

Sprawozdanie

Systemy wbudowane

Część 2

**Temat: „Sterowanie modelem windy
pasażerskiej”**

Michał Maciej

L3

167814

Opis ćwiczenia:

Sterowanie modelem windy polega na realizacji obsługi 4 dostępnych pięter windy. Wykorzystywane będą tylko przyciski przywoławcze na poszczególnych piętrach wraz z diodami sygnalizującymi. Po uruchomieniu sterownika winda powinna przemieścić się na piętro najniższe. Należy zrealizować zapamiętanie przywołania windy na poszczególne piętra - wciśnięcie przycisku przywoławczego na danym piętrze powinno spowodować zaświecenie odpowiedniej diody LED. Kabina windy powinna poruszać się pomiędzy piętrami dla których nastąpiło przywołanie, zatrzymując się na danym piętrze przez ok. 3s.

Wejścia obiektu:

KZ1-KZ4: czujniki kontaktronowe zatrzymujące kabinę windy na piętrach 1-4

P1-P4: przyciski przywołujące windę na piętrach 1-4

Wyjścia obiektu:

PGA: uruchomienie kabiny windy - jazda w górę

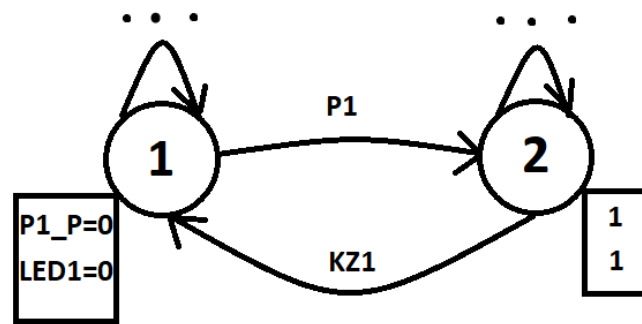
PDA: uruchomienie kabiny windy - jazda w dół

LED1-LED4: diody LED dla przycisków przywoławczych na piętrach 1-4



Widok medulu windy pasażera

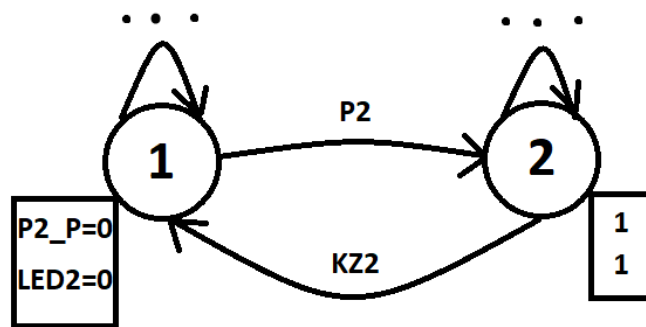
Grafy:



Graf obsługi LED1 i P1

Opisy stanów:

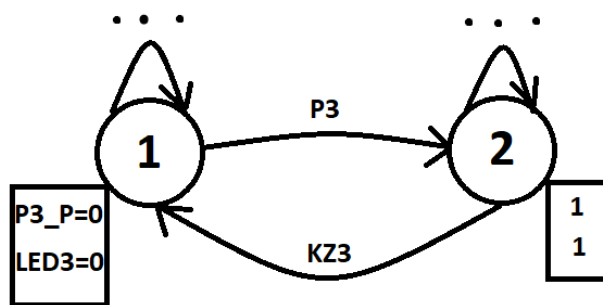
1. Oczekiwanie na wciśnięcie P1
2. Oczekiwanie na zatrzymanie się na pierwszym piętrze



Graf obsługi LED2 i P2

Opisy stanów:

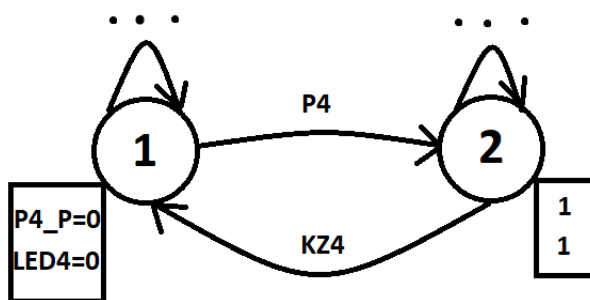
1. Oczekiwanie na wciśnięcie P2
2. Oczekiwanie na zatrzymanie się na drugim piętrze



