**КУРСОВОЙ ПРОЕКТ**

**на тему:**

Разработка системы управления автоматического склада

по **МДК 01.02 Разработка и моделирование мехатронных систем**

Выполнил студент группы №6031

Гафаров Кирилл Борисович

Специальность 15.02.10

«Мехатроника и мобильная робототехника»

Проверил преподаватель Шемякин В. В.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_

Дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_/

Санкт-Петербург 2024 г.

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 3](#_Toc164159507)

[1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 5](#_Toc164159508)

[1.1 ПРОГРАММЫ И ЯЗЫКИ ПРОГРАММИРОВАНИЯ 5](#_Toc164159509)

[1.1.1 Выбор языка программирования: 5](#_Toc164159510)

[1.1.2 Выбор программ: 10](#_Toc164159511)

[1.2 КОНТРОЛЛЕРЫ И ОБОРУДОВАНИЕ 12](#_Toc164159512)

[1.2.1 Выбор ПЛК контроллера: 12](#_Toc164159513)

[1.2.2 Выбор оборудования: 14](#_Toc164159514)

[2. ПРАКТИЧЕСКАЯ ЧАСТЬ 21](#_Toc164159515)

[2.1 УСТАНОВКА ПРОГРАММ 21](#_Toc164159516)

[2.2 СВЯЗЬ *TIA PORTAL* С *FACTORY IO* 22](#_Toc164159517)

[2.3 ПРОЕКТИРОВАНИЕ АВТОМАТИЧЕСКОГО СКЛАДА В *FACTORY IO* 22](#_Toc164159518)

[2.4 РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ АВТОМАТИЧЕСКИМ СКЛАДОМ 24](#_Toc164159519)

[2.4.1 Подготовка 24](#_Toc164159520)

[2.4.2 Разработка 25](#_Toc164159521)

# **ВВЕДЕНИЕ**

Актуальность:

В современном мире все больше внимания уделяется большим складам хранения, автоматизация складских процессов и работ сильно упростит множество трудностей, которые терпит компания без автоматизации, среди преимуществ таких складов выделяют:

* Автоматический склад поможет быстрее обрабатывать заказы, ведь на нем каждый день движутся множество ценностей и медлить чревато денежными расходами;
* Исключит потребность в персонале, без автоматизации персонал становится незаменим, а заболевание или увольнение работника может привести к хаосу на производстве;
* Уменьшит процент ошибок как следствие человеческого фактора, ведь роботы в отличие от людей не устают и не допускают глупых ошибок по невнимательности;
* А также ускорит упростит транспортировку груза по складу, ведь человеку не придется раскладывать товар вручную

Цель работы:

разработать систему управления автоматического склада на языке программирования *LAD*, используя программу *TIA Portal*, а также программу *Factory IO*, для визуализации протекающих процессов склада.

Задачи:

1. Выбрать контроллер и оборудование для оптимальной работы автоматического склада
2. Установить программы *TIA Portal* и *Factory IO* на Персональный Компьютер
3. Подключить автоматический склад к *TIA Portal*
4. Спроектировать и разработать автоматический склад в программе *Factory IO*
5. Разработать систему управления и работы данного склада
6. Проверить работу автоматического склада
7. Сделать вывод

# 1. **ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

1.1 ПРОГРАММЫ И ЯЗЫКИ ПРОГРАММИРОВАНИЯ

1.1.1 Выбор языка программирования:

Выбор языка программирования — важный шаг для разработчика, будь то опытный специалист или новичок. В зависимости от языка может определиться сложность программы и ее возможность в принципе.

Существует множество языков программирования, самые популярные из которых это *JavaScript, Python, C++* и т. д. Но они не подходят, потому что автоматический склад будет работать с помощью программируемых логических контроллеров (далее - ПЛК) контроллера, а для программирования ПЛК подходят лишь несколько языков программирования:

1. *LAD (Ladder Diagram)*
2. *FBD (Function Block Diagram)*
3. *SFC (Sequential Function Chart)*
4. *IL (Instruction List)*
5. *ST (Structured Text)*
6. *CEM (****Cause and Effect Matrix)***

Они также разделяются на графические и текстовые. Графические более удобны, т. к. графическое представление облегчает понятие логики, а также обладает рядом преимуществ:

1. Для освоения основ визуального программирования не требуется знание синтаксиса текстового языка.
2. Средства визуального программирования исключают возникновение ошибок, связанных с опечатками в тексте.
3. Визуальное программирование позволяет представлять данные в виде графических элементов.
4. С помощью визуального программирования можно решать серьёзные задачи, например, в конструкторах приложений, программировании потоков данных и т.п.

Поэтому выбор языка падает на *LAD*, *FBD*, *SFC* и *CEM* потому что *SIMATIC S7-1200* поддерживает только эти 4 языка программирования. Однако CEM является новым языком программирования и на его изучение придется потратить немало времени, поэтому уберем его из списка. Рассмотрим каждый из оставшихся и решим, какой использовать будет более удобно и эффективно.

1. *FBD*:

Язык *FBD (Function Block Diagram)* – графический язык программирования, обеспечивающий управление потоками данных всех типов, позволяющий использовать алгоритмы вызовом функций и функциональных блоков (см. Рисунок 1).

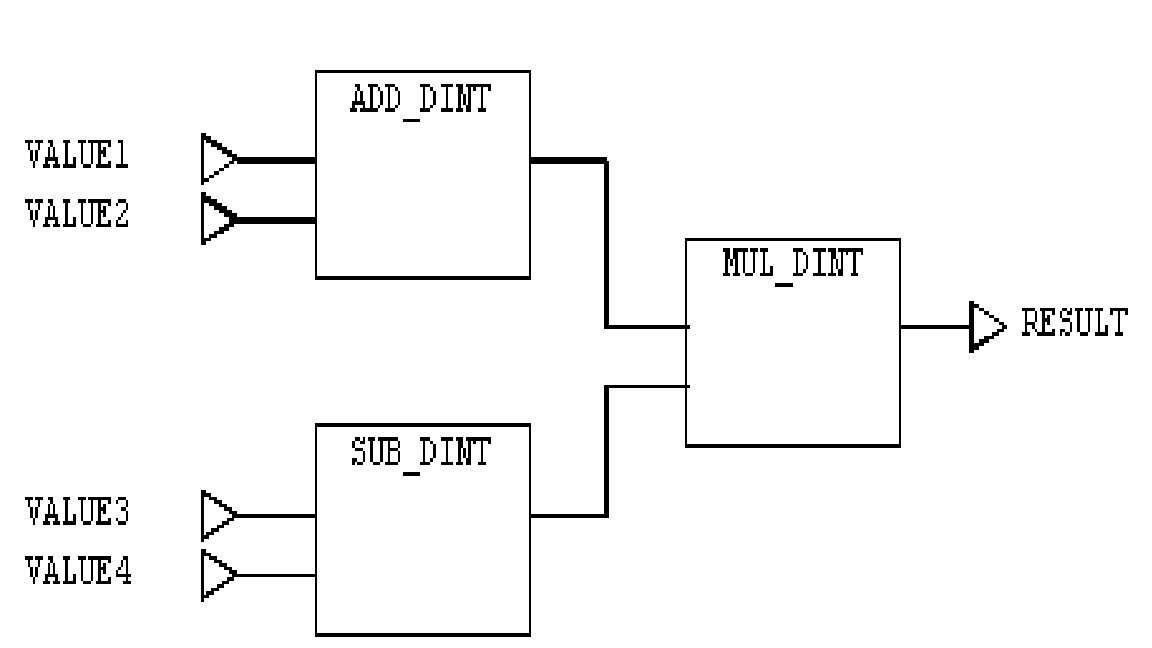


Рисунок 1 - пример языка *FBD*

Достоинства *FBD*:

* Наглядное и интуитивно понятное представление программы в виде блок-схемы.
* Удобен для отображения параллельных процессов и асинхронных сигналов.
* Легко читаем и понятен для новичков.

