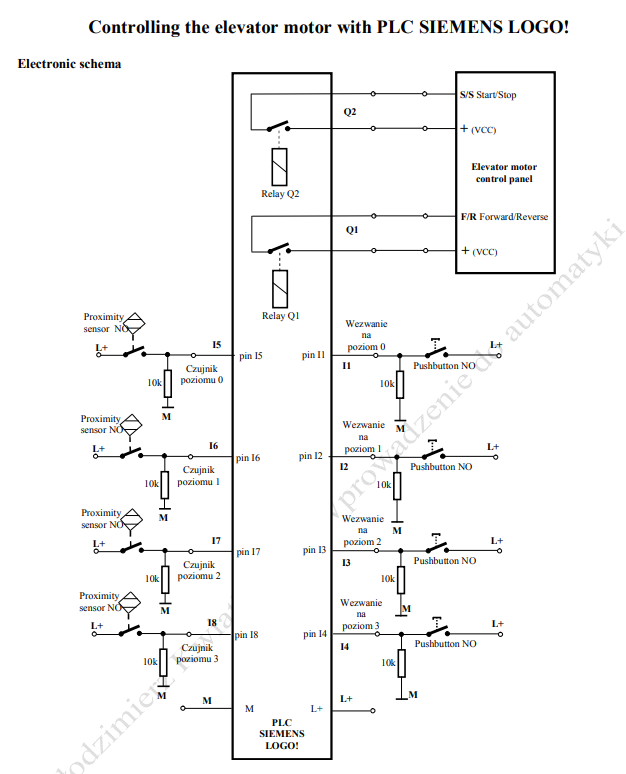
Dane:

Wariant 4

Obsługiwane piętra: 0, 1, 3

Piętro początkowe: 0

**Schemat podłączenia sterownika PLC:**



**Schemat podłączenia Arduino do urządzenia sterowanego:**

**Obraz zawierający tekst, diagram, Plan, Rysunek techniczny

Opis wygenerowany automatycznie**

**Definicja stanów maszyny stanowej:**

Stan M1: winda stoi na piętrze trzecim

Stan M2: winda stoi na piętrze zero

Stan M3: winda jedzie w górę na piętro pierwsze

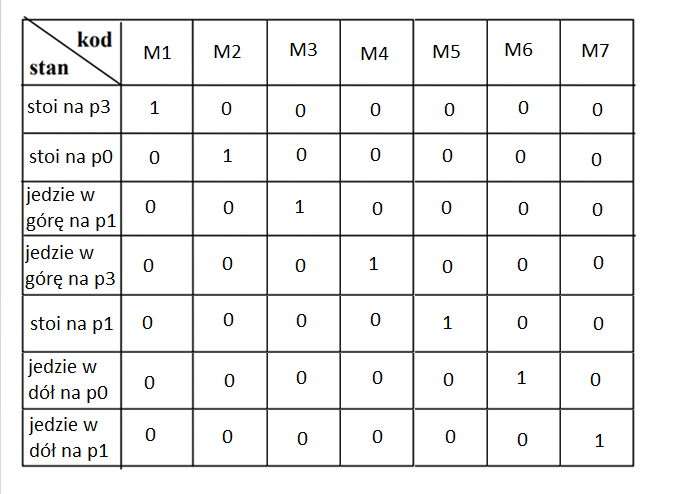
Stan M4: winda jedzie w górę na piętro trzecie

