

Algorytmy realizowane przez system

Piotr Popis

29 maj

1 Algorytm komponentów

1.1 Color Sensor

1.1.1 Słownie

Kolor sensor przetwarza następujące procedury.

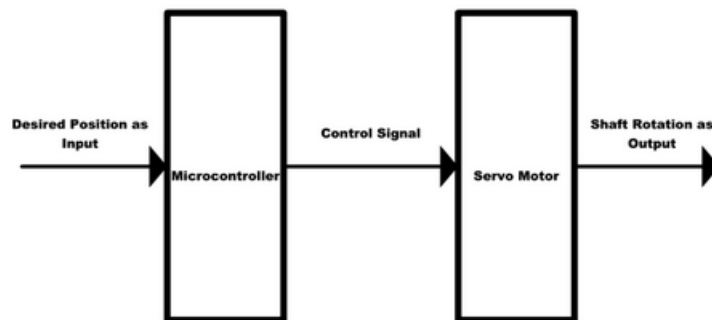
1. Wykrywa kolor obiektu,
2. następnie wysyła sygnał do mikrokontrolera ,aby wykonać akcję.
3. Jest zintegrowany z 4 LED,które frontowo naświetlają obiekt. Wykrywają tak kolor w tablicy 64- fotodiod. Każdy czerwony, niebieski, zielony lub bez filtra.
4. Określony zostaje kolor na podstawie natężenia odbicia światła.
5. Mikrokontroler odbiera sygnał z TCS.

1.2 Motor Servo

1.2.1 Słownie

Jest to DC silnik, który kontroluje pozycję w krokach. Jest kontrolowany przez technikę modulacji pulsowej szerokości. Sygnał PWM ma długość około 20 ms. Szerokość pulsu ma od 1 do 2 ms. Długość pulsu determinuje jak daleko poruszy się silnik.

1.2.2 Graficznie



2 Algorytm w stanie aktywności

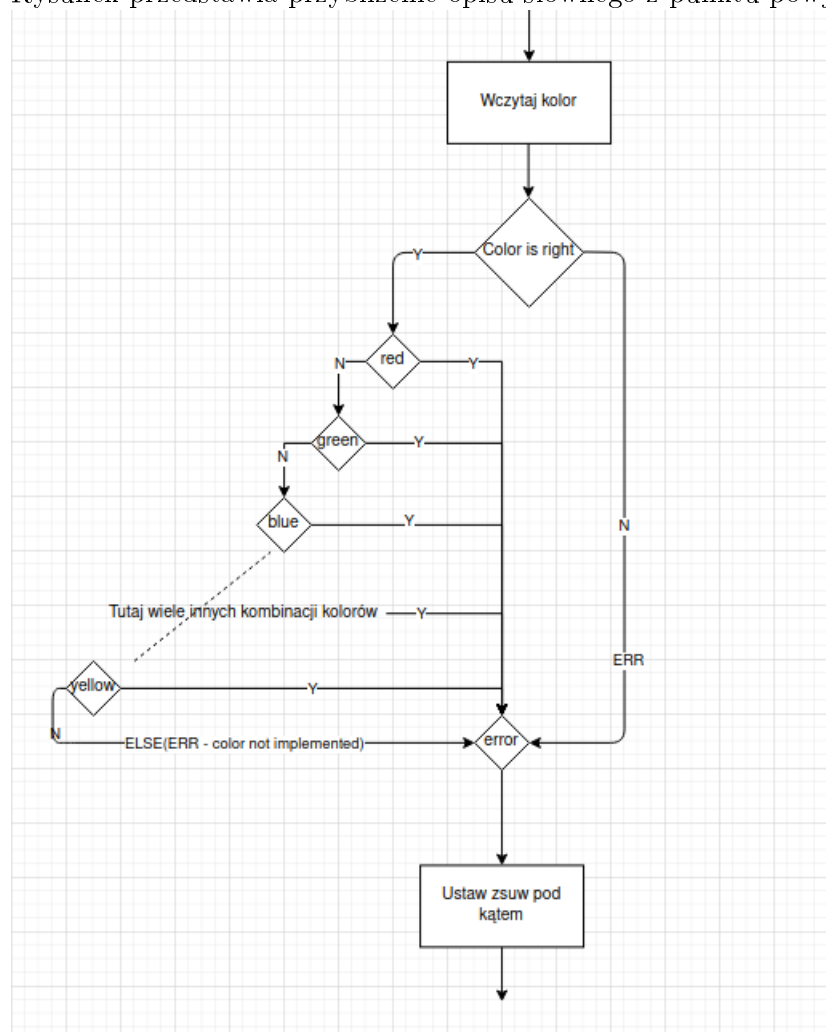
2.1 Słownie

1. Wczytujemy kolor obiektu podanego przez serwo
2. Określamy czy udało się zbadać kolor (wynik jest w zasięgu sensora), jeśli tak to szukamy odpowiedniego koloru, jeśli nie raise error
3. Dprowadzamy iteracyjnie nasze natężenia, czy ich kombinacja pasuje do któregoś koloru. Jeśli Tak przechodzimy do parsowania erroru, jeśli nie to sprawdzamy kolejne kolory aż do ostatniego - jeśli nie został zaimplementowany to "abstrakcyjnie raise error".
4. Teraz jeśli mamy podniesiony sygnał erroru zsuw zostaje ustawiony pod kątem kontenera "error container", jeśli wykryto któryś z kolorów ustawiamy zsuw pod odpowiednim kątem do odpowiadającego zbiornika

Czyli nasza obsługa błędów nie zmienia nic w działaniu po prostu obiekt jest zepchnięty do przeznaczonego do tego kontenera, powala to na uniknięcie "blokowania się obiektu podczas analizy, wewnątrz systemu".

2.2 Diagram

Rysunek przedstawia przybliżenie opisu słownego z punktu powyższego.



3 Algorytm całego systemu w pseudokodzie

3.1 Pseudokod

```
loop () {  
    moveServo1Degrees(X)  
    color = detectColor()  
    decision = chooseContainer(color)
```

```

        moveSerwo2( decision )
        pushObject ()
        moveSerwo1Degrees(-X)
    }

    detectColor () {
        //sprawdzamy czerwone
        digitalWrite(X, LOW);
        digitalWrite(X, LOW);
        R = pulseIn(sensor ,LOW);
        //sprawdzamy zielone
        digitalWrite(X, LOW);
        digitalWrite(X, HIGH);
        G = pulseIn(sensor ,LOW);
        //sprawdzamy niebieskie
        digitalWrite(X, HIGH);
        digitalWrite(X, HIGH);
        B = pulseIn(sensor ,LOW);
        color = colorCombinations(R,G,B);
        return color;
    }

    colorCombinations(R,G,B){
        if(R> 24 and R < 38 and G> 30 and G<44)\{
            color = 1 // yellow
        }
        if(B> 22 and B < 19 and G> 22 and G<25)\{
            color = 2 // orange
        }
        .
        .
        .
        return color;
    }

    chooseContainer( color){
        if( color == 1){
            serwo.write(X);
        }
        if( color == 2){
            serwo.write(X);
        }
        .
        .
        .
    }

```