Недостатки FBD:

* Не позволяет реализовать сложные логические выражения и алгоритмы.
* Требует большого количества блоков и соединений для простых программ.
* Программы могут быть громоздкими и трудно расшифруемыми.

1. *SFC*:

Язык программирования *SFC (Sequential Function Chart)* – это язык программирования, предназначенный для программирования промышленных контроллеров. Его используют для написания программ последовательного управления технологическим процессом. При программировании контроллеров семейства *SIMATIC* от компании Siemens используют две версии этого языка: *Graph 7* и *STEP 7*.

*SFC* –это мощный графический язык, но не способный заменить другие, т. к. он относится к отсталым языкам программирования ПЛК и будет более полезным использовать *LAD* или *FBD*.

1. *LAD (LD)*:

Язык логических диаграмм (*LD*) - это графический язык программирования, который используется для программирования логических контроллеров. *LD* использует схемы, состоящие из контактов и катушек, которые соответствуют сигналам и действиям, выполняемым ПЛК. Язык *LD* содержит графические элементы, которые образуют схемы, содержащие графические элементы, такие как контакты, катушки, блоки и связки. Катушка обозначает не, например, катушку реле, а логический выход, который обычно называют таковым в документации.

Различаются нормально замкнутые и нормально разомкнутые контактные элементы, которые можно сопоставить с [нормально замкнутыми](https://ru.wikipedia.org/wiki/%D0%9D%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE_%D0%B7%D0%B0%D0%BC%D0%BA%D0%BD%D1%83%D1%82%D1%8B%D0%B5_%D0%BA%D0%BE%D0%BD%D1%82%D0%B0%D0%BA%D1%82%D1%8B) и [нормально разомкнутыми](https://ru.wikipedia.org/wiki/%D0%9D%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE_%D1%80%D0%B0%D0%B7%D0%BE%D0%BC%D0%BA%D0%BD%D1%83%D1%82%D1%8B%D0%B5_%D0%BA%D0%BE%D0%BD%D1%82%D0%B0%D0%BA%D1%82%D1%8B) кнопками в электрических цепях:

* I0.0, I0.1, I0.4 – Нормально разомкнутые контакты, которые пропускают сигнал, при истинном значении и перекрывают при обратном.
* I0.5 – Нормально замкнутый контакт, который напротив закрывает сигнал при истинном значении и открывает при ложном.
* Q0.0 – Итог логической цепочки (катушка). (см. Рисунок 2)

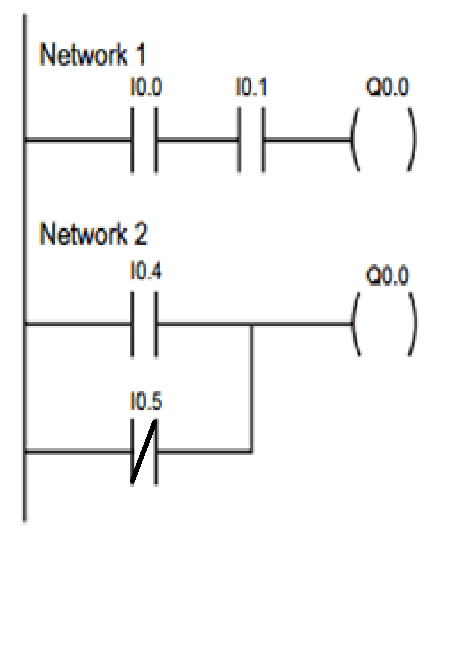


Рисунок 2 – Пример языка *LAD*

Достоинства *LAD*:

1. Простота восприятия. Графическое представление делает операции более понятными и легкими для восприятия. Это позволяет даже новичкам быстро освоить его и создавать программы для ПЛК.

2. Наглядность. Графическое представление наглядно представляет последовательность операций и условий. Это упрощает отладку и позволяет быстро обнаружить ошибки в программе.

3. Простота модификации. В языке *LD* легко добавлять, изменять или удалять операции и условия, что позволяет быстро вносить изменения.

4. Широкое применение. Этот язык широко используется в программировании ПЛК, поэтому знакомые с этим языком, могут легко работать с различными типами ПЛК и применять свои навыки в разных отраслях.

Недостатки:

1. Ограниченные возможности. Язык *LAD* может быть ограничен в возможностях реализации сложных алгоритмов и математических операций, так как больше предназначен для логических операций.

2. Сложность отладки. При наличии большого количества операций и условий может быть сложно найти и исправить ошибки.

3. Ограниченная переносимость. Программа не на все типы ПЛК может перенести программу без изменений.

Язык *LD* является одним из самых распространенных языков программирования для программных логических контроллеров. Он используется для создания схем управления различными процессами в промышленности, самый простой и понятный в использовании, имеет высокую производительность, и поэтому выбор языка для системы управления автоматическим складом пал на него.

### 1.1.2 Выбор программ:

Для программирования одного языка недостаточно, чтобы использовать его требуется программа, в которую будет писаться код. Самые популярные языки, которые подходят для программирования ПЛК на графическом языке это *Tia* *Portal* и *CoDeSys*. Рассмотри оба и решим какой будет использовать.

* 1. *CoDeSys*:

*CoDeSys (Controller Development System)* —инструментальный программный комплекс промышленной автоматизации, основой которого является разработка программ для [ПЛК](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D1%83%D0%B5%D0%BC%D1%8B%D0%B9_%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%BB%D0%B5%D1%80). Для программирования контроллера в него должна быть встроена система исполнения *(Controller Runtime System)*, которую устанавливают в него в процессе изготовления. Также существует специальный инструмент *(*[*Software development kit*](https://ru.wikipedia.org/wiki/Software_development_kit)*)*, позволяющий адаптировать её к различным аппаратным и программным платформам.

Изначально *CoDeSys* был нацелен на задачи, требующие автономности, надежности и предельного быстродействия при минимуме аппаратных средств. Благодаря этому он вышел далеко за рамки традиционных для МЭК 61131-3 (раздел международного стандарта, описывающий языки программирования для ПЛК) систем ПЛК. Сегодня автомобили, краны, экскаваторы, самосвалы, яхты, печатные машины, деревообрабатывающие станки, литейные и прокатные машины, сборочные автоматы крупнейших мировых брендов включают один или группу встроенных контроллеров с *CoDeSys*.

* 1. *Tia Portal*:

*TIA Portal (Totally Integrated Automation Portal)* — [интегрированная среда разработки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) [систем автоматизации технологических процессов](https://ru.wikipedia.org/wiki/%D0%90%D0%A1%D0%A3_%D0%A2%D0%9F) от уровня приводов и контроллеров до уровня человеко-машинного интерфейса. Является воплощением концепции комплексной автоматизации *(Totally Integrated Automation)* и эволюционным развитием семейства систем автоматизации [*Simatic*](https://ru.wikipedia.org/wiki/Simatic) компании [*Siemens AG*](https://ru.wikipedia.org/wiki/Siemens_AG).

В *TIA PORTAL* можно разрабатывать программы для логических контроллеров на языках *LAD/FBD/STL*.