Stan M5: winda stoi na piętrze pierwszym

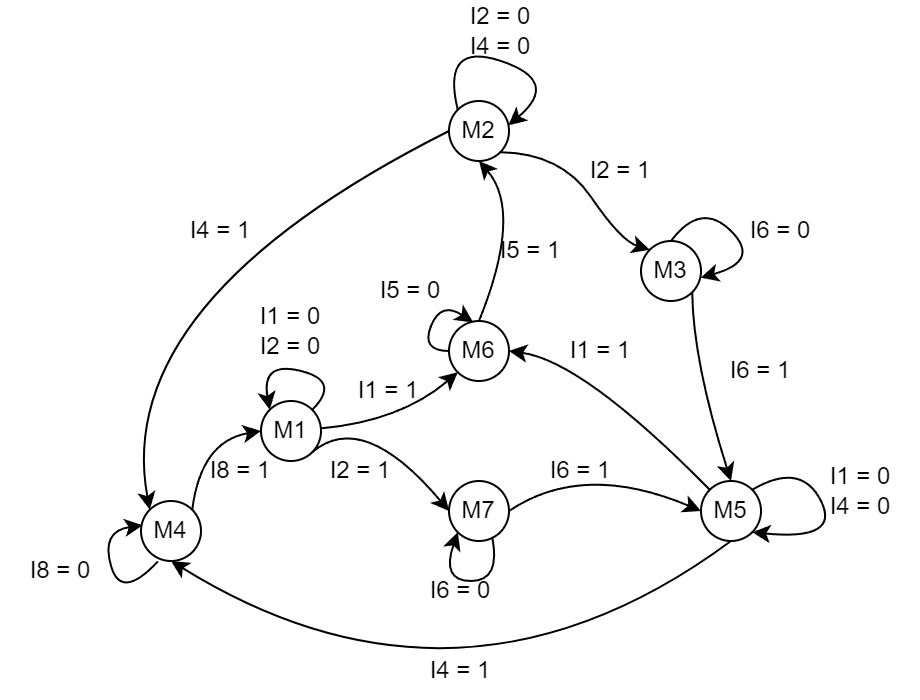
Stan M6: winda jedzie w dół na piętro zero

Stan M7: winda jedzie w dół na piętro pierwsze

**Sposób kodowania stanów:**



**Diagram przejść stanów:**



**Tabela przejść stanów:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| kod stan | M1 | M2 | M3 | M4 | M5 | M6 | M7 | I1 | I2 | I4 | I5 | I6 | I8 | M1' | M2' | M3' | M4' | M5' | M6' | M7' |
|  | 0 | 1 | 0 | 0 | 0 | 0 | 0 | \* | 0 | 0 | \* | \* | \* | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| stoi na p0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | \* | 0 | 1 | \* | \* | \* | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
|  | 0 | 1 | 0 | 0 | 0 | 0 | 0 | \* | 1 | 0 | \* | \* | \* | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| p0->p1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | 0 | \* | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 1 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | 1 | \* | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| p0->p3 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | \* | \* | \* | \* | \* | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| p1->p3 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | \* | \* | \* | \* | \* | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | \* | 0 | \* | \* | \* | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| stoi na p1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | \* | 0 | \* | \* | \* | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
|  | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | \* | 1 | \* | \* | \* | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| p3->p0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | \* | \* | \* | 0 | \* | \* | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| p1->p0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | \* | \* | \* | 1 | \* | \* | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| p3->p1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | \* | \* | \* | \* | 0 | \* | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | \* | \* | \* | \* | 1 | \* | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
|  | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| stoi na p3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | \* | \* | \* | \* | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
|  | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | \* | \* | \* | \* | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| start | 0 | 0 | 0 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | \* | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | \* | \* | \* | \* | \* | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

