

- DESIGN
- DEVELOPMENT
- CONSTRUCTION
- SUPPLY

**ИСПОЛЬЗОВАНИЕ ЯЗЫКОВ СТАНДАРТА IEC 61131-3
(МЭК61131-3) ПРИ ПРОЕКТИРОВАНИИ
СОВРЕМЕННЫХ МИКРОПРОЦЕССОРНЫХ СИСТЕМ
ЖЕЛЕЗНОДОРОЖНОЙ АВТОМАТИКИ И
ТЕЛЕМЕХАНИКИ**

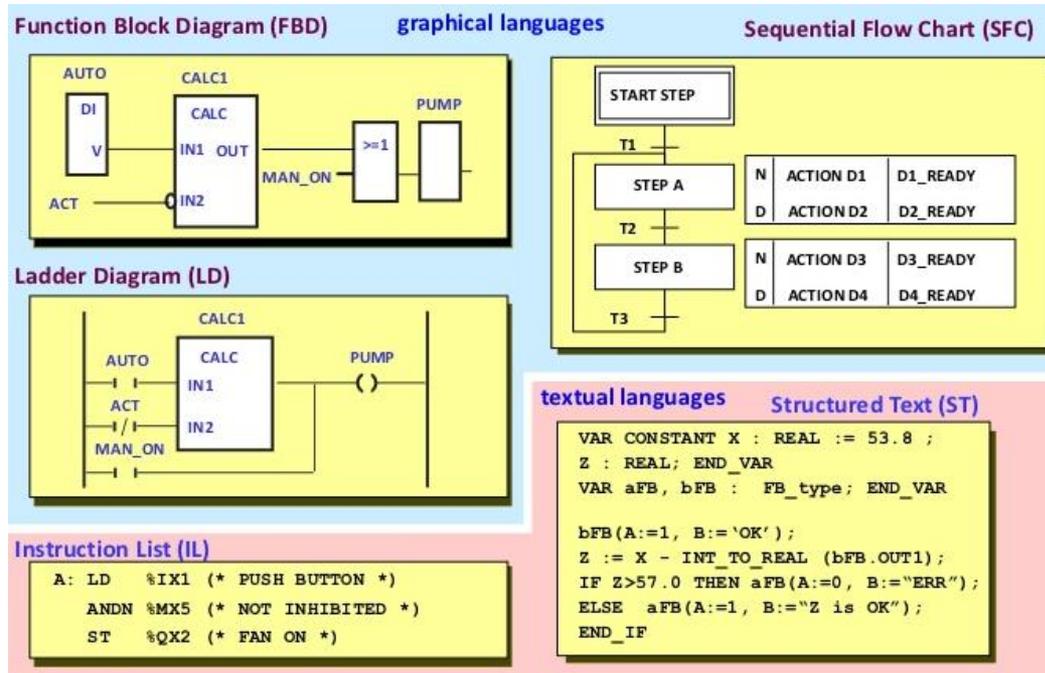
**г.Черноморск
27-29.09.2016**



Применение на рельсовом транспорте микропроцессорных систем железнодорожной автоматики и телемеханики (ЖАТ) с открытой архитектурой по принципам АСУ ТП где под технологией производства используется технология работы железнодорожного транспорта позволяет в рамках существующего штата хозяйства сигнализации и связи сформировать собственное эксплуатационное подразделение способное самостоятельно обслуживать систему в соответствии с принципами и порядком обслуживания существующих релейных систем.

Для этого заказчику передается полный пакет технической документации, включая прикладное ПО, которое отображается в доступном графическом виде и повторяет релейные схемы СЦБ. Распечатанный текст ППО позволяет сохранять исходное прикладное программное обеспечение и вносить изменения в соответствии с действующей на железнодорожном транспорте процедуры внесения изменений в релейных схем СЦБ.

При разработке прикладного программного обеспечения для системы железнодорожной автоматики нашего предприятия используются языки стандарта МЭК 61131-3 выпущенного в свет Международной электротехнической комиссией в 1993 году и входящего в группу МЭК 61131 стандартов, которые охватывают различные аспекты использования программируемых логических контроллеров - ПЛК.





Поскольку программы пишутся человеком и исключительно для человека, то в силу особенностей человеческой психики языки должны быть просты в изучении. Кроме того, языки должны предоставлять механизмы структуризации алгоритма (в нашем случае – языковые средства организации совместного функционирования логически параллельных частей) и механизмы абстрагирования (в нашем случае – понятийный переход от датчиков и исполнительных органов к целевому технологическому процессу).



Главным требованием к ПЛК всегда была и остается возможность его эксплуатации существующим техническим персоналом и возможность быстрой замены старого оборудования. Поэтому языки программирования компьютеров и встраиваемых микропроцессорных систем управления плохо подходят для программирования ПЛК. Здесь нужны более простые и наглядные языки, позволяющие излагать задачу в близких к применяемым технологиям категориях. Привлечение же к программированию специализированной фирмы неизбежно порождает зависимость, если реализация не является достаточно прозрачной.

Системы программирования, основанные на МЭК 61131-3, характеризуются следующими показателями:

- надежностью создаваемого программного обеспечения которая обеспечивается тем, что программы для ПЛК создаются с помощью специально предназначенной для этого среды разработки, которая содержит все необходимые средства для написания, тестирования и отладки программ с помощью эмуляторов и реальных ПЛК, а также множество готовых фрагментов программного кода;
- возможностью простой модификации программы и наращивания ее функциональности;
- переносимостью проекта с одного ПЛК на другой;
- возможностью повторного использования отработанных фрагментов программы;
- простотой языка и ограничением количества его элементов.



Языки МЭК 61131-3 появились не как теоретическая разработка, а как результат анализа множества языков, уже используемых на практике и предлагаемых рынку производителями ПЛК.

Стандарт устанавливает пять языков программирования со следующими названиями:

- структурированный текст (ST - Structured Text);
- последовательные функциональные схемы (SFC - "Sequential Function Chart");
- диаграммы функциональных блоков (FBD - Function Block Diagram);
- релейно-контактные схемы, или релейные диаграммы (LD - Ladder Diagram);
- список инструкций (IL - Instruction List).