Состав программного обеспечения, интегрируемого в среду *TIA Portal*, выбирается произвольно и может быть адаптирован к кругу решаемых задач. Версии программных продуктов должны соответствовать используемой версии *TIA Portal*.

В настоящее время *TIA Portal* позволяет использовать в своем составе:  
• *SIMATIC STEP 7 Professional/Basic* для конфигурирования, программирования, выполнения пуско-наладочных работ, диагностики и обслуживания систем автоматизации на базе программируемых контроллеров *SIMATIC S7-1200/ S7-1500/ S7-300/ S7-400/ WinAC*, а также периферийных контроллеров на базе аппаратуры станций *SIMATIC ET 200*.  
• *SIMATIC WinCC Basic/ Comfort/ Advanced/ Professional* для решения всего спектра задач построения систем человеко-машинного интерфейса: от конфигурирования простейших панелей операторов до построения многоместных компьютерных систем визуализации, оперативного управления и мониторинга.  
• *SINAMICS StartDrive* для конфигурирования, программирования, выполнения пуско-наладочных работ, диагностики и обслуживания приводных систем на базе регулируемых приводов серий *SINAMICS G110M/*  
*G120/ G120C/ G120D/ G120P*.  
• *SIMOTION Scout TIA* для конфигурирования, программирования, выполнения пуско-наладочных работ, диагностики и обслуживания систем управления перемещением на базе компонентов серии *SIMOTION*.  
• *SIRIUS SIMOCODE ES* для конфигурирования аппаратуры управления и защиты двигателей серии *SIRIUS*.  
• *SIRIUS Soft Starter ES* для конфигурирования устройств плавного пуска серии *SIRIUS 3RW44.*

Для программирования системы управления автоматическим складом стоит выбрать *Tia Portal*, а именно *SIMATIC STEP 7 Professional/Basic*. Потому что он, при необходимости может дополняться опциональным программным обеспечением, расширяющим функциональные возможности всех, нескольких или отдельных пакетов программ *TIA Portal*.

## 1.2 КОНТРОЛЛЕРЫ И ОБОРУДОВАНИЕ

### 1.2.1 Выбор ПЛК контроллера:

ПЛК – универсальные микропроцессорные устройства, предназначенные для управления работой различных машин и технологических процессов. Программируемые логические контроллеры являются важным компонентом любой современной автоматической системы управления. Компания *Siemens* давно прочно заняла первое место по количеству произведенных и применяемых контроллеров в мире. Это самый популярный бренд в этой сфере. Контроллеры *Siemens*, которые постоянно разрабатываются и совершенствуются, задают тенденции в области промышленной автоматизации.

Поэтому в выборе ПЛК для системы управления автоматическим складом остановимся на контроллерах Siemens таких как *Simatic S7-300, Simatic S7-400*, *Simatic S7-1200*, *Simatic S7-1500*. Рассмотрим каждый из них.

1. Промышленная система управления *Simatic S7-300* — это самая продаваемая система управления из широкого ассортимента компании *Siemens AG*. Он предназначен для реализации различных задач автоматизации среднего уровня.

Это один из самых популярных программируемых контроллеров, используемых для построения как автоматических, так и распределенных систем управления на основе сетей связи.

*Simatic S7-300* предоставляет универсальную платформу автоматизации для системных решений с упором на производственные технологии.

2. Промышленная система управления *Simatic S7-400* - разработана для решения самых сложных задач автоматизации в больших масштабах.

Его сфера деятельности — это в основном крупные производственные единицы, системы для сбора, архивирования и обработки технологических данных, которые типичны для энергетики, фармацевтики, химии, пищевой промышленности и т. д.

3. Промышленная система управления *Simatic S7-1200* - может быть адаптирована практически для любого промышленного применения благодаря широкому диапазону модульных расширений. Интегрированный интерфейс *PROFINET* обеспечивает идеальное взаимодействие других элементов технологии автоматизации в единой инженерной среде *TIA Portal*. Модульная конструкция системы управления облегчает ее расширение, причем без изменения ее физических размеров.

4. Промышленная система управления *Simatic S7-1500* - отличается высочайшей производительностью. Она разработана для управления машинами среднего размера для требовательных приложений, требующих, помимо максимальной производительности, надежной связи, гибкости и технологических функций.

Из исследованного складывается вывод, что лучшим контроллером для системы автоматического склада будет *S7-1500*, но также нам стоит учитывать не только характеристики ПЛК, но и его оптимальность. Цена ПЛК контроллера 1500 находится в диапазоне от 150тыс. до 1,5 млн. рублей, поэтому будет не разумно использовать его. Так, контроллер 1200, который также неплох по характеристикам, как и 1500, но его цена разница между 30тыс.-200тыс. рублей, поэтому разработка систему управления будет осуществляться на *S7-1200*.

### 1.2.2 Выбор оборудования:

Автоматический склад — это склад, который состоит из стеллажных конструкций, автоматических кранов-штабелеров и систем загрузки-выгрузки.

Автоматический склад предназначен для хранения материалов в паллетах, ящиках, контейнерах или коробках в складах с вертикальной организацией.

Система состоит из автоматических кранов штабелёров, перемещающих единицы хранения между ячейками стеллажей и зонами погрузки и выгрузки, транспортных систем, и непосредственно стеллажной конструкции и обеспечивает высокий уровень автоматизации складской логистики, и поэтому находит широкое применение в разных областях.

1) Нам понадобится конвейер для транспортировки груза по складу. Самым оптимальным и эффективным можно считать приводные роликовые конвейеры, потому что в отличие от ленточных способны выдержать больший груз и меньше подвержены повреждениям. На складе больше предстоит перемещать тяжелый груз и именно поэтому стоит брать его. Роликовый конвейер — [конвейер](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B2%D0%B5%D0%B9%D0%B5%D1%80), роликами которого, закреплёнными на небольшом расстоянии друг от друга, перемещаются грузы. (см. Рисунок 3) Подобные транспортеры активно применяют в разных сферах промышленности. Используются на участках, на которых затруднено применение ленточных транспортеров из-за риска повреждения ленты грузом или при больших нагрузках.

Предназначена подобная конвейерная система для перемещения:

* бревен;
* коробок;
* паллет;
* ящиков;
* досок;
* стоп фанеры и картона.

Исходя из особенностей транспортируемого груза, в качестве рабочей поверхности могут устанавливаться ролики пластиковые или металлические. В движение они приводятся при помощи цепи через вал от мотор-редуктора. В конструкции приводного роликового конвейера при необходимости могут использоваться специальные фрикционные ролики. Они останавливаются, если нагрузка увеличивается, что способствует снижению трения, уменьшается вероятность износа.