**Wyrażenia algebraiczne do obliczania wartości funkcji przejścia:**

M1’ = M4I8 + M1

M2’ = M2 + M6I5

M3’ = M2I2 + M3

M4’ = M2I4 + M4 + M5

M5’ = M3I6 + M5 + M7I6

M6’ = I8 + M5I1 + M6 + M1I1

M7’ = M7 + M1I2

**Tabela funkcji wyjścia:**

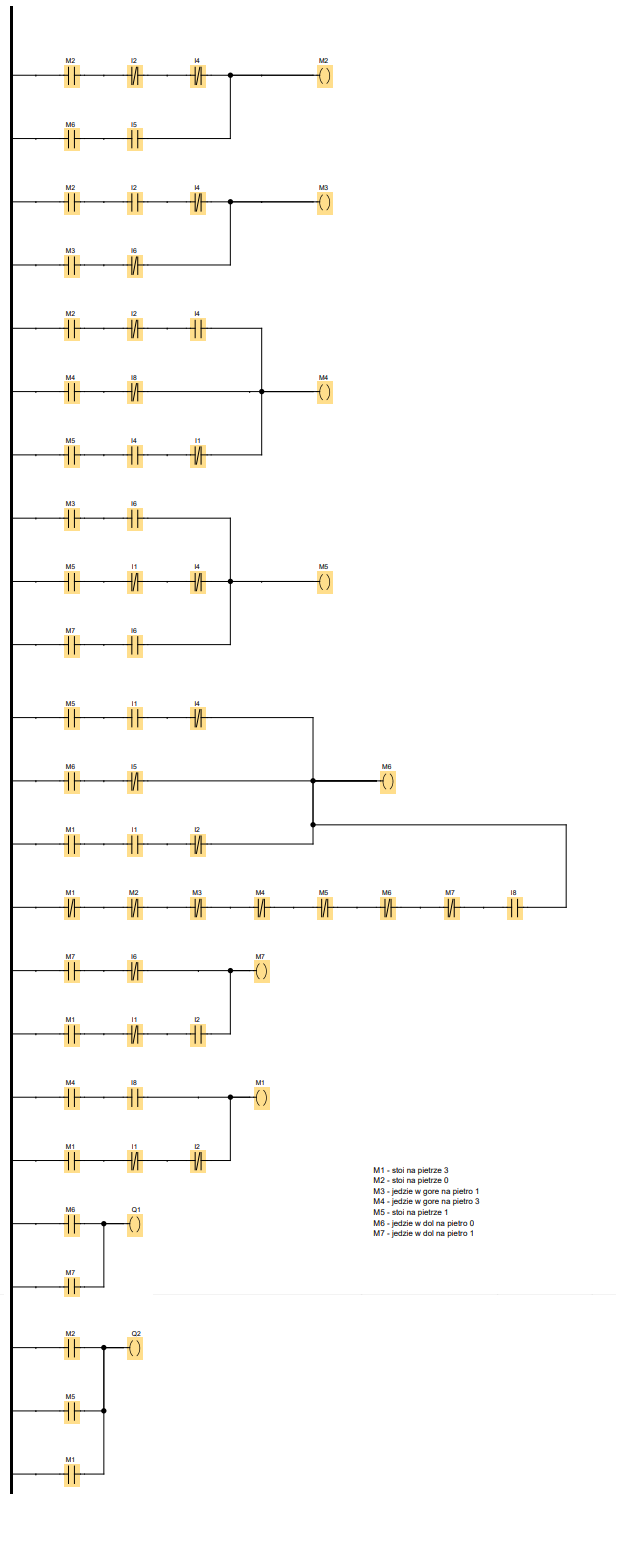
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | M1 | M2 | M3 | M4 | M5 | M6 | M7 |
| Q1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Q2 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |

**Wyrażenia algebraiczne do obliczania wartości funkcji wyjścia:**

Q1 = M6 + M7

Q2 = M1 + M2 + M5

**Wydruk programu w języku LD: Wydruk programu w języku FBD:**

**Obraz zawierający diagram, linia, Plan, Rysunek techniczny

Opis wygenerowany automatycznie**

**Tabulogram programu („szkicu”) realizujący projekt na zestawie Arduino:**

/\*

KAMIL BORKOWSKI WCY22IY1S1

wariant 4

pietra 0 1 3

start 0

M1 - jest na pietrze 3

M2 - jest na pietrze 0

M3 - jedzie w gore na pietro 1

M4 - jedzie w gore na pietro 3

M5 - jest na pietrze 1

M6 - jedzie w dol na pietro 0

M7 - jedzie w dol na pietro 1

\*/

#define czyRaport 1

#define lag 0

//Przyłacza Arduino

int Button1Pin = A0; //nazwa pinu dla "Przycisk żądanie 0-go poziomu"