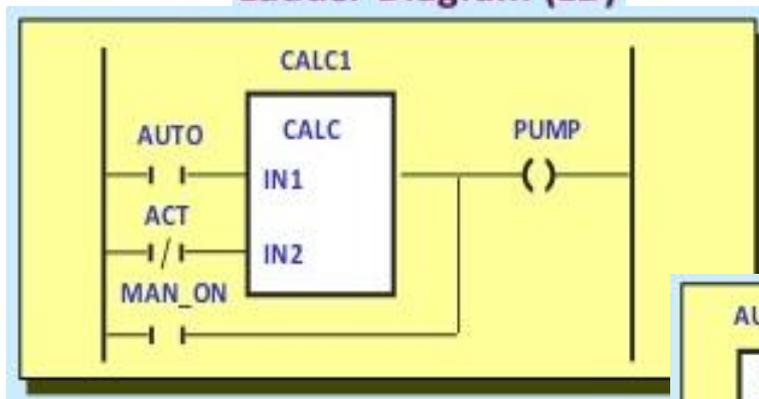
Графическими языками являются SFC, FBD, LD. Языки IL и ST являются текстовыми.

Языки МЭК 61131-3 базируются на следующих принципах:

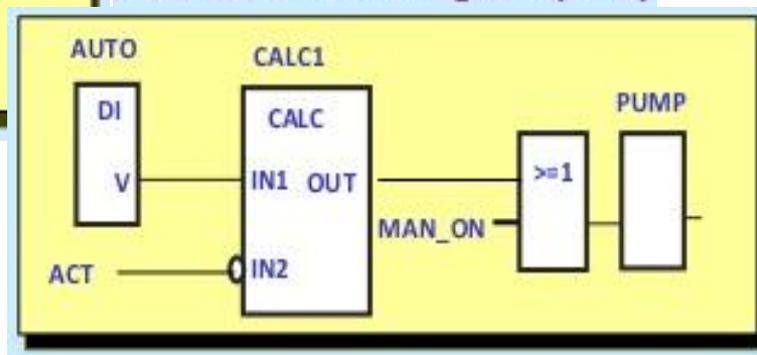
- вся программа разбивается на множество функциональных элементов - Program Organization Units (POU), каждый из которых может состоять из функций, функциональных блоков и программ. Любой элемент МЭК-программы может быть сконструирован иерархически из более простых элементов;
- стандарт требует строгой типизации данных. Указание типов данных позволяет легко обнаруживать большинство ошибок в программе до ее исполнения;
- имеются средства для исполнения разных фрагментов программы в разное время, с разной скоростью, а также параллельно. Например, один фрагмент программы может сканировать концевой датчик с частотой 100 раз в секунду, в то время как второй фрагмент будет сканировать другой датчик с частотой один раз в 10 сек;
- стандарт обеспечивает совместное использование всех пяти языков, поэтому для каждого фрагмента задачи может быть выбран любой, наиболее удобный, язык;
- программа, написанная для одного контроллера, может быть перенесена на любой контроллер, совместимый со стандартом МЭК 61131-3.

Наше предприятие использует в основном два языка программирования – релейной логики LD и функциональных блоков FBD как наиболее близких и понятных эксплуатационному штату обслуживающему релейные системы железнодорожной автоматики телемеханики.

Ladder Diagram (LD)



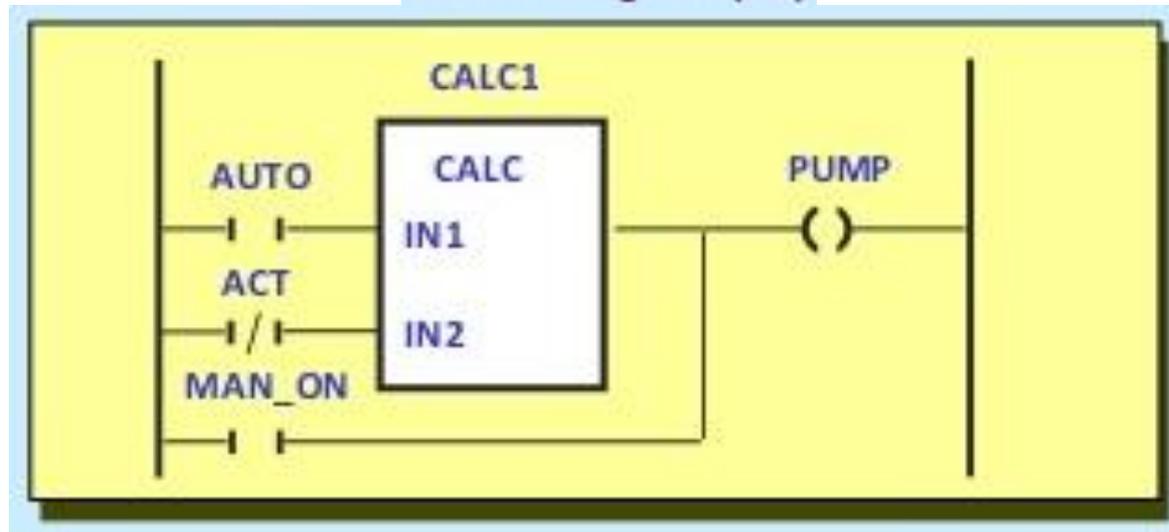
Function Block Diagram (FBD)



• LD

Графический язык релейной логики впервые появился в виде электрических схем, которые состояли из контактов и обмоток электромагнитных реле. Язык релейной логики интуитивно понятен людям, знакомым с электротехникой и поэтому оказался наиболее распространенным в промышленной автоматике. Наглядно показано на примере №1 и №2. Обслуживающий персонал легко находил отказ в оборудовании, прослеживая путь сигнала по релейной диаграмме.