Graf obsługi LED3 i P3

Opisy stanów:

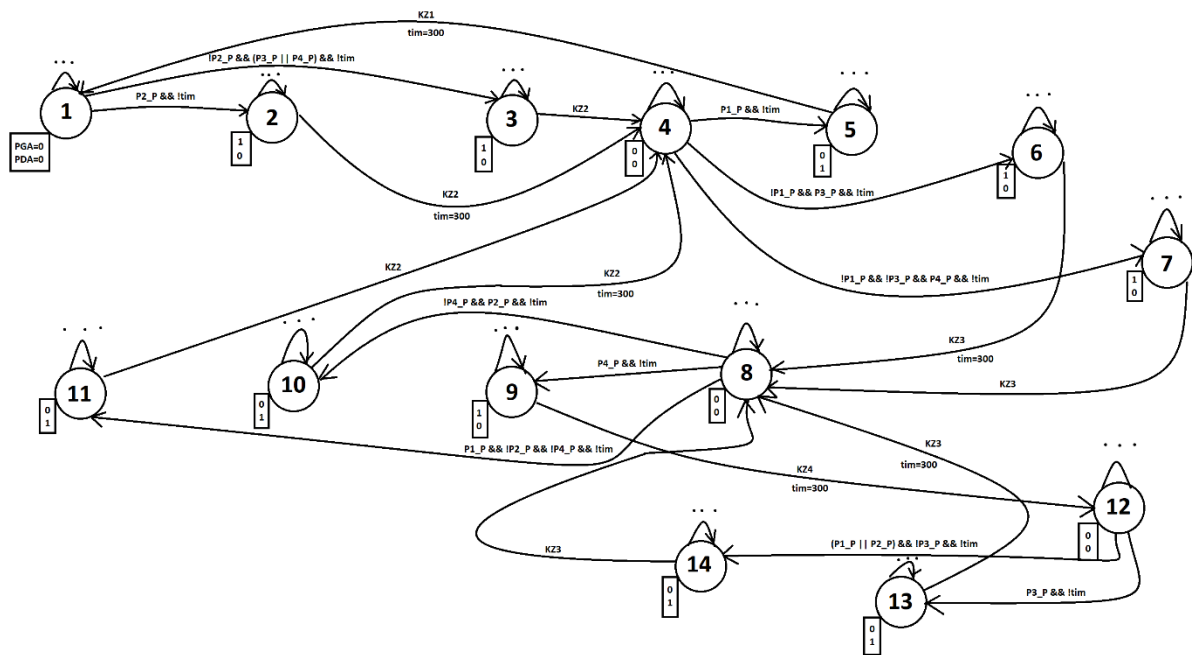
1. Oczekiwanie na wciśnięcie P3
2. Oczekiwanie na zatrzymanie się na trzecim piętrze



Graf obsługi LED4 i P4

Opisy stanów:

1. Oczekiwanie na wciśnięcie P4
2. Oczekiwanie na zatrzymanie się na czwartym piętrze

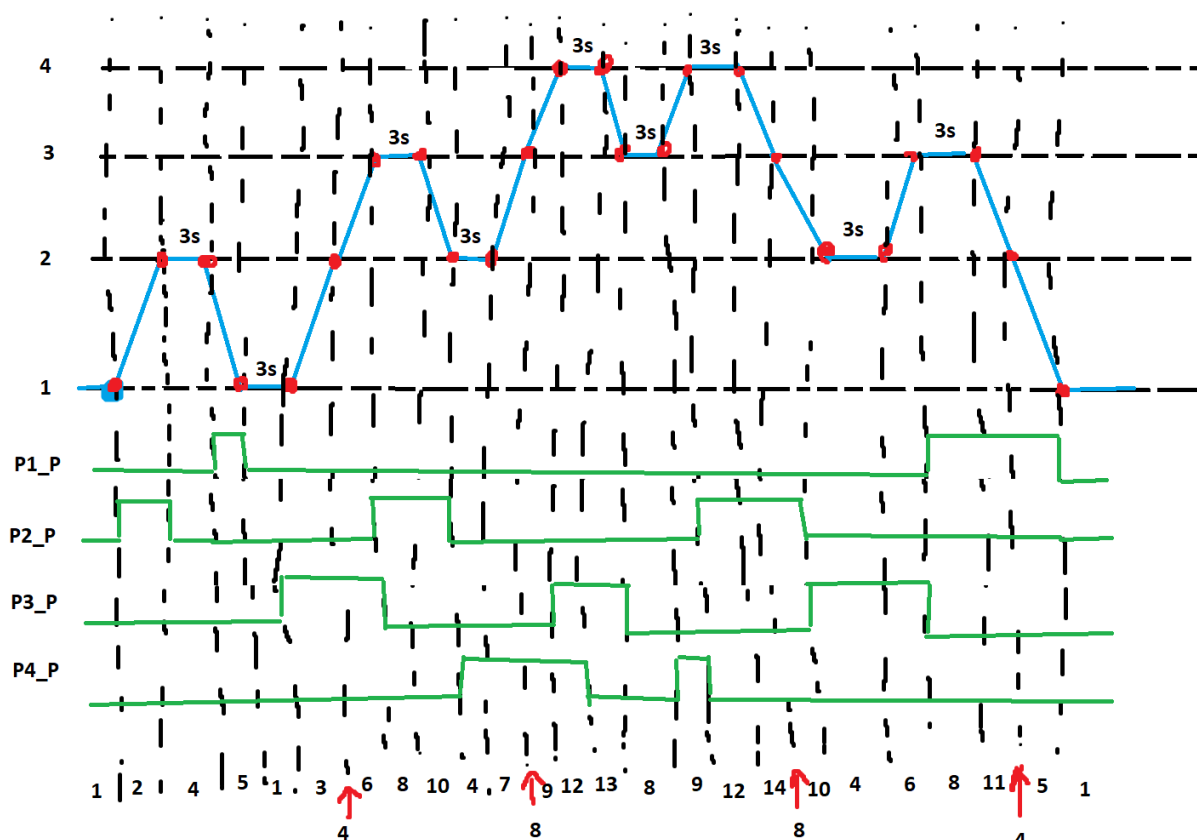


Graf obsługi windy

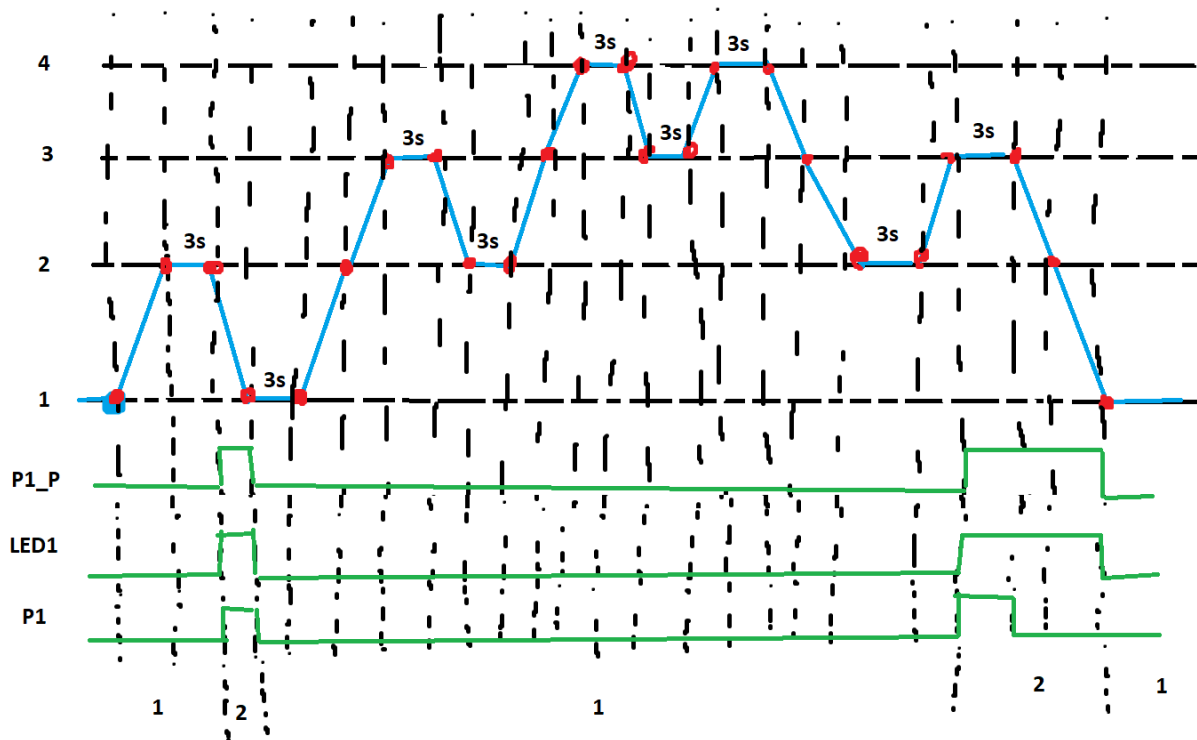
Opisy stanów:

1. Oczekiwanie na 1 piętrze na wciśnięcie jakiegóż guzika
2. Jazda w górę aż do zatrzymania się na 2 piętrze na 3s
3. Jazda w górę aż do 2 piętra bez zatrzymania
4. Oczekiwanie na 2 piętrze na wciśnięcie jakiegóż guzika
5. Jazda w dół aż do zatrzymania się na 1 piętrze na 3s
6. Jazda w górę aż do zatrzymania się na 3 piętrze na 3s
7. Jazda w górę aż do 3 piętra bez zatrzymania
8. Oczekiwanie na 3 piętrze na wciśnięcie jakiegóż guzika
9. Jazda w górę aż do zatrzymania się na 4 piętrze na 3s
10. Jazda w dół aż do zatrzymania się na 2 piętrze na 3s
11. Jazda w dół aż do 2 piętra bez zatrzymania
12. Oczekiwanie na 4 piętrze na wciśnięcie jakiegóż guzika
13. Jazda w dół aż do zatrzymania się na 3 piętrze na 3s
14. Jazda w dół aż do 3 piętra bez zatrzymania

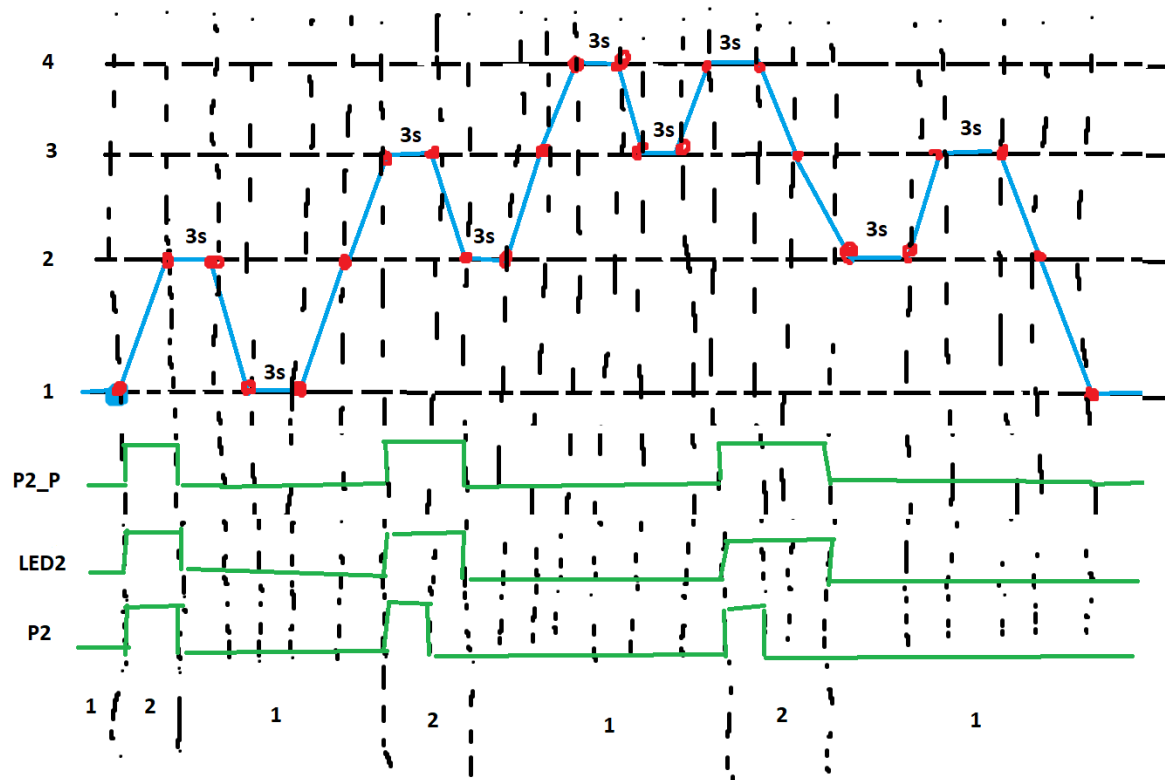
Przebiegi czasowe:



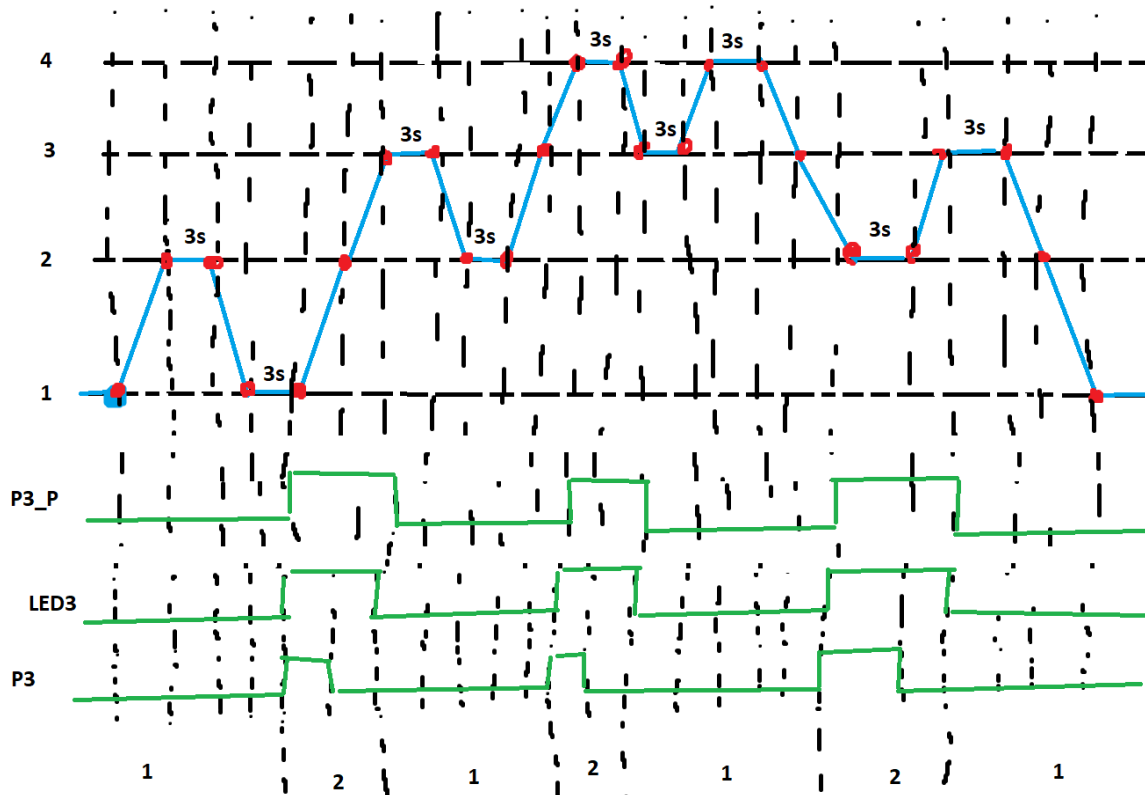
Przebieg czasowy obsługi windy



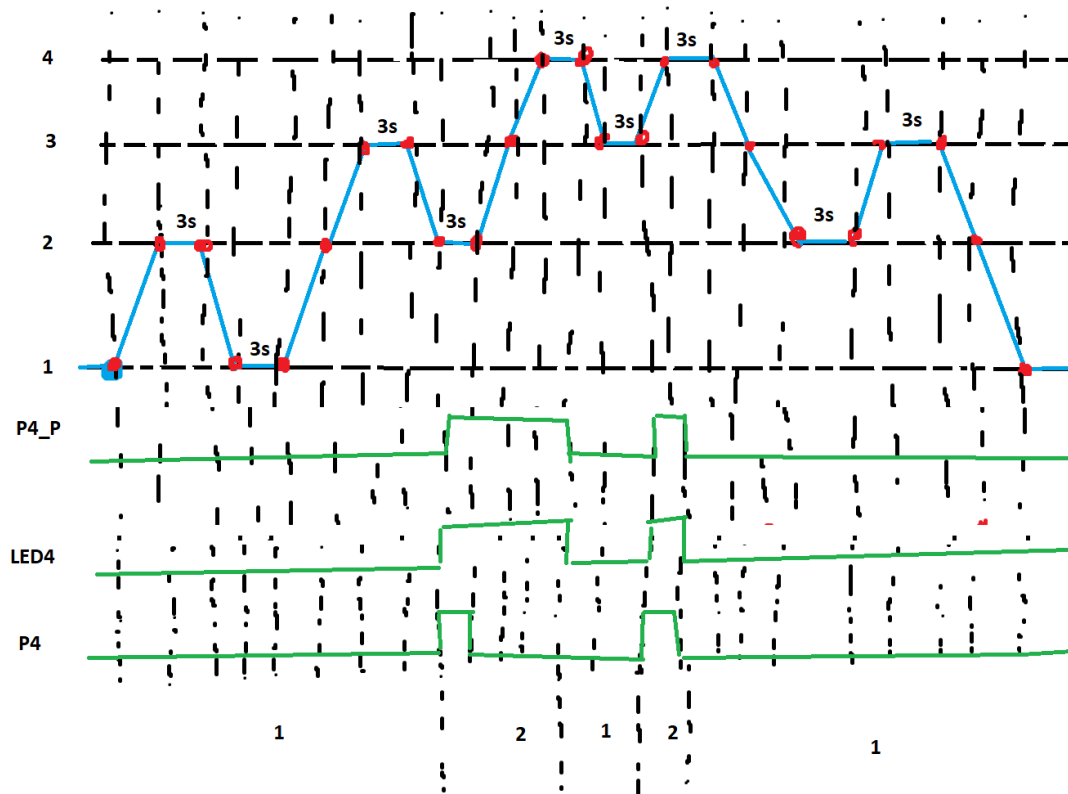
Przebieg czasowy obsługi LED1 i P1



Przebieg czasowy obsługi LED2 i P2



Przebieg czasowy obsługi LED3 i P3



Przebieg czasowy obsługi LED4 i P4

Kod w C:

```
#define i0 A2
#define i1 12
#define i2 11
#define i3 10
#define i4 9
#define i5 8
#define i6 A3
#define i7 A4
#define o0 7
#define o1 6
#define o2 5
#define o3 4
#define o4 3
#define o5 2
#define o6 A0
#define o7 A1
#define k13 13

void setup() {

    pinMode(k13, OUTPUT);
    pinMode(i0, INPUT);
    pinMode(i1, INPUT);
    pinMode(i2, INPUT);
    pinMode(i3, INPUT);
    pinMode(i4, INPUT);
    pinMode(i5, INPUT);
    pinMode(i6, INPUT);
    pinMode(i7, INPUT);
    pinMode(o0, OUTPUT);
    pinMode(o1, OUTPUT);
    pinMode(o2, OUTPUT);
    pinMode(o3, OUTPUT);
    pinMode(o4, OUTPUT);
    pinMode(o5, OUTPUT);
    pinMode(o6, OUTPUT);
    pinMode(o7, OUTPUT);
}

char LED; // LED na płytce ARDUINO