Таблица 1

Технические характеристики роликовых конвейеров:

|  |  |  |  |
| --- | --- | --- | --- |
| Обозначение | Характеристика | Единица измерения |  |
|  | Нагрузка | кг/м | До 2000 |
|  | Скорость | м/мин | До 30 |
| L | Длина транспортирования | мм | 10000 |
| B | Рабочая ширина | мм | 300 — 1500 |
| H | Высота (минимальная) | мм | 160 |
| P | Шаг ролика | мм | Кратное 55 |
| DR | Диаметр ролика | мм | 48, 60, 89 |

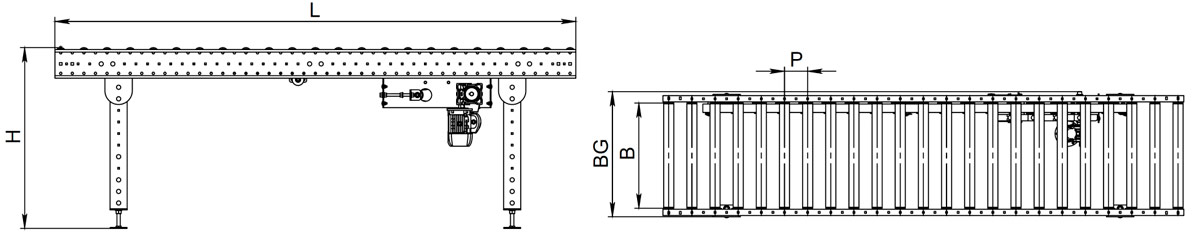


Рисунок 3 – Схема роликового конвейера

Цена такого прямого приводного рольганга начинает от 45 000 за 1 погонный метр.

2) Автоматический кран штабелёр — это кран, применяемый на складах, для перемещения единиц хранения внутри зоны складирования без участия человека.

Кран-штабелер – электроприводная конструкция с перемещающейся тележкой. На ней прикреплена поворотная колонна, по которой передвигается захват. Максимальная грузоподъемность составляет 10 т. Кран-штабелер мостовой может двигаться по рельсам, установленным на конструктивных элементах склада (стенах, перекрытиях) или на стеллажах. Это позволяет использовать их практически в любых помещениях. Кран-штабелер может обслуживать большое количество стеллажных конструкций, установленных в несколько рядов. Он подходит для применения на крупномасштабных складских комплексах.

С помощью кранов-штабелеров обустраиваются механизированные склады промышленных предприятий, торговых баз, логистических комплексов и других объектов. Их использование имеет следующие преимущества:

* значительное повышение производительности;
* высокая безопасность;
* возможность использования в проходах минимальной ширины, что позволяет разместить в помещении максимальное количество стеллажей;
* напольные устройства могут устанавливаться на полах с неровностями;
* управление осуществляется из операторской кабины или снаружи;
* доступная стоимость;
* отсутствие необходимости регистрировать штабелер в надзорных органах.

Таблица 2

Технические характеристики кранов-штабелеров:

|  |  |  |
| --- | --- | --- |
| Характеристика |  | Единицы измерения |
| Грузоподъемность | 0,25 — 10 | т |
| Пролет моста | 2,5 — 22,5 | м |
| Высота подъема груза | 5,1 | м |
| Скорость передвижения моста | 1,12/0,33 | м/с |
| Скорость передвижения тележки | 0,06/0,2 | м/с |
| Скорость подъема груза | 0,13 | м/с |
| Скорость поворота колонны | 1,25/4 | об/мин |

3) Датчики:

Датчик – это устройство, воспринимающее внешние воздействия и реагирующее на них. изменением электрических сигналов. Под внешним воздействием понимается количественная характеристика объекта (его свойство или качество), которую необходимо воспринять и преобразовать в электрический.

Для работы понадобятся оптические датчики. Оптические датчики — это устройства, призванные реагировать на изменения в окружающей среде — появление в зоне видимости полупрозрачных и непрозрачных предметов, дыма, пара, аэрозолей. Они подают сигнал на вход регистрирующей или управляющей системы в момент, когда происходит воздействие электромагнитного излучения в видимом, инфракрасном и ультрафиолетовом диапазонах.

Оптические бесконтактные выключатели имеют в своей конструкции следующие элементы:

* излучатель. В него входят корпус, сам излучатель, подстрочный элемент, генератор и индикатор движения;
* приемник. Состоит из основной части, фотодиода, подстрочного элемента, электронного ключа, триггера, демодулятора и индикатора.

Индикатор рабочего состояния позволяет регулировать чувствительность оптического устройства. С его помощью можно настроить точность восприятия объектов, находящихся на неблагоприятном фоне. В качестве источников излучения выступают светодиоды.

Датчики подразделяются на 3 типа в зависимости от принципа действия:

1. *T* — барьерные. В таких оптических бесконтактных устройствах излучатель и приемник расположены в отдельных корпусах, установленных на одной оси друг против друга. Они могут быть размещены на расстоянии до 100 м. Луч принимается от отдельного передатчика сигнала. Когда объект попадает в зону действия устройства, он прерывает луч. Приемник регистрирует изменения и подает сигнал на управляемое оборудование.

2. *R* — рефлекторные. Излучатель и приемник находятся в одном корпусе. Перед поступлением на датчик луч отражается от рефлектора ― катафота. Дальность работы устройства может достигать 8 м. Дополнительно в подобное оборудование устанавливаются поляризационные фильтры, позволяющие выявлять предметы с зеркальной или отражающей поверхностью. Оптические бесконтактные датчики этого типа широко применяются на конвейерах для подсчета количества изделий.

3. *D* — диффузионные. В таких бесконтактных устройствах приемник и передатчик сигнала устанавливаются в общий корпус. Луч отражается от объекта в рассеянном виде. Приемник учитывает его интенсивность и, если она достаточна, передает сигнал к действию на подключенное устройство. Для более точного определения плотности объекта и обеспечения безошибочного срабатывания используется подавление фона. Бесконтактный оптический датчик со стандартными настройками может действовать на расстоянии до 2 м. Дальность зависит от отражательных свойств предметов и может быть увеличена благодаря использованию поправочного коэффициента.

В автоматическом складе понадобятся датчики, реагирующие на наличие объекта около него, и передающие сигнал. Диффузионные однозначно не подходят, потому что рассеивают луч, барьерные и рефлекторные нам подходят, но диапазон видимости барьерных слишком велик и это не к чему, к тому же его цена составляет около 15-30т. Поэтому в выборе стоит остановиться на ретро-рефлекторных датчиках, цена которых намного меньше и составляет около 5т. рублей.

В основе работы этих датчиков лежит принцип ограничения пути луча света, отражённого от объекта, на расстояние между источником света и приёмником.

Принцип работы таких устройств заключается в нахождении объекта, когда он блокирует свет, излучаемый сенсором и отраженный отражателем. (см. Рисунок 4)

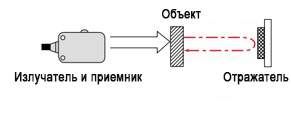


Рисунок 4 – Ретро-рефлекторные датчики

Преимущества:

* Может быть установлен в узком пространстве;
* По сравнению с рефлекторными устройствами, расстояние срабатывания больше;
* Очень простая настройка оптической оси;
* На сканирование непрозрачных объектов не влияет их форма, цвет и материал.

Недостаток:

* Ретро-рефлекторный датчик не сможет обнаружить объект, если его поверхность является зеркальной, однако можно использовать поляризационные сенсоры, которые могут стабильно функционировать с ними.

2. **ПРАКТИЧЕСКАЯ ЧАСТЬ**

2.1 УСТАНОВКА ПРОГРАММ

Скачиваем лицензионную версию *Tia Portal* с официального сайта *siemens.com*, в котором будет находиться проекция нашего ПЛК контроллера, а также предоставлена возможность его программировать. Порядок действий:

1. Переходим на официальный сайт
2. Выбираем раздел “Продукты и услуги” (*Products & Services*)
3. В разделе выбираем пункт “Промышленная автоматизация” ([*Industrial Automation*](https://www.siemens.com/global/en/products/automation.html))
4. Далее находим пункт “Отраслевое программное обеспечение” (*Industry software*)
5. Выбираем пункт “Программное обеспечение для автоматизации” ([*Automation Software*](https://www.siemens.com/global/en/products/automation/industry-software/automation-software.html))
6. Выбираем *Tia Portal*
7. Листаем вниз и берем “Программирование ПЛК с помощью *SIMATIC STEP 7*” (*PLC programming with SIMATIC STEP 7*)
8. В всплывшем окне выбираем тоже самое
9. Листаем вниз, находим *SIMATIC STEP 7 Professional* и *Basic*, в которых можно использовать пробную лицензию на 21 день. Лицензионное ПО стоит около 80 тыс. рублей.