Ladder Diagram (LD)



LD Пример №1

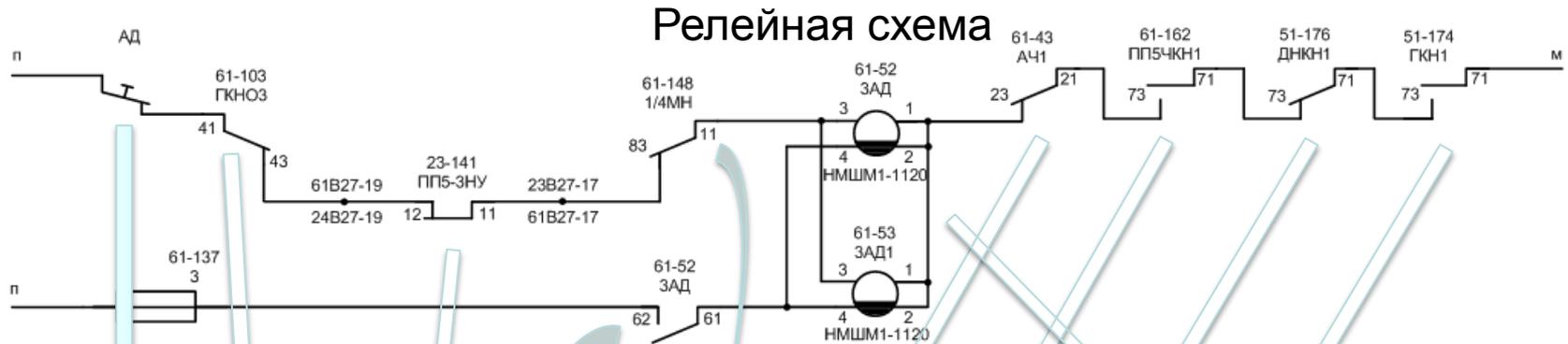
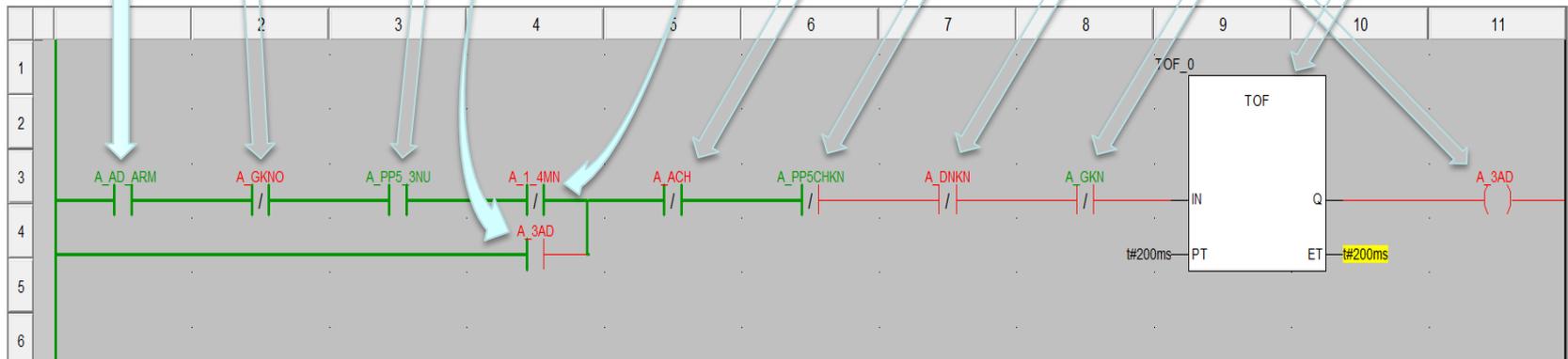


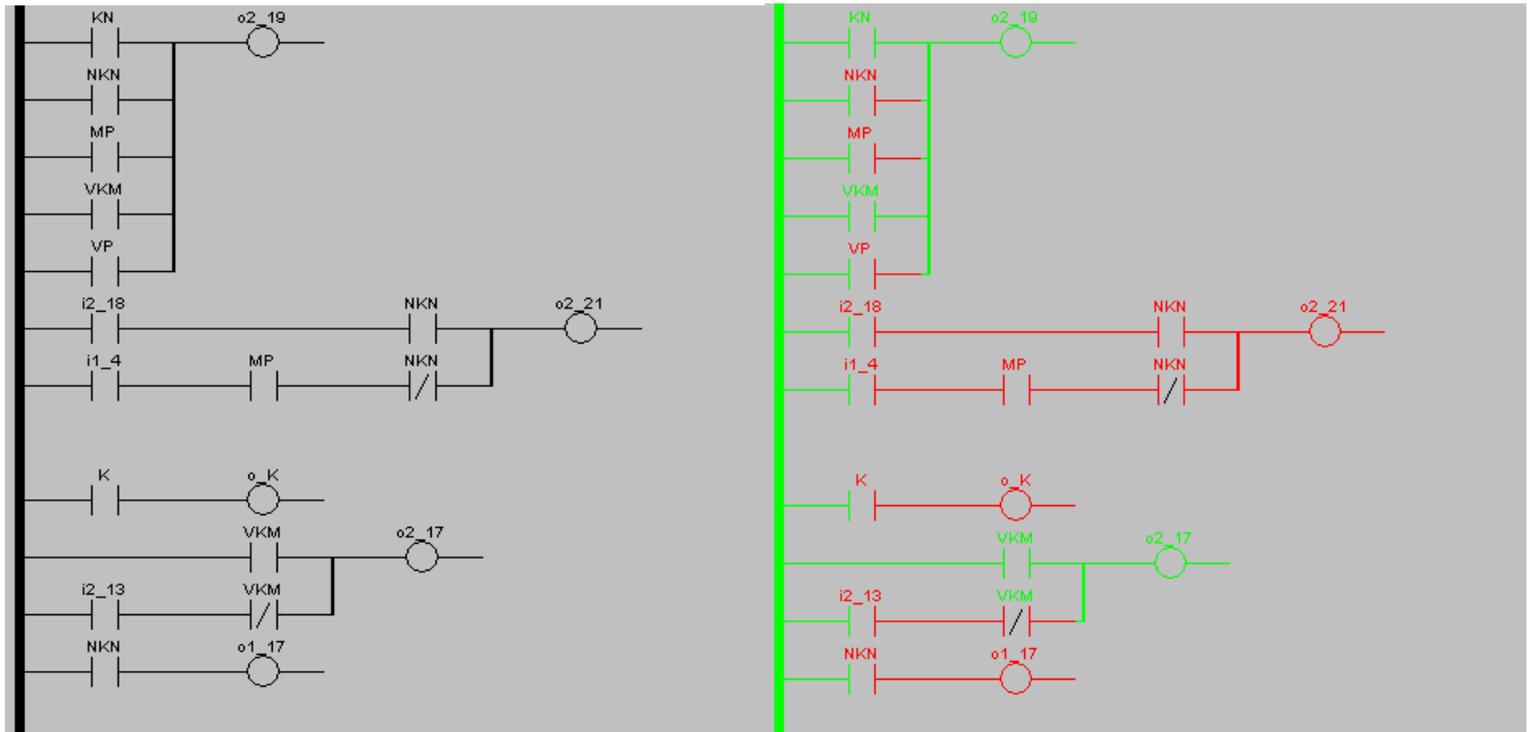
Схема реле автооборота системы МРЦ (действующая схема метрополитена)