int Button2Pin = A1; //nazwa pinu dla "Przycisk żądanie 1-go poziomu"

int Button3Pin = A3; //nazwa pinu dla "Przycisk żądanie 2-go poziomu"

int Sensor1Pin = A4; //nazwa pinu dla "Sensor obecności kabiny na 0-ym poziomie"

int Sensor2Pin = A5; //nazwa pinu dla "Sensor obecności kabiny na 1-im poziomie"

int Sensor3Pin = 13; //nazwa pinu dla "Sensor obecności kabiny na 3-im poziomie"

int Relay4Pin = 4;//numer pinu dla Control Signal of Relay 4 / Output Signal Q2 / Start/Stop Signal

int Relay3Pin = 5;//numer pinu dla Control Signal of Relay 3 / Output Signal Q1 / Forward/Reverse Signal

//Zmienne globalne

boolean Q1 = 0;//Forward/Reverse Control Signal Relay 3

boolean Q2 = 0;//Start/Stop Control Signal Relay 4

boolean I1 = 0;//Odczyt z "Przycisk żądanie 0-go poziomu"

boolean I2 = 0;//Odczyt z "Przycisk żądanie 1-go poziomu"

boolean I4 = 0;//Odczyt z "Przycisk żądanie 3-go poziomu"

boolean I5 = 0;//Odczyt z "Sensor obecności kabiny na 0. poziomie"

boolean I6 = 0;//Odczyt z "Sensor obecności kabiny na 1. poziomie"

boolean I8 = 0;//Odczyt z "Sensor obecności kabiny na 3. poziomie"

//Flagi stanow

boolean M1 = 0;

boolean M2 = 0;

boolean M3 = 0;

boolean M4 = 0;

boolean M5 = 0;

boolean M6 = 0;

boolean M7 = 0;

boolean M1p = 0;

boolean M2p = 0;

boolean M3p = 0;

boolean M4p = 0;

boolean M5p = 0;

boolean M6p = 0;

boolean M7p = 0;

void raport()

{

Serial.println("Odczyt przyciskow ");

Serial.print(I1);

Serial.print(I2);

Serial.print(I4);

Serial.println();

Serial.println("Odczyt sensorow ");

Serial.print(I5);

Serial.print(I6);

Serial.print(I8);

Serial.println();

Serial.println("Flagi ");

Serial.print(M1);

Serial.print(M2);

Serial.print(M3);

Serial.print(M4);

Serial.print(M5);

Serial.print(M6);

Serial.print(M7);

Serial.println();

Serial.println("Wyjscia ");

Serial.print(Q1);

Serial.print(Q2);

Serial.println();

}

void odczytWejsc()

{

I1 = digitalRead(Button1Pin);

I2 = digitalRead(Button2Pin);

I4 = digitalRead(Button3Pin);

I5 = digitalRead(Sensor1Pin);

I6 = digitalRead(Sensor2Pin);

I8 = digitalRead(Sensor3Pin);

}

void funkcjaPrzejscia()

{

//Obliczenie wartości funkcji przejśc stanów (flag)

M1p = (M4 & I8) | (M1 & !I1 & !I2);

M2p = (M2 & !I2 & !I4) | (M6 & I5);

M3p = (M2 & I2 & !I4) | (M3 & !I6);

M4p = (M2 & !I2 & I4) | (M4 & !I8) | (M5 & I4 & !I1);

M5p = (M3 & I6) | (M5 & !I1 & !I4) | (M7 & I6);

M6p = (!M1 & !M2 & !M3 & !M4 & !M5 & !M6 & !M7 & I8) | (M5 & I1 & !I4) | (M6 & !I5) | (M1 & I1 & !I2);