Далее нам понадобиться *Factory IO*, которое также скачивается с официального сайта *factoryio.com*:

1. Переходим на официальный сайт
2. Выбираем пункт “Попробуйте бесплатно”
3. Вводим свои данные
4. После сразу начнется загрузка *Factory IO* с временной лицензией 30 дней, после истечения которых в нем нельзя будет программировать

Из программ нам больше ничего не понадобиться.

2.2 СВЯЗЬ *TIA PORTAL* С *FACTORY IO*

Переходим на сайт *docs.factoryio.com*. В нем есть четкая инструкция по созданию связи между *Tia Portal* и *Factory IO*. Он понадобиться, чтобы скачать специальный шаблон контроллера, для создания проекта в *Tia Portal*. Среди всех перечней версий *Tia* и контроллеров в нем выбираем нужную на *v18* и *s7-1200* соответственно, а затем скачиваем его.

Далее действуем по инструкции ниже:

1. Открываем скаченный шаблон с помощью *Tia Portal* (который, обычно, находится по пути: “C:\Program Files\Siemens\Automation\Portal V18\Bin”)
2. Далее появляется окно, в котором нужно нажать “Upgrade”
3. После открытия выбираем нижнюю кнопку “Open the project view”, ведь нам не нужно создавать контроллер, так как мы используем шаблон, и он уже был встроен
4. Однако, уже встроенный 1211 контроллер может не подойти по своим характеристикам и настоятельно рекомендуется сменить его на максимально приближенный к 1500 – 1214FC DC/DC/RLY. Для этого нажимаем на PLC\_1 правой кнопкой мыши и выбираем “Change device”. В открывшемся окне выбираем нужный
5. Сохраняем его с желаемым именем
6. Далее требуется выбрать устройство PLC\_1 и включить симуляцию, нажав “*Start Simulation*”
7. Если ничего не выбрано, выбираем “*PN/E”* в качестве типа интерфейса, а ниже берем “*PLCSIM S7-1200/1500”*, далее начинаем поиск, по окончанию которого нажимаем на устройство и скачиваем его
8. В *S7-PLCSIM* нажимаем “*RUN”* мод
9. Теперь открываем *Factory IO*, далее новый проект, нажимаем на раздел “*FILE”* в левом верхнем углу, а в открывшемся окне жмем на “*DRIVERS”*
10. Среди перечня выбираем драйвер “*Siemens* *S7-PLCSIM”*
11. Справа сверху переходим в конфигурацию драйвера, где выбираем в качестве модели контроллера “*S7-1200”*
12. После этого можно подключиться к *Tia Portal*, с помощью кнопки “*Connect*”.

2.3 ПРОЕКТИРОВАНИЕ АВТОМАТИЧЕСКОГО СКЛАДА В *FACTORY IO*

Порядок сборки (в скобках указаны имена, которыми нужно назвать программируемые приводы, сенсоры и т. д.):

1) Чтобы выстроить склад нам понадобиться кран-штабелер, выбираем “Stacker Cran” из списка оборудования и ставим его по середине.

2) Ставим к поднимающему лифту два конвейера, ближайший к которому с пробором по середине (Load Conveyor), чтобы поднимающиеся балки могли выдвинуться и поднять груз, и второй длинной 2 м, на котором будут появляться коробки (Entry Conveyor), на нем размещаем “Emitter”.

3) C обратной стороны делаем также, но на конце ставим “Remover”, чтобы коробки исчезали, доезжая до него.

4) По середине Emitter(а) расположим ретро-рефлекторный датчик (At Entry), который будет отвечать за условие появления коробки на конвейере, еще один такой же нужно поставить перед REMOVER(ом) (At Exit), чтобы датчик прочитал груз раньше его исчезновения. Чтобы датчик работал нужно поставить его с одной стороны, а с другой приемник сигнала “reflector”.

5) Также понадобятся такие же датчики с каждой стороны около лифта, но у завозящего конвейера на 1 клетку дальше (At Load), чтобы коробка успела остановиться и на лифт встала четко по середине, а второй датчик максимально на краю (At Unload).

6) Теперь с каждой из сторон крана-штабелера нужно расставить стеллажи “Rack” около крайней балки штабелера белого цвета. В общей сумме с каждой стороны будут по 54 ячейки, а значит общая вместимость склада составляет 108 коробок.

7) Окружим весь склад забором размера L на расстоянии от склада примерно 1-1,5 м для безопасности.

8) Перед забором ставим стоящую балку, на которой расположен электрический распределительный щит. На него расставляем кнопку сброса значения счетчика, окошко счетчика (Вывод ячеек), кнопку старт и стоп, в которых встроены лампы (Старт (лампа) и Стоп (лампа) соответственно), а также регулятор, от которого будет зависеть режим завоза на склад и вывоза со склада.

## 2.4 РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ АВТОМАТИЧЕСКИМ СКЛАДОМ

### 2.4.1 Подготовка

Наконец переходим к программированию системы управления складом. Наша задача состоит в автоматическом управлении склада. Требуется чтобы он мог отвезти груз к крану и погрузить его в ячейку стеллажа в режиме завоза на склад, а в режиме вывоза - вывести груз и отвезти к выходу.

Еще одной причиной в выборе контроллера именно максимально приближенный к 1500, так как у него больше байт на входы и выходы, а значит на нем могут функционировать более мощные системы.

Входами называют датчики, сенсоры, кнопки и т. д., а выходами поршни, конвейеры и приводы. Однако даже 1214 контроллер не обладает нужным количеством входов и выходов, поэтому ему требуется добавить некоторое количество модулей, которые расширяют их. Ему понадобиться 1 модуль DI 16/DQ 16\*24VDC, 3 модуля DQ 16\*24VDC, еще можно добавить 1 сигнальную плату (DQ 4\* 24VDC). DI 16\*24 добавляет 2 байта на вход, DQ 16\*24 столько же на выход, а сигнальная плата DQ 4 \*24 всего 4 бита на выход. После этого нужно им присвоить их байты для входов начиная с 5, а для выходов с 10, также и в настройках драйверов Factory IO. Начинать требуется не с 0, потому что первые байты займет счетчик. Также в Factory IO нужно перетащить сенсоры и приводы в нужные на ваш взгляд ячейки.

Теперь для удобства стоит дать каждому сенсору и приводу имя в Tia Portal. Для этого в разделе контроллера PLC\_1 переходим в папку “PLC tags” и выбираем “Show all tags”, где мы переписываем все из Factory IO. Главное, чтобы у входов “I” и выходов “Q” был тип данных “Bool” – логический тип данных, принимающий значения – True/False, у крана-штабелера QD12 тип “Dint” – целое число двойной длинны, а у окошка с выводом ячеек QD16 тип “Real” – изменчивые переменные.

После того как Tia Portal подключился к Factory IO можно приступать к написанию программы.

