Projekt zaliczeniowy

Z przedmiotu

Implementacja komputerowych architektur sprzętowych

***Symulacja sortowni w programie Factory IO***



Prace wykonali:

Łukasz Kozak

Arkadiusz Kruszyński

AiR WARiE A3 2020

Spis treści

[Założenia projektowe. 3](#_Toc42430198)

[Podział pracy w grupie. 3](#_Toc42430199)

[Budowa projektu linii sortowni. 3](#_Toc42430200)

[Linia początkowa składa się z: 3](#_Toc42430201)

[Linia paletyzacji: 3](#_Toc42430202)

[Linia sortowania kolorowych przedmiotów: 3](#_Toc42430203)

[Program drabinkowy w TIA Portal. 4](#_Toc42430204)

[Spis tagów użytych w programie. 5](#_Toc42430205)

[Zużycie pamięci sterownika 7](#_Toc42430206)

[Spis podłączeń w Factory IO. 8](#_Toc42430207)

[Weryfikacja działania programu. 8](#_Toc42430208)

[Literatura. 8](#_Toc42430209)

# Założenia projektowe.

Projekt bazuję na ścisłej współpracy programu Factory IO oraz TIA Portal v13 z Symulatorem. W aplikacji TIA Portal stworzyliśmy program drabinkowy, natomiast Factory IO ułożyliśmy imitację sortowni, która ma za zadanie:

* Sortować paczki oraz je paletyzować.
* Sortować kolorowe surowe materiały.
* Zostało również zasymulowana usterka w materiale za pomocą niebieskiej pokrywki.

# Podział pracy w grupie.

Głównym podziałem w grupie był ze względu na linie sortowni, które mogą działać niezależnie i nie występuję problem zależnej pracy od drugiego członka zespołu. Każdy z nas odpowiada za swoją część w programie, symulacji oraz dokumentacji.

Naszą pracę oparliśmy na usłudze GitHub na prywatnym repozytorium.

# Budowa projektu linii sortowni.

W symulacji można wyróżnić trzy gałęzie linii:

* Linia początkowa, startowa z której pojawiają się losowo elementy do posortowania wstępnego.
* Linia paletyzacji.
* Linia sortowania kolorowych produktów.

Taki podział usprawnił współpracę w projekcie, ponieważ każdy miał swoje niezależne zadanie do zrealizowania. Chcieliśmy również stworzyć jak najbardziej realną symulację linii wstępnego sortowania. Dlatego skrzynia elektryczna znajduje się bezpośrednio przy emiterze przedmiotów oraz posiada niezbędne elementy do funkcjonowania sortowni rzeczy.

## Linia początkowa składa się z:

* Emitera przedmiotów, który generuję: kolorowe surowe materiały, niebieskie pokrywki oraz paczki do sortowania.
* Czujnika wizji, którego wyjściem są liczby odpowiadające kolorowi materiału znajdującego się bezpośrednio pod nim oraz czujnika optycznego.
* Sortera dla lekkich przedmiotów……..

## Linia paletyzacji:

* Taśmy transportowej, która naprowadza paczki w odcienie miejsce przed automatem paletyzującym
* Emitera pustych palet oraz dwóch taśm transportowych które doprowadzają oraz odbierają palety do automatu paletyzującego.
* Czujnika obecności paczki na taśmie transportowej, umieszczonego przed automatem paletyzującym.
* Czujnika obecności palety przed oraz za automatem paletyzującym.
* Wyjścia zabierającego paletę z paczkami.
* Automatu do paletyzacji.

## Linia sortowania kolorowych przedmiotów:

* Dwóch taśm transportowych dla lekkich przedmiotów.
* Czujnika wizji, którego wyjściem są liczby odpowiadające kolorowi materiału znajdującego się bezpośrednio pod nim.
* Trzech czujników optycznych i jednego odbiciowego.
* Czterech ramion z taśmą.
* Trzech wyjść które zbierają surowe elementy każdego koloru po posortowaniu.

# Program drabinkowy w TIA Portal.

Do zrealizowania programów użyliśmy oddzielnych bloków funkcyjnych, które są wywoływane w głównym wątku sterownika. Drabinka również znajduję się w załączniku nr 1.

Pomijając elementy konieczne do prawidłowego działania TIA portal z Factory IO, które znajdują się w bloku FC9000 to nasz program składa się:

* Ustawień wstępnych: Start, Stop, Emergency. Każdy z tych statusów są konieczne do działania linii produkcyjnej. W Emergency znajdują się wszystkie błędy jakie mogą się pojawić, między innymi nasz zasymulowany błąd produktu jako pokrywka niebieska.
* Ustawień wstępnych, które odpowiadają za załączenie się taśm i innych niezbędnych rzeczy do działania sortowni.
* Z dużej części kodu sortowania elementów po kolorze surowego materiału.
* Z obsługi symulowanego błędu.