Схема на языке LD в режиме реального времени (режим online).



LD Пример №2

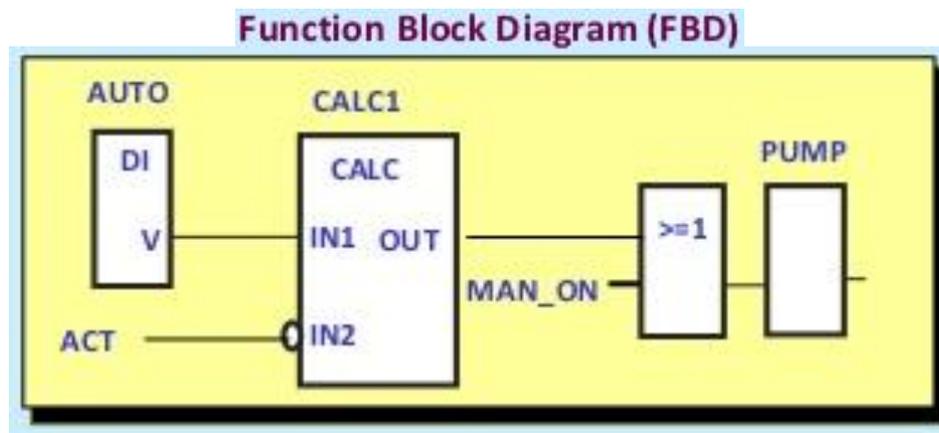
Фрагмент блока наборной группы НМ1 на языке релейно-контактных схем (LD)



• FBD

FBD является графическим языком и наиболее удобен для программирования процессов прохождения сигналов через функциональные блоки. Язык FBD удобен для схемотехников, которые легко могут составить электрическую схему системы управления на "жесткой логике", но не имеют опыта программирования. Наглядно показано на примере №3 и №4.

Функциональные блоки представляют собой фрагменты программ, написанных на других языках, которые могут быть многократно использованы в разных частях программы и которым соответствует графическое изображение, принятое при разработке функциональных схем электронных устройств.



• FBD Пример №3

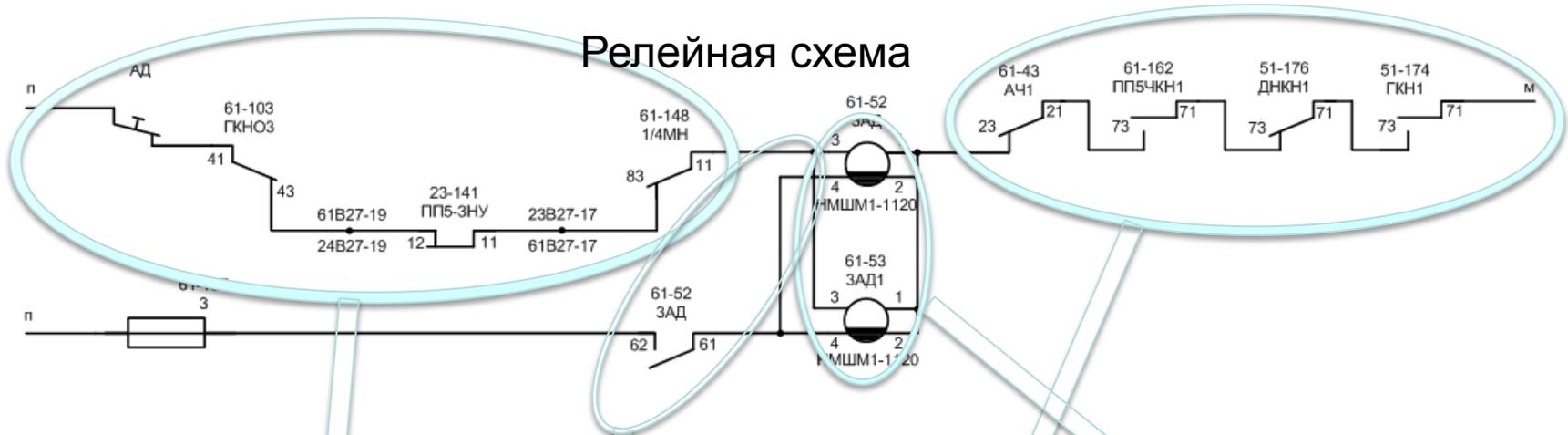
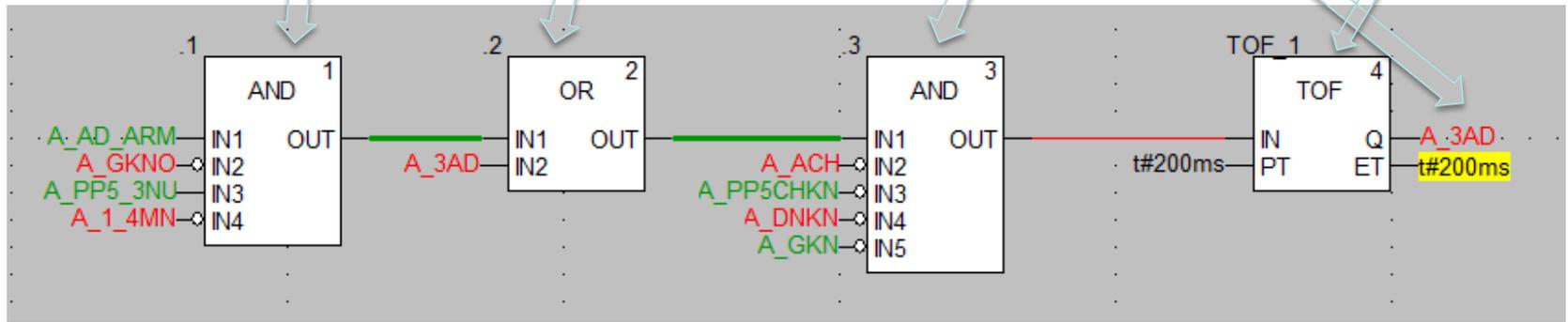