### 2.4.2 Разработка

Открываем весь раздел нашего контроллера PLC\_1, переходим в папку “Program blocks”, далее выбираем “Main [OB1]”. Здесь с помощью контактов, катушек и различных функций пишется череда условий из входов и вытекающее следствие в качестве выхода.

Network 1 “Связь с Factory IO”:

Первый network оставляем с функцией, изначально встроенной в нее. Она играет роль создания связи с Factory IO.

Network 2 “Маркеры”:

Создаем второй network, который можно назвать “Маркеры”. Маркерами являются какие-либо пустые контакты, которые не являются приводом или сенсором в работе, они просто могут получить сигнал и являться условием для других приводов.

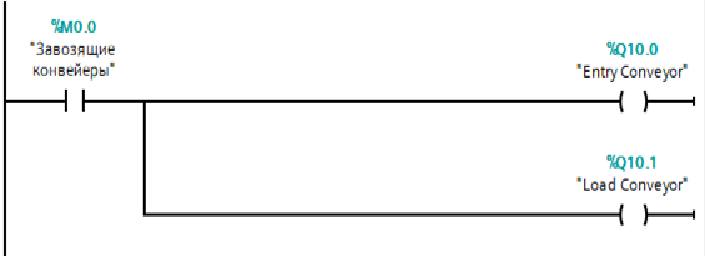
Например, первыми маркерами будут завозящие и вывозящие конвейеры M0.0 и M0.6 соответственно, передав и запомнив сигнал на них, а затем поставив их как условие для передачи сигнала на два разных конвейера их можно запустить. Для M0.0 – Q10.0 (Entry Conveyor) и Q10.1 (Load Conveyor) (см. Рисунок 5.1), а для M0.6 – Q10.5 (Unload Conveyor) и Q10.6 (Exit Conveyor) (см. Рисунок 5.2).

Рисунок 5.1 – Завозящие конвейеры

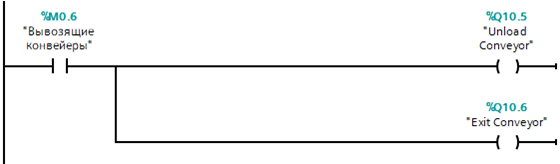


Рисунок 5.2 – Вывозящие конвейеры

В этом же network(е) ставим по условию нажатия кнопки старт I6.1 будет передаваться и запоминаться сигнал на лампе кнопки старт Q10.7 (Set) и выключаться лампа кнопки стоп Q11.0 (Reset). Чтобы элемент на катушке мог запомнить сигнал, то есть даже, когда условия перестанут выполняться, этот элемент все равно обладал значением истинна, нужно поставить не открытую катушку, а с буквой S (Set), а для сбрасывания буква R (Reset). C кнопкой стоп I6.2 все то же самое, но наоборот, будет загораться лампа стоп (Set) и вырубаться лампа старт (Reset). Однако, стоит отметить, что кнопка стоп и аварийный стоп обладают “NC – контактом”, то есть, когда кнопка нажата – сигнал на контакте перекрывается (см. Рисунок 6).



Рисунок 6 – Кнопка старт и стоп

Далее понадобятся 2 полезных маркера, которые можно использовать для определения движения крана-штабелера. Для этого ставим первой строчкой условия Moving X I5.7 и Moving Z I6.0, являющиеся сенсорами движения крана по осям, по которым запоминается маркер M0.1 (Перемещение предмета) (Set). А второй строчкой также Moving X и Z, но не на одной строчке, а из двух, которые вытекают к маркеру M0.2 (Движение по какой-то из осей) (см. Рисунок 7). Если, как в первом случае, условия находятся на одной строчке, это условие “И”, значит сигнал проходит только если все условия выполняются, а если, как во-втором случае, на разных, то это условие “ИЛИ” – сигнал пройдет по любому из всех условий.

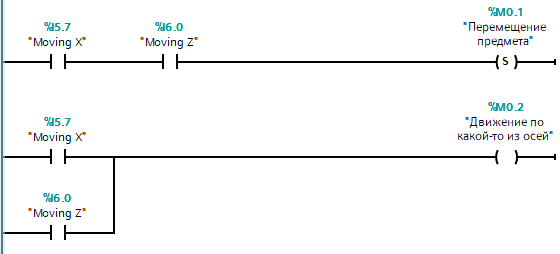


Рисунок 7 – Движение крана

Network 3 “Завоз к складу”:

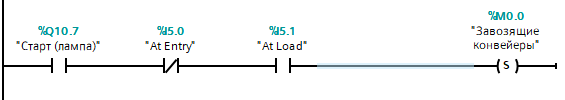
Теперь переходим в следующий network, который называем “Завоз к складу”. Здесь, первой строкой по условиям нажатой кнопки старт, появления коробки на конвейере I5.0 (At Entry) и отсутствии коробки у загрузки I5.1 (At Load) передается и запоминается сигнал на маркер M0.0 (Set). (см. Рисунок 8)

Рисунок 8 – Завозящие конвейеры

Далее по условию наличия коробки у загрузки происходят следующие действия (см. Рисунок 9):

* маркер “Завозящие конвейеры” отключается (Reset)
* условие неравенства, которое находится в библиотеках (“Instructions”). Среди базовых находим “Comparator operations”, выбираем любой “CMP” (их можно менять) и ставим в программу, что “Ячейка” больше 54-ех, далее условие “Перемещение предмета”, после которого таймер задержки “TON” на 1 секунду, чтобы не произошло неполадок от скорости прохождения сигнала. Чтобы поставить таймер нужно в библиотеке найти “Timer operations”, в котором есть нужный “TON”, задерживающий сигнал и “TOF”, который обрубает сигнал по окончанию времени. Чтобы поставить его также перетаскиваем на место в строке и в всплывшем окне нажимаем “Ok”. После таймера ставим условие, что лифт не двигается по осям, то есть закрытый M0.2 “Движение по какой-то из осей”. Для этого нам и нужен был таймер, так как был шанс, что сигнал успеет пройти в моменте смены сигнала движения на остановку. От всех этих условий будет отключаться маркер M0.1 “Перемещение предмета” (Reset) и, если включен режим завоза на склад I6.4 “Завоз на склад” отключается еще и лифт Q10.4 “Lift”
* условие отключенного маркера M0.1 “Перемещение предмета” и горящая лампа кнопки старт Q10.7, что обозначает, что кнопка стоп не была нажата и склад должен продолжать функционировать, от которых идет разветвление на два. В первом случае – это режим завоза на склад и отключенный лифт, при которых балки выдвигаются влево Q10.2 “Forks left” (Set), на которых стоит палетка с коробкой. Они не только держат их, но и забирают и кладут на конвейеры и стеллажи, в данный момент балки забирают палетку с завозящего конвейера. Во-втором случае – режим вывоза со склада I6.5 “Вывоз со склада”, при котором, если маркер M0.4 “Вернулся с грузом” отключен, то есть на нем нет коробки, то просто выключается маркер M0.3 “Возврат лифта” (Reset), что означает, что лифт просто поедет за коробкой, но если “Вернулся с грузом” горит, то происходит еще одно разветвление. Если лифт включен, то балки выдвигаются вправо Q10.3 “Forks right” (Set), а когда они выдвинулись I5.4 “At right” – лифт отключается (Reset). А когда лифт выключен, то при условии, что балки справа ставим задержку “TON” на 1 секунду после которого балки задвигаются обратно Q10.3 “Forks right” (Reset). Таймер здесь нужен для того, чтобы лифт не опускался одновременно с движением балок и коробка встала ровно. Теперь от выключенного лифта ставим открытую катушку маркера M0.5 “Ячейка убрана”, являющейся значением, что ячейка стеллажа освободилась, а также условие I5.3 “At middle”, что означает, что балки не выдвинуты, а расположены по середине лифта, при котором отключается маркер M0.4 “Вернулся с грузом” (Reset).