M7p = (M7 & !I6) | (M1 & !I1 & I2);

//Przepisanie "nowych" wartości do "starych"

M1 = M1p;

M2 = M2p;

M3 = M3p;

M4 = M4p;

M5 = M5p;

M6 = M6p;

M7 = M7p;

}

void funkcjaWyjscia()

{

Q1 = M6 | M7;

Q2 = M2 | M5 | M1;

}

void zapisWyjscia()

{

digitalWrite(Relay3Pin, Q1);

digitalWrite(Relay4Pin, Q2);

}

void setup()

{

pinMode(Button1Pin, INPUT);

pinMode(Button2Pin, INPUT);

pinMode(Button3Pin, INPUT);

pinMode(Sensor1Pin, INPUT);

pinMode(Sensor2Pin, INPUT);

pinMode(Sensor3Pin, INPUT);

pinMode(Relay3Pin, OUTPUT);

pinMode(Relay4Pin, OUTPUT);

Serial.begin(9600);

Serial.println("Koniec wykonywania Setup");

}

void loop()

{

odczytWejsc();

funkcjaPrzejscia();

funkcjaWyjscia();

zapisWyjscia();

raport();

}

**Wyniki:**

Poprawność działania programu w języku LD i FBD została sprawdzona za pomocą LOGO!Soft Comfort: Po włączeniu programu winda jedzie piętro trzecie:

 Po dotarciu na piętro trzecie winda jedzie na piętro początkowe, w moim przypadku na piętro 0:

Obraz zawierający zrzut ekranu, tekst, Czcionka, design

Opis wygenerowany automatycznie

Po dojechaniu na piętro 0 winda się zatrzymuje i czeka na podanie piętra na które ma się udać, z wyjątkiem piętra 2, które nie jest obsługiwane:

Obraz zawierający zrzut ekranu, tekst, Czcionka, design

Opis wygenerowany automatycznie

Dla przykładu winda zostaje wysłana na piętro 1:

Obraz zawierający zrzut ekranu, Czcionka, tekst

Opis wygenerowany automatycznie

Na które dojeżdża i czeka na kolejne wytyczne:

Obraz zawierający tekst, Czcionka, zrzut ekranu

Opis wygenerowany automatycznie

Poprawność działania szkicu została sprawdzona za pomocą strony wokwi.com

Niebieska dioda odpowiada Q1, a czerwona Q2

Po włączeniu programu winda jedzie piętro trzecie:

Obraz zawierający zrzut ekranu, diagram, Wielobarwność, modelowanie 3D

Opis wygenerowany automatycznie

Po dotarciu na piętro trzecie winda Po dojechaniu na piętro 0 winda się jedzie na piętro początkowe, w moim zatrzymuje i czeka na podanie przypadku na piętro 0: piętra na które ma się udać, z wyjątkiem p piętra 2, które nie jest obsługiwane:

Obraz zawierający Wielobarwność

Opis wygenerowany automatycznie Obraz zawierający Wielobarwność

Opis wygenerowany automatycznie

Dla przykładu winda zostaje Na które dojeżdża i czeka wysłana na piętro 1: na kolejne wytyczne:

 Obraz zawierający Wielobarwność

Opis wygenerowany automatycznie

Wnioski: Program działa zgodnie z założeniami zadania. Gdy winda porusza się między piętrami zatrzyma się dopiero, gdy dojedzie do zadanego przez użytkownika piętra. Q1 oraz Q2 informuję nas jaki ruch wykonuje winda, jeśli tylko Q1 = 1 to znaczy, że winda porusza się w dół, jeśli tylko Q2 = 1 to znaczy, że winda stoi na piętrze, jeśli oba równają się 0 to winda porusza się w górę, jeśli oba równają się 1 to znaczy że program źle działa. Na szczęście w moim programie ostatni przypadek nie występuje.