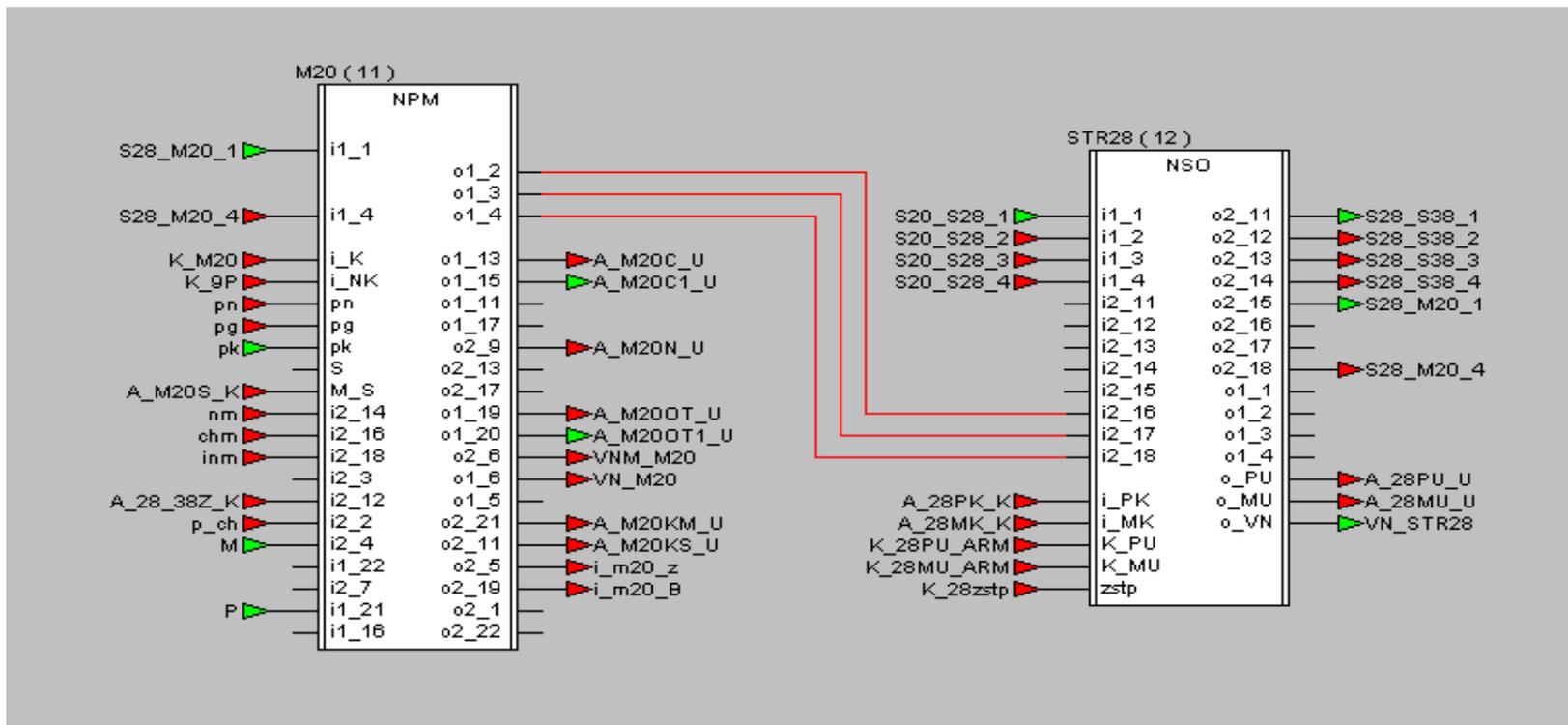
Схема на языке FBD в режиме реального времени (режим online).



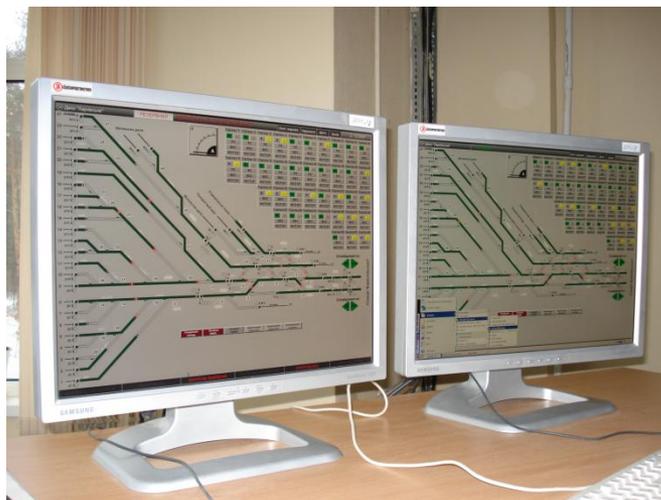
Замедление

• FBD Пример №4

Внешний вид функциональных блоков НПМ и НСО на основе альбома МРЦ13, в режиме реального времени (режим online).



Большинство систем автоматизации функционирует с участием человека – оператора, диспетчера. Человеко – машинный интерфейс предназначен для взаимодействия человека с автоматизированным технологическим процессом, его называют SCADA-системой (Supervisory Control And Data Acquisition). Этот термин переводится буквально как "диспетчерское управление и сбор данных", но на практике его трактуют гораздо шире, а современные SCADA-пакеты включают в себя широчайший набор функциональных возможностей, далеко выходящий за рамки сбора данных и диспетчерского управления.