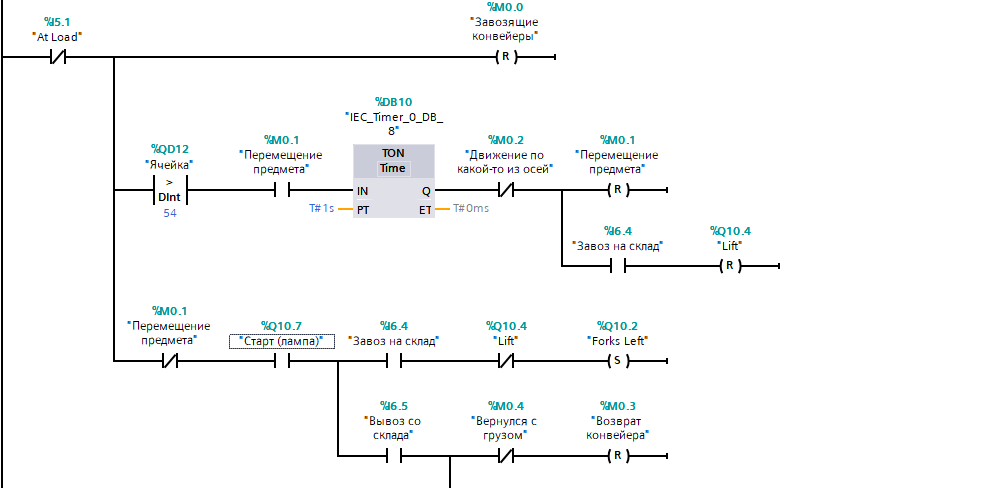
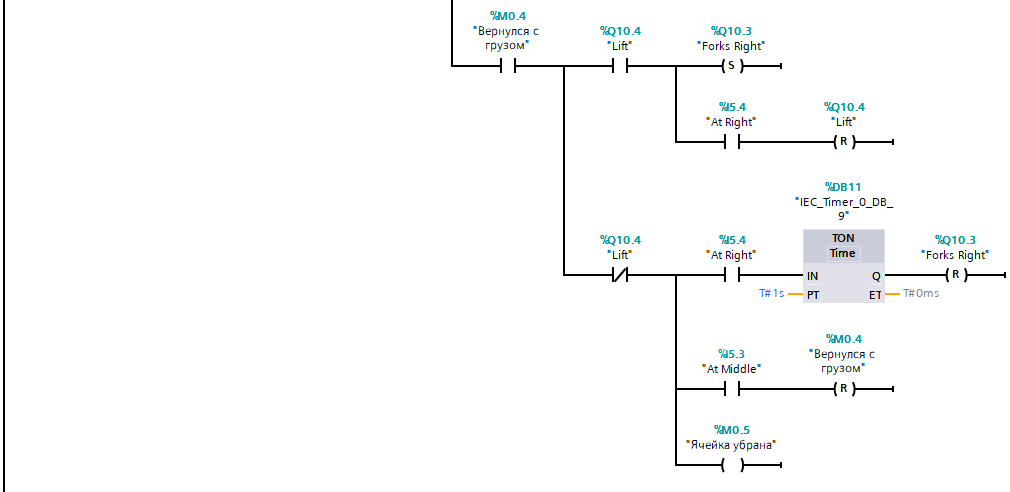


Рисунок 9 – Завоз к складу

Network 4 “Распределение по складу”:

Данная часть программы будет отвечать за распределение коробок на палетках по стеллажу. В режиме завоза на склад, лифт будет брать груз и отвозить на ближайшую свободную ячейку, а в режиме вывоза забирать с ближайшей и убирать на конвейер. Для начала загрузим ящик на лифт:

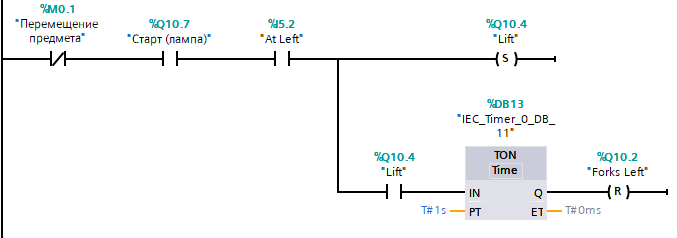
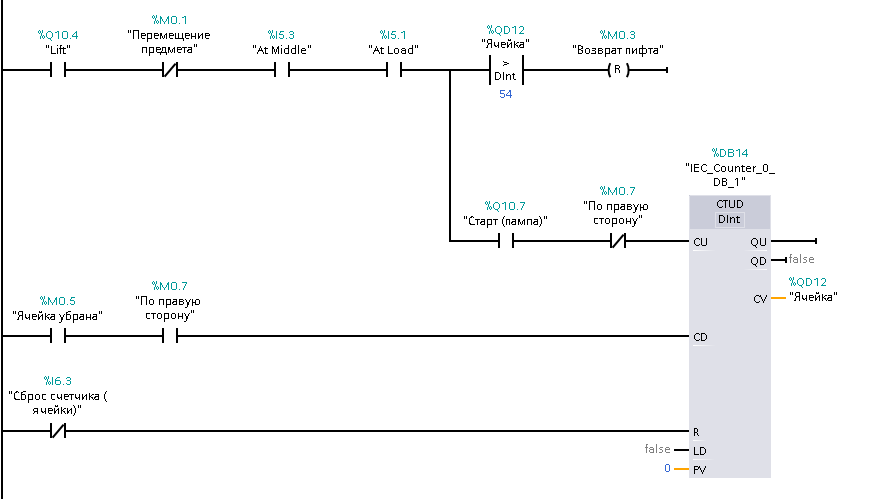
Если маркер M0.1 “Перемещение предмета”, горит лампа кнопки старт и балки выдвинуты влево I5.2 “At left”, чтобы забрать палетку, то поднимется лифт (Set), а когда лифт поднимется с задержкой в 1 секунду балки задвинутся обратно Q10.2 “Forks left” (Reset) (см. Рисунок 10).

Рисунок 10 – Загрузка на лифт

Теперь, главная часть, которая основа на счетчике, определяющим позицию лифта, то есть номер ячейки на стеллаже. Например, если придать QD12 “Ячейка” значение 1, то лифт переместится к первой ячейке и так далее. Всего 1 такой кран сможет обслуживать 54 ячейки, но так как в данной работе стеллажи поставлены с обеих сторон, то вместимость увеличивается в 2 раза, то есть до 108.

Сначала, при условиях, что лифт включен, маркер перемещения предмета выключен, балки не выдвинуты и находятся по середине I5.3 “At middle”, а коробка находится у загрузки I5.1 “At Load”, то сигнал разветвляется, в первом случае еще одно условие неравенства, означающее, что если номер ячейки QD12 больше 54, то есть лифт должен вернуться в начальное положение, то отключается маркер возврата лифта M0.3 (Reset), а во-втором случае, при условии, что лампа старт горит и маркер M0.7 “По правую сторону” выключен, то сигнал проходит на CU счетчика, отвечающего за QD12. Чтобы поставить счетчик, также как и с таймером переходим в библиотеки, в котором есть “Counter operations”. В нем существуют счетчики CTU, добавляющий +1, и CTD, убавляющий -1, но есть и CTUD, который комбинирует обе функции, выбираем его.