Linia paletyzacji składa się z 8 sieci których zadania zostały podzielone następująco:

1. Odpowiedzialna za dostarczanie oraz odbieranie palet z automaty paletyzującego.
2. Odpowiedzialna za obsługę windy w automacie.
3. Odpowiedzialna za określanie pozycji windy w danym momencie (dla każdej nowej warstwy paczek, winda musiała się delikatnie opuścić).
4. Odpowiedzialna za zliczanie warstw umieszczonych na palecie. Obecnie program jest ustawiony na 3 warstwy, aby nie komplikować programu. Ilość warstw określona jest jednym parametrem w liczniku.
5. Odpowiedzialna za zliczanie ilości paczek umieszczonych na jednej warstwie (z podziałem na kolumny i rzędy, gdyż paczki są umieszczane naprzemiennie). Dodatkowo sieć ta jest odpowiedzialna za sterowanie taśmociągiem oraz określanie kiedy paczki będą w układzie 2 rzędy 3 kolumny lub 3 rzędy 2 kolumny.
6. Odpowiedzialna za sterowanie popychacza, który umieszcza i układa paczki nad paletą.
7. Odpowiedzialna za obsługę szczęk, których zadaniem jest zwarcie paczek, aby były równo umieszczane na palecie. Oraz za otwieranie zapadni, która upuszcza przygotowaną warstwę na stos na palecie.
8. Odpowiedzialna za utrzymywanie niektórych wartości w określonych zakresach.

Podczas wykonywania sortowni kolorowych przedmiotów za pomocą tablicy Karnaugha został zdefiniowany warunek działania taśm transportowych.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| lp. | Czujka na początku taśmy | Czujka na końcu taśmy | Bufor sortowni | Wyjście |
| 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 2 | 0 | 1 | 0 | 1 |
| 3 | 0 | 1 | 1 | 1 |
| 4 | 1 | 0 | 0 | 1 |
| 5 | 1 | 0 | 1 | 0 |
| 6 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 0 |

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 00 | 1 | 1 |
| 01 | 1 | 1 |
| 11 | 1 | 0 |
| 10 | 1 | 0 |

Z tabeli jasno wynika, że warunek przyjmuje postać:

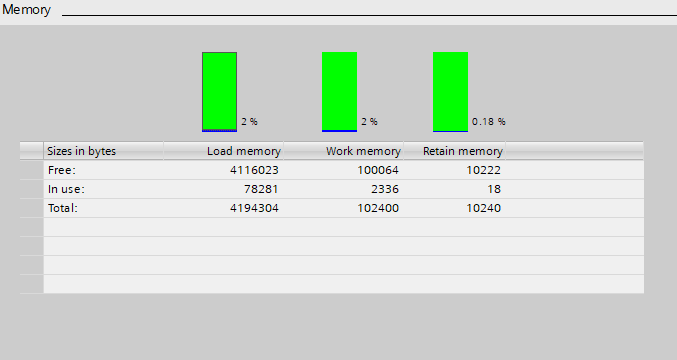
Dlatego zrezygnowaliśmy z czujki na początku taśmy, bo była nadmiarowa.