Существующие в настоящее время SCADA-пакеты выполняют множество функций, которые можно разделить на несколько групп:

- настройка SCADA на конкретную задачу (т. е. разработка программной части системы автоматизации);
- диспетчерское управление;
- автоматическое управление;
- хранение истории процессов;
- выполнение функций безопасности;
- выполнение общесистемных функций.

Несмотря на множество функций, выполняемых SCADA, основным ее отличительным признаком является наличие интерфейса с пользователем.



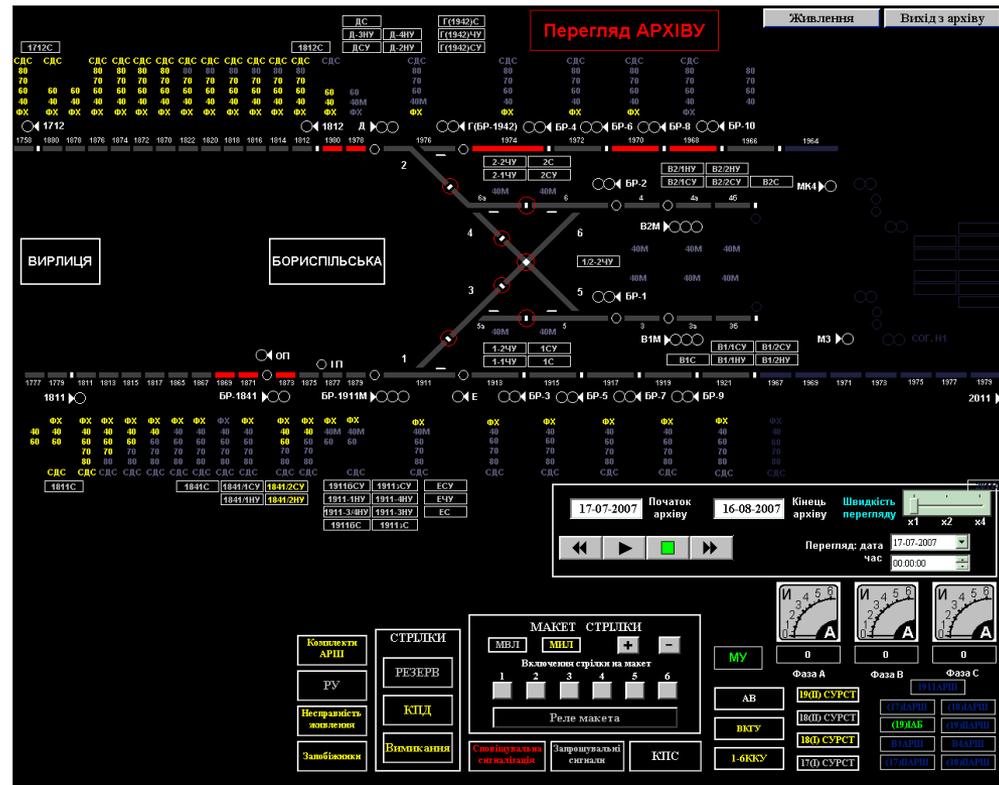
Качество решений, принятых оператором (диспетчером), часто влияет не только на качество выполняемой работы, но и на жизнь людей. Поэтому комфорт рабочего места, понятность интерфейса, наличие подсказок и блокировка явных ошибок оператора являются наиболее важными свойствами SCADA, а дальнейшее их развитие осуществляется в направлении улучшения эргономики и создания экспертных подсистем.



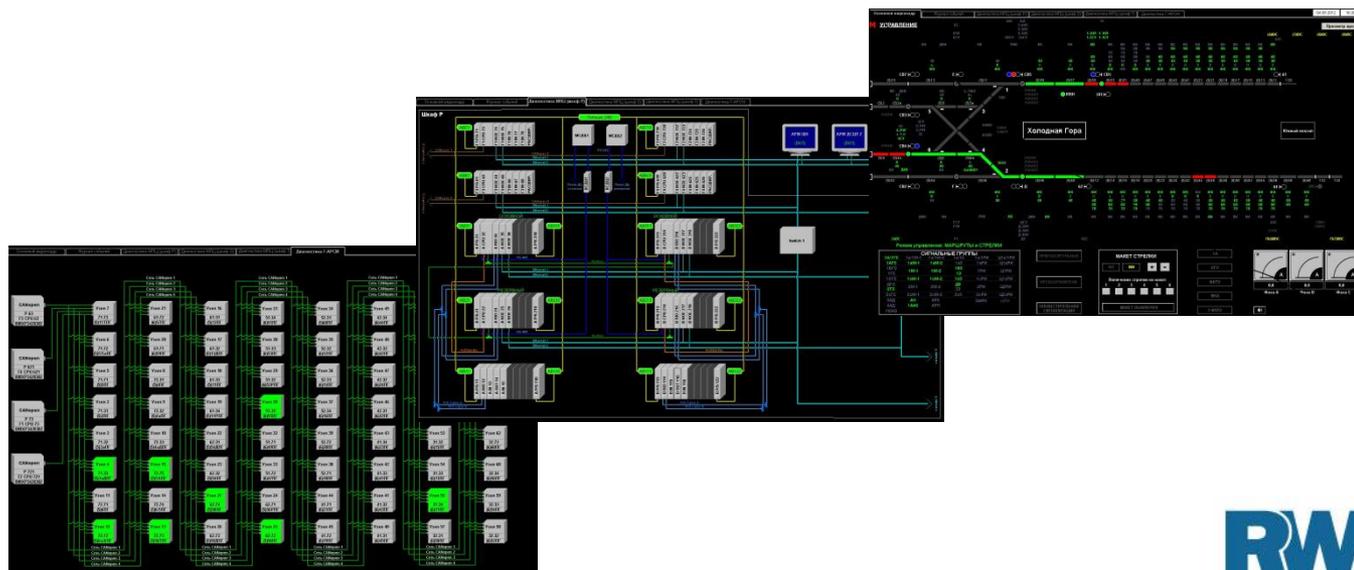
- В SCADA-пакетах используют понятие аларма и события.

Событие - это изменение некоторых состояний в системе. События не требуют срочного вмешательства оператора, а просто информируют его о состоянии системы.