unsigned long stime;

char KZ1,KZ2,KZ3,KZ4;
char P1,P2,P3,P4;
char PGA,PDA;
char LED1,LED2,LED3,LED4;
```

```

////////////////////////////////////
// deklaracje uŹytkownika

char P1_P=0;
char P2_P=0;
char P3_P=0;
char P4_P=0;
char stan=1;
int tim=0;
char ST_LED1=1;
char ST_LED2=1;
char ST_LED3=1;
char ST_LED4=1;

////////////////////////////////////

void loop() {
    stime=millis();

    // odczyt wejŹci
    P1=!digitalRead(i0);
    P2=!digitalRead(i1);
    P3=!digitalRead(i2);
    P4=!digitalRead(i3);
    KZ1=!digitalRead(i4);
    KZ2=!digitalRead(i5);
    KZ4=!digitalRead(i6);
    KZ3=!digitalRead(i7);

    //////////////////////////////////
    // kod uŹytkownika
    // UWAGA: pŹtla pracuje z cyklem 10ms (dla 1s wartoŹci timera powinna
    // wynosiŹ 100)

    switch (ST_LED1) {
        case 1: P1_P = 0; LED1 = 0;
            if ( P1 ) ST_LED1 = 2;
            break;
        case 2: P1_P = 1; LED1 = 1;
            if ( KZ1 ) ST_LED1 = 1;
            break;
    }
}