CU дает счетчику +1, а маркер M0.7 “По правую сторону” отвечает за определение, был ли лифт у данной ячейки раньше. Чуть ниже к CD счетчика подходят 2 условия: M0.5 “Ячейка убрана” и M0.7 “По правую сторону”. CDотвечает за -1 к счетчику. Еще ниже условие нажатия кнопки I6.3 “Сброс счетчика (ячейки)”, которое можно протянуть либо к R (сброс счетчика), либо к LD (сброс счетчика до начального значения), что особого значения в работе не имеет, также у счетчика обязательно нужно выдать PV, то есть начальное значение, с которого будет вести счет, у нас это 0. QU пропускает сигнал, при условии, что значение на CV>=PV, а QD, наоборот, если CV<=PV, но нам они не нужны. А CV это параметр, который будет принимать значение счетчика, здесь это QD12 “Ячейка” (см. Рисунок 11).

Рисунок 11 – Распределение по складу

Network 5 Возврат лифта и действия на стеллаже:

С помощью счетчика можно контролировать езду и положение лифта, однако, если мы сбросим его до 0, то лифт не вернется в начальное положение, при этом положение 1 также не является его начальным, поэтому было принято решение выводить его за пределы, ведь если придать лифту крана значение больше 54, то есть количества, которое он может обслуживать, то он вернется в самое начало к загрузке. В этом поможет функция “ADD”, которая присваивает к уже существующему значение то, которое мы выберем сами, а именно 55, на 1 единицу больше, чем количество стеллажей на одной стороне. Ранее уже был упомянут маркер M0.3 “Возврат лифта”, именно по нему сигнал будет проходить к этой функции, но когда сигнал с нее уходит, то есть маркер отключается, то к значению вернется первоначальное число (см. Рисунок 12)

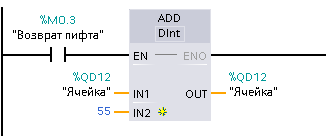


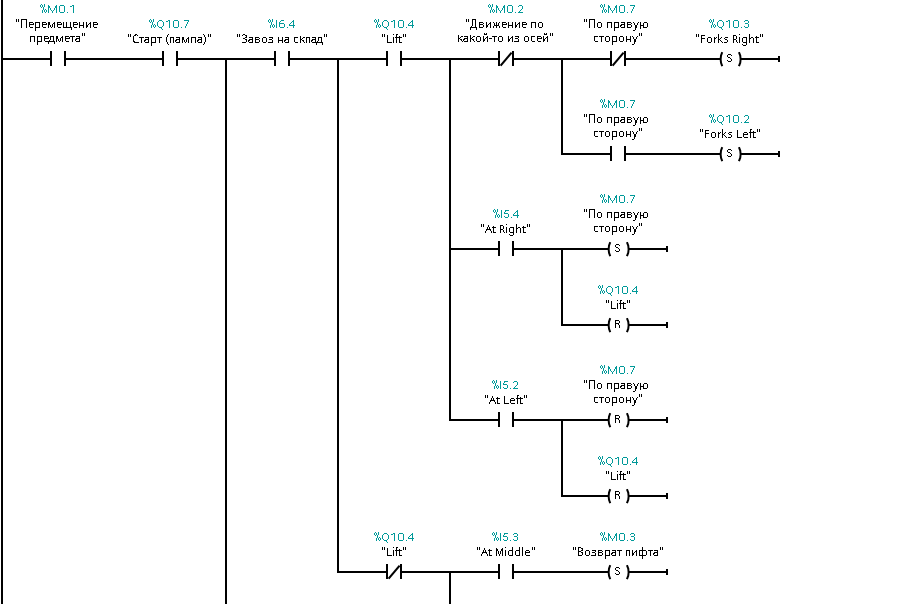
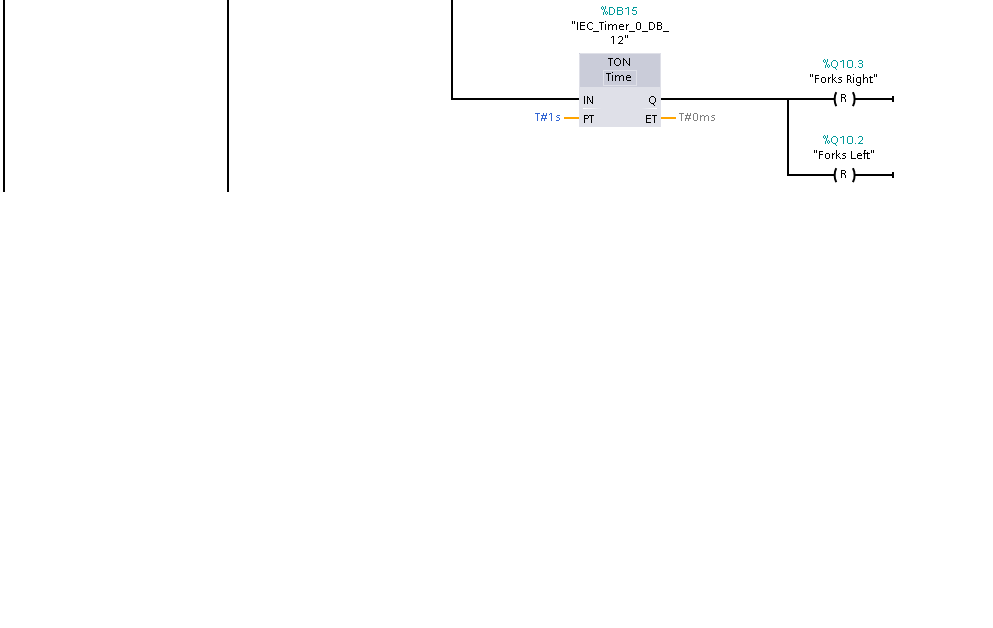
Рисунок 12 – Возврат лифта

Переходим к действиям у стеллажа. Задачей здесь является: приехать к ячейке, ровно и аккуратно положить груз на стеллаж и уехать в начальное положение, учитывая, что стеллаж есть с обеих сторон, которые он должен заполнить.

От условий перемещения предмета и горящей лампы кнопки старт идет еще условие, что режим выбран завоза на склад, от которого:

Лифт включен: “Движение по какой-то из осей” M0.2 выключен, что будет означать, что кран двигался, закончил движение и уже у нужной ячейки. А здесь, если маркер M0.7 “По правую сторону” включен, то балки выдвигаются влево Q10.2 “Forks left” (Set), а если выключен, то вправо Q10.3 “Forks right” (Set). Теперь, если балки выдвинулись вправо, то включается маркер M0.7 “по правую сторону” (Set), а если они оказались слева, то этой маркер выключается (Reset), также, в обоих случаях выключается лифт (Reset).

Лифт выключен: с задержкой “TON” в 1 секунду балки возвращаются в середину, не зависимо в какую сторону они были выдвинуты, а когда они окажутся в середине, загорится маркер M0.3 “Возврат лифта” (см. Рисунок 13).

Рисунок 13 – Действия у стеллажа в режиме завоза на склад

Теперь, перейдем к другому режиму. От тех же условий перемещения предмета ставим режим “Вывоз со склада” I6.5, с тройным разветвлением.

В первом случае его обычная остановка: лифт выключен, маркер движения какой-то из осей не горит, а здесь, если маркер “По правую сторону” выключен