В отличие от события, аларм (от английского "alarm" - "сигнал тревоги") представляет собой предупреждение о важном событии, в ответ на которое нужно срочно предпринять некоторые действия.



- В процесс разработки человеко – машинного интерфейса входят следующие операции:
 - создание графического интерфейса (мнемосхем, графиков, таблиц, всплывающих окон, элементов для ввода команд оператора и т.д.);
 - программирование и отладка алгоритмов работы системы автоматизации. Многие SCADA позволяют выполнять отладку системы как в режиме эмуляции оборудования, так и с подключенным оборудованием;
 - настройка системы коммуникации (сетей, модемов, коммуникационные контроллеры и т.п.);
 - создание баз данных и подключение к ним SCADA.





Как система диспетчерского управления SCADA может выполнять следующие задачи:

- взаимодействие с оператором (выдача визуальной и слуховой информации, передача в систему команд оператора);
- помощь оператору в принятии решений (функции экспертной системы);
- автоматическая сигнализация об авариях и критических ситуациях;
- выдача информационных сообщений на пульт оператора;
- ведение журнала событий в системе;
- извлечение информации из архива и представление ее оператору в удобном для восприятия виде;
- подготовка отчетов;
- учет наработки технологического оборудования.



SCADA обычно выполняет следующие задачи автоматического управления:

- автоматическое регулирование;
- управление последовательностью операций в системе автоматизации;
- адаптация к изменению условий протекания технологического процесса;
- автоматическая блокировка исполнительных устройств при выполнении заранее заданных условий.

Знание предыстории управляемого процесса позволяет улучшить будущее поведение системы, проанализировать причины возникновения опасных ситуаций, выявить ошибки оператора.

Для создания истории система выполняет следующие операции:

- сбор данных и их обработка;
- архивирование данных (действий оператора, собранных и обработанных данных, событий, алармов, графиков, экранных форм, файлов конфигурации, отчетов и т. п.);
- управление базами данных (реального времени и архивных).

Журнал неисправностей					
	Час появи	Час зникнення	Несправність	стор. 0001	стр. 00001
1	06.06.2008 03:34:52		- Вмикання запрошувального сигнала		
2	06.06.2008 03:34:40		- Вмикання запрошувального сигнала		
3	06.06.2008 02:46:29		- Вмикання загальної КШРМ		
4	06.06.2008 02:45:10	06.06.2008 02:48:26	Контроль ЗС		
5	06.06.2008 02:45:05		- Вмикання запрошувального сигнала		
6	06.06.2008 02:44:21		- Вмикання запрошувального сигнала		
7	06.06.2008 02:44:10	06.06.2008 02:44:36	Контроль ЗС		
8	06.06.2008 02:44:05		- Вмикання запрошувального сигнала		
9	06.06.2008 02:43:06		- Вмикання запрошувального сигнала		
10	06.06.2008 02:42:48	06.06.2008 02:43:32	Контроль ЗС		
11	06.06.2008 02:42:45		- Вмикання запрошувального сигнала		
12	06.06.2008 02:28:11		- Вмикання запрошувального сигнала		
13	06.06.2008 02:28:02		- Вмикання запрошувального сигнала		
14	06.06.2008 02:25:01		- Вмикання запрошувального сигнала		
15	06.06.2008 02:24:42		- Вмикання запрошувального сигнала		



Применение SCADA в системах удаленного доступа через интернет резко повысило уязвимость SCADA к действиям враждебных лиц.

Для повышения безопасности SCADA используют следующие методы:

- разграничение доступа к системе между разными категориями пользователей (разные права доступа к информации и к модификации настроек системы);
- защиту информации (путем шифрования информации и обеспечения секретности протоколов связи);
- обеспечение безопасности оператора благодаря его отдалению от опасного управляемого процесса (дистанционное управление), дистанционный контроль и дистанционное управление выполняются по проводной сети, радиоканалу (через GSM- или радиомодем), через интернет и т.д.;
- специальные методы защиты от кибер-атак;
- применение межсетевых экранов.

ВИЗУАЛІЗАЦІЯ

ЗА АРМ ДСП **КОНТРОЛЬ** 27.09.2007 09:11:11 Адміністратор

ст. КОВ'ЯГИ

РЕЖИМ СИГНАЛІВ

ДЕНЬ | НІЧ | АВТ. РЕГУЛ. | РУЧН. РЕГУЛ. | ЗНИЖ. НАПР. | ФІДЕР 1 | ФІДЕР 2 | ВИКЛ. ФІДЕР

ПОІЗН. | ВІЛЬН. | МАНЕВР. | ШТУЧИНЕ РОЗМІК.

АВТОД. СИГН. | **ОЧИСТКА**

НЕПАРН. | ПАРН. | ЗАПОВІД. | ЗВІЛЯ. | ВИБІР. | ЦИЛІЛ.

ЗАМИК. СТР.

ВИПРЯМ. | КОНТР. ЦВП. | НЕПАРН. | ПАРН. | РОЗРІЗ СТР.

РЕЗЕРВ. ВВ. | КОНТР. МІГР. | ВИКЛ. СТР.

ВІДМІНА

РЕЗЕРВ. 2 КВЛ. | КОНТР. ЖИЛ. КАБ. | ЗАМИК. ДІЛ. КОЛ. | ЗАМИК. ПЕРЕГОНУ.

2ЧСК.

ПОНАБ | **КОНТРОЛЬ** | **ПЕРЕГРІВ**

ПЕРЕЇЗД ПК 461+75 | ПЕРЕЇЗД ПК 485+88 | ПЕРЕЇЗД ПК | ПЕРЕЇЗД ПК 527+65

СПРАВН. | КОНТР. | ЗАКР. | КОНТР. В. | СПРАВН. | КОНТР. | ЗАКР. | КОНТР. В. | СПРАВН. | КОНТР. | ЗАКР. | КОНТР. В.

ОГУЛЬЦІ

ЗАМИК. ПЕРЕГОНУ | ЗАМИК. ДІЛ. КОЛ. | КОНТР. ЖИЛ. КАБ. | РЕЗЕРВ. 2 КВЛ.

2ЧСК.