```

```

switch (ST_LED2) {
    case 1: P2_P = 0; LED2 = 0;
        if ( P2 ) ST_LED2=2;
        break;
    case 2: P2_P = 1; LED2 = 1;
        if ( KZ2 ) ST_LED2 = 1;
        break;
}

switch (ST_LED3) {
    case 1: P3_P = 0; LED3 = 0;
        if ( P3 ) ST_LED3 = 2;
        break;
    case 2: P3_P = 1; LED3 = 1;
        if ( KZ3 ) ST_LED3 = 1;
        break;
}

switch (ST_LED4) {
    case 1: P4_P = 0; LED4 = 0;
        if ( P4 ) ST_LED4 = 2;
        break;
    case 2: P4_P = 1; LED4 = 1;
        if ( KZ4 ) ST_LED4 = 1;
        break;
}

switch (stan) {

    case 1: PGA = 0; PDA = 0;
        if ( P2_P && !tim ) stan = 2;
        else if ( !P2_P && ( P3_P || P4_P ) && !tim ) stan = 3;
        break;
    case 2: PGA = 1; PDA = 0;
        if ( KZ2 ) { tim = 300; stan = 4; }
        break;
    case 3: PGA=1; PDA=0;
        if ( KZ2 ) stan = 4;
        break;

    case 4: PGA = 0; PDA = 0;
        if ( P1_P && !tim ) stan = 5;
        else if ( !P1_P && P3_P && !tim ) stan = 6;
        else if ( !P1_P && !P3_P && P4_P && !tim ) stan=7;
        break;
    case 5: PGA = 0; PDA = 1;
        if ( KZ1 ) { tim = 300; stan = 1; }
}

```

```

        break;
    case 6: PGA = 1; PDA = 0;
        if ( KZ3 ) { tim = 300; stan = 8; }
        break;
    case 7: PGA = 1; PDA = 0;
        if ( KZ3 ) stan = 8;
        break;

    case 8: PGA = 0; PDA = 0;
        if ( P4_P && !tim ) stan = 9;
        else if ( !P4_P && P2_P && !tim ) stan = 10;
        else if ( P1_P && !P2_P && !P4_P && !tim ) stan=11;
        break;
    case 9: PGA = 1; PDA = 0;
        if ( KZ4 ) { tim = 300; stan = 12; }
        break;
    case 10: PGA = 0; PDA = 1;
        if ( KZ2 ) { tim = 300; stan = 4; }
        break;
    case 11:
        if ( KZ2 ) stan = 4;
        break;

    case 12: PGA = 0; PDA = 0;
        if ( P3_P && !tim ) stan = 13;
        else if ( (P1_P || P2_P) && !P3_P && !tim ) stan = 14;
        break;
    case 13: PGA = 0; PDA = 1;
        if ( KZ3 ) { tim = 300; stan = 8; }
        break;
    case 14: PGA = 0; PDA = 1;
        if ( KZ3 ) stan = 8;
        break;
    }
    if ( tim ) --tim;
    //////////////////////////////////////

    // aktualizacja wyjść
    digitalWrite(k13,LED);

    digitalWrite(o0,LED1);
    digitalWrite(o1,LED2);
    digitalWrite(o2,LED4);
    digitalWrite(o3,LED3);

    digitalWrite(o4,PDA);
    digitalWrite(o5,PGA);

```

```
//digitalWrite(o6,pp6);  
//digitalWrite(o7,pp7);  
  
// oczekiwanie na koniec cyklu  
// UWAGA: cykl trwa 10ms  
while(millis()-stime<10);  
  
}
```