# Spis tagów użytych w programie.

|  |  |  |
| --- | --- | --- |
| **Name** | **Data Type** | **Logical Address** |
| Tag\_2 | Bool | %I0.0 |
| Tag\_3 | Bool | %I0.1 |
| Start btn | Bool | %I0.2 |
| Emergency | Bool | %I0.4 |
| Stop btn | Bool | %I0.3 |
| Sstart | Bool | %I0.5 |
| VS sorting 1 | DWord | %ID38 |
| VS sorting2 | DWord | %ID34 |
| ColorStopS | Bool | %I0.6 |
| Threshold1 | Bool | %I0.7 |
| ResetBufor | Bool | %I1.0 |
| ErrorSensor | Bool | %I1.1 |
| SolveError | Bool | %I1.2 |
| PallPalletAtEntry | Bool | %I2.0 |
| PallPalletAtExit | Bool | %I2.1 |
| PallPalletLoaded | Bool | %I2.2 |
| PallBoxAtEntry | Bool | %I2.3 |
| PallPusherLimit | Bool | %I2.4 |
| PallPlateLimit | Bool | %I2.5 |
| PallClamped | Bool | %I2.6 |
| PallElevatorMoving | Bool | %I2.7 |
| PallReset | Bool | %I1.3 |
| Tag\_1 | Bool | %Q0.0 |
| Start LED | Bool | %Q0.2 |
| Stop LED | Bool | %Q0.3 |
| ALARM | Bool | %Q0.4 |
| 1st Belt | Bool | %Q2.2 |
| Sorter | Bool | %Q2.3 |
| SorterToRight | Bool | %Q2.1 |
| Arm4th | Bool | %Q0.7 |
| Arm2nd | Bool | %Q1.0 |
| 2nd Belt | Bool | %Q2.4 |
| 3rd Arm | Bool | %Q1.1 |
| PallPalletFeeder | Bool | %Q3.0 |
| PallLoadPallet | Bool | %Q3.1 |
| PallExitConveyor | Bool | %Q3.2 |
| PallMoveToLimit | Bool | %Q3.3 |
| PallElevatorUp | Bool | %Q3.4 |
| PallElevatorDown | Bool | %Q3.5 |
| PallBoxFeeder | Bool | %Q3.6 |
| PallTurn | Bool | %Q3.7 |
| PallPush | Bool | %Q4.0 |
| PallLoadBelt | Bool | %Q4.1 |
| PallClamp | Bool | %Q4.2 |
| PallOpenPlate | Bool | %Q4.3 |
| Priv ST | Bool | %M0.0 |
| START | Bool | %M0.1 |
| Tag\_4 | Bool | %M0.2 |
| Tag\_5 | Bool | %M0.3 |
| Tag\_6 | Bool | %M0.4 |
| Change direction | Bool | %M0.5 |
| Green Mark | Bool | %M1.0 |
| BlueMark | Bool | %M1.1 |
| BuforSort | Bool | %M8.0 |
| FULLSORT | Bool | %M1.2 |
| MetalMark | Bool | %M1.3 |
| ERROR | Bool | %M8.2 |
| ErrorMark | Bool | %M1.4 |
| PallOpenPlateRising | Bool | %M3.0 |
| PallPalletAtExitRising | Bool | %M3.1 |
| PallElevatorPosition | Bool | %M3.2 |
| PallBoxAtEntryRising | Bool | %M3.3 |
| PallPusherLimitRising | Bool | %M3.4 |
| PallPlateLimitRising | Bool | %M3.5 |
| PallElevatorDownRising | Bool | %M3.6 |
| PallBoxesRows | Int | %MW10 |
| PallNumberBoxRows | Int | %MW14 |
| PallAlturningFlag | Bool | %M3.7 |
| PallPushRising | Bool | %M4.0 |
| PallElevatorMovingFalling | Bool | %M4.1 |
| PallPushBoxRising | Bool | %M4.2 |
| PallPusherLimitFalling | Bool | %M4.3 |
| PallElevatorDownRising2 | Bool | %M4.4 |
| SORTERisFULL | Bool | %M8.3 |
| 4th belt | Bool | %Q2.5 |
| Arm 4th | Bool | %Q1.2 |
| 1st pivot | Bool | %Q1.3 |
| 2nd pivot | Bool | %Q1.4 |
| 3rd pivot | Bool | %Q1.5 |
| 4th pivot | Bool | %Q1.6 |
| GreenTakeOFF | Bool | %Q0.5 |
| BlueTakeOFF | Bool | %Q0.6 |
| MetalTakeOFF | Bool | %Q1.7 |
| ErrorTakeOFF | Bool | %Q2.0 |

Tag 1, Tag 2, Tag 3 są używane przez szablon komunikacyjny pomiędzy TIA portal a Factory IO.

# Zużycie pamięci sterownika



# Spis podłączeń w Factory IO.



# Weryfikacja działania programu.

Program wraz z dokumentacja znajduje się na portalu GitHUB:

<https://github.com/xxBeWolfxx/Factory-IO/releases>

Symulacja linii sortowni znajduje się na portalu YouTube:

1. Etap: <https://www.youtube.com/watch?v=4FlKk7kkpLs>
2. Etap: <https://www.youtube.com/watch?v=yXeAZVB6a0Q>

Zostało to w taki sposób, żeby w łatwy sposób zaprezentować działanie paletyzera, ponieważ przy sortowaniu wszystkich kolorowych materiałów proces paletyzacji może trwać bardzo długo.

# Literatura.

1. <https://docs.factoryio.com/> - dokumentacja Factory IO
2. <https://www.youtube.com/user/RealGamesLDA> - kanał YouTube twórców Factory IO