РЕЖИМ СИГНАЛІВ

ДЕНЬ | НІЧ | АВТ. РЕГУЛ. | РУЧН. РЕГУЛ.

ПОІЗН. | ВІЛЬН. | МАНЕВР. | ШТУЧИНЕ РОЗМІК.

ЗАПРОШУВАЛЬНІ СИГНАЛИ

НД | Н | Ч2 | Ч

Н1 | ЧД | Ч

ДЗВІНКИ

ДЗВІНОК ФІДЕРА | ДЗВІНОК РОЗРІЗУ

ВИКЛ. СТР. | ГРУПОВА ВІДМІНА

МАКЕТ

СТРІЛКА №

АВТОДІЯ СИГН. НЕПАРН. | АВТОДІЯ СИГН. ПАРН.

ВОДЯНА

ЗМІНА НАПР. | ДАВАН. ЗГОДИ | ДОПОМ. ПРИЙМ. | ПІДХІД "НД" | РОЗМІК. ДІЛ. КОЛ. | РОЗМІК. ПЕРЕГОН. | ЗАМИК. ПЕРЕГОН.

ЗМІНА НАПР. | ДАВАН. ЗГОДИ | ДОПОМ. ПРИЙМ. | ПІДХІД "Н" | РОЗМІК. ДІЛ. КОЛ. | РОЗМІК. ПЕРЕГОН. | ЗАМИК. ПЕРЕГОН.

СТРІЛКИ НЕПАРНІ

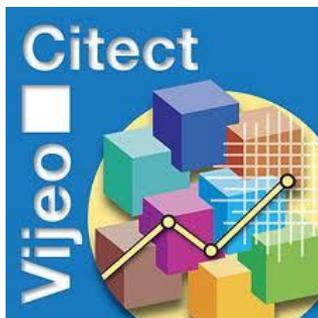
1	3	5	7	9
+	-	+	-	+
В	В	В	В	В

СВІТЛОФОРИ

М1	М11	Ч1	НД	Ч1	Н1	Ч
М3	М13	Ч2	Н	Ч2	Н2	ЧД
М5		Ч3		Ч3	Н3	
М7		Ч4		Ч4	Н4	
М9		Ч5		Ч5	Н5	

СТРІЛКИ ПАРНІ

2	4	6	8	10	12	14
+	-	+	-	+	-	+
В	В	В	В	В	В	В



Для разработки интерфейсов АРМ дежурного по станции, диспетчера и электромеханика используется пакет программного обеспечения Vijeo Citect (Schneider Electric).

SCADA система Citect является одним из лидирующих программных продуктов для систем мониторинга, управления и сбора данных (Supervisory, Control And Data Acquisition). По всему миру установлено более 150 000 лицензий практически во всех отраслях промышленности.

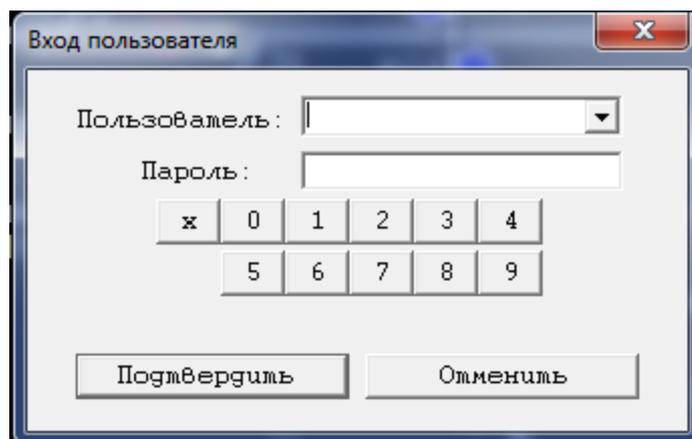
Высокая производительность определяется тем, что SCADA система Citect построена на базе мультитасового ядра реального времени. SCADA система Citect может работать с большими объемами данных и при увеличении количества параметров время отклика изменяется незначительно.



Данный подход к созданию ППО позволяет эксплуатировать системы железнодорожной автоматики телемеханики разработки нашего предприятия существующим штатом специалистов отрасли.

При введении в эксплуатацию системы заказчику передается полный пакет технической документации, включая прикладное ПО, которое отображается в доступном графическом виде и, как вы уже увидели, повторяет релейные схемы СЦБ. Распечатанный текст ППО позволяет сохранять исходное прикладное программное обеспечение и вносить изменения в соответствии с действующей на железнодорожном транспорте процедуры внесения изменений в релейных схем СЦБ.

Для внесения изменений в ППО не требуются программисты - достаточно специалистов отрасли с высшим железнодорожным образованием, прошедшие обучение или курсы повышения квалификации на базе УкрДУЗТ, в учебных центрах нашего предприятия, непосредственно на объекте во время проведения пуско-наладочных работ. Этим специалистам предоставляются права на внесение изменений в ППО аналогично действующему порядку эксплуатации релейных систем (внесение изменений в экземпляр службы, дистанции, околотка).



Вход пользователя

Пользователь:

Пароль:

x	0	1	2	3	4
5	6	7	8	9	

Подтвердить Отменить



Для защиты от несанкционированного внесения изменений в ППО эксплуатационному персоналу предоставляются соответствующих уровня доступа:

- просмотр;
- просмотр и внесение изменений;
- запрет просмотра.

Список лиц, имеющих право на внесение изменений в ППО определяется руководством дистанции и утверждается руководством службы сигнализации и связи. Вышеописанный принцип относится ко всем разновидностям систем железнодорожной автоматики и телемеханики - станционным, перегонным, горочной централизации, переездной сигнализации, ДК, ДЦ и т.д.

• СПАСИБО ЗА ВНИМАНИЕ !

ООО «НПП «ЖЕЛДОРАВТОМАТИКА»

Адрес:

пр. Науки 36, г. Харьков
Харьковская обл., Украина

Тел/факс: +38 (057) 719-62-45,
+38 (057) 755-19-62;

ж.д. 4-10-27;
5-55-95.

Эл.почта: rwa@rwa.com.ua

<http://www.rwa.ua>

Контактное лицо для вопросов и предложений:

Гаевский Виталий Викторович