

# **Z21**

## **Спецификация протокола локальной сети**

# Юридический, отказ от ответственности

Компания Modeleisenbahn GmbH прямо заявляет, что ни в коем случае не несет юридической ответственности за содержание в этом документе или за любую дополнительную информацию, указанную в этом документе. Юридическая ответственность лежит исключительно на пользователе предоставленных данных или на издателе соответствующей дополнительной информации.

За все убытки, возникшие в результате использования или неиспользования указанной информации, компания Modeleisenbahn GmbH, Plainbachstraße 4, A-5101 Бергхайм, Австрия, не несет никакой ответственности. Модельная железная дорога ГмбХ, Plainbachstraße 4, A-5101 Бергхайм, Австрия, не несет никакой ответственности за актуальность, правильность, полноту или качество предоставленной информации. Претензии об ответственности, связанные с материальным, нематериальным или материальным ущербом, причиненным в результате использования или неиспользования предоставленной информации, в принципе исключаются. Модельная железная дорога ГмбХ, Плейнбахштрассе 4, A-5101 Бергхайм, Австрия, оставляет за собой право изменять, дополнять или удалять предоставленную информацию без отдельного уведомления.

Все товарные знаки и товарные знаки, упомянутые в документе и защищенные третьими сторонами, где это применимо, полностью подчиняются положениям действующего закона о товарных знаках и правам собственности соответствующих зарегистрированных владельцев.

Авторские права на опубликованную информацию, подготовленную Model Eisenbahn GmbH, Plainbachstraße 4, A-5101 Бергхайм, Австрия, в любом случае остаются за Model Eisenbahn GmbH, Plainbachstraße 4, A-5101 Бергхайм, Австрия, исключительно за Model Eisenbahn GmbH, Plainbachstraße 4, A-5101 Бергхайм, Австрия. Любая публикация или использование предоставленной информации в других электронных или печатных публикациях не допускается без явного согласия.

Если части или отдельные формулировки Заявления об отказе от ответственности не соответствуют, перестают соответствовать или не полностью соответствуют действующему законодательству, остальные части Заявления об отказе от ответственности остаются неизменными по своему содержанию и действительности.

## Отпечаток

Apple, iPad, iPhone, iOS are trademarks of Apple Inc., registered in the U.S. and other countries.  
App Store is a service mark of Apple Inc.  
Android является товарным знаком Google Inc.  
Google Play является знаком обслуживания Google Inc.  
RailCom является зарегистрированной торговой маркой компании Lenz Electronics GmbH.  
Motorola is a registered trademark of Motorola Inc., Tempe-Phoenix, USA.  
LocoNet is a registered trademark of Digitrax, Inc.

Все права, изменения, ошибки и способы доставки защищены.  
Спецификации и иллюстрации без гарантии. Мы оставляем за собой право на изменение.

*Издатель: Model Eisenbahn GmbH, Plainbachstraße 4, A-5101 Бергхайм, Австрия*

Документ версии 1.04

25.03.2014

2/48

Спецификация протокола локальной сети Z21



## История изменений

Дата	Изменение версии документа	
06.02.2013	1.00	Описание интерфейса локальной сети для Z21 FW версии 1.10, 1.11 и SmartRail FW версии 1.12
20.03.2013	1.01	Z21 FW версия 1.20 LAN_SET_BROADCASTФЛАГИ: новые флаги

		LAN_GET_HWINFO: новая команда LAN_SET_TURNOUTМОД: ММ формат ЛокоНет: функциональность шлюза
29.10.2013	1.02	SmartRail FW версии 1.13 LAN_GET_HWINFO: новая команда <b>Z21 FW версия 1.22:</b> декодер для чтения и записи резюме POM Чтение и вспомогательный декодер: новые команды <b>LocoNet Dispatch и детектор загруженности путей</b> LAN_LOCONET_DISPATCH_ADDR : новый ответ LAN_SET_BROADCASTФЛАГИ: новый флаг LAN_LOCONET_DETECTOR: новая команда <b>Z21 FW версия 1.23</b> Исправление длинного адреса транспортного средства в главе 4 Вождение LAN_X_MM_WRITE_BYTE LAN_LOCONET_DETECTOR: расширение для Лисси
12.02.2014	1.03	LAN_SET_BROADCASTФЛАГИ: флаг 0x00010000 Переключение Глава 5: Объяснение переключения адресации LAN_X_GET_TURNOUT_INFO: расширение бит очереди
25.03.2014	1.04	LAN_LOCONET_DETECTOR: расширение для Лисси LAN_SET_BROADCASTФЛАГИ: флаг 0x00010000 Переключение Глава 5: Объяснение переключения адресации LAN_X_GET_TURNOUT_INFO: расширение бит очереди



## Содержание

<b>1</b>	<b>ОСНОВЫ</b>	<b>7</b>
<b>1.1</b>	<b>Общение</b>	<b>7</b>
<b>1.2</b>	<b>Набор данных Z21</b>	
1.2.1	Конструкция	7
1.2.2	Туннелирование протокола X-BUS	7
1.2.3	Туннелирование LocoNet	8
<b>1.3</b>	<b>Объединение записей в один UDP-пакет</b>	<b>9</b>
<b>2</b>	<b>СИСТЕМА, СОСТОЯНИЕ, ВЕРСИИ</b>	<b>10</b>

2.1	LAN_GET_SERIAL_NUMBER .....	10
2.2	LAN_LOGOFF .....	10
2.3	LAN_X_GET_VERSION .....	10
2.4	LAN_X_GET_STATUS .....	11
2.5	LAN_X_SET_TRACK_POWER_OFF .....	11
2.6	LAN_X_SET_TRACK_POWER_ON.....	11
2.7	LAN_X_BC_TRACK_POWER_OFF.....	12
2.8	LAN_X_BC_TRACK_POWER_ON.....	12
2.9	LAN_X_BC_PROGRAMMING_MODE .....	12
2.10	LAN_X_BC_TRACK_SHORT_CIRCUIT .....	12
2.11	LAN_X_UNKNOWN_COMMAND .....	13
2.12	LAN_X_STATUS_CHANGED.....	13
2.13	LAN_X_SET_STOP .....	14
2.14	LAN_X_BC_STOPPED .....	14
2.15	LAN_X_GET_FIRMWARE_VERSION .....	14
2.16	LAN_SET_BROADCASTФЛАГИ .....	15
2.17	LAN_GET_BROADCASTФЛАГИ .....	16
2.18	LAN_SYSTEMSTATE_DATACHANGED .....	16
2.19	LAN_SYSTEMSTATE_GETDATA .....	18
2.20	LAN_GET_HWINFO .....	18

Документ версии 1.04

25.03.2014

4/48



Спецификация протокола локальной сети Z21

3	НАСТРОЙКИ .....	19
3.1	LAN_GET_LOCOMODE .....	19
3.2	LAN_SET_LOCOMODE .....	19
3.3	LAN_GET_TURNOUTMODE.....	20
3.4	LAN_SET_TURNOUTMODE .....	20
4	ВОЖДЕНИЕ .....	21
4.1	LAN_X_GET_LOCO_INFO .....	21
4.2	LAN_X_SET_LOCO_DRIVE .....	22
4.3	LAN_X_SET_LOCO_FUNCTION .....	22
4.4	LAN_X_LOCO_INFO .....	23
5	ПЕРЕКЛЮЧЕНИЕ .....	24
5.1	LAN_X_GET_TURNOUT_INFO .....	25
5.2	LAN_X_SET_TURNOUT .....	
5.2.125	LAN_X_SET_TURNOUT с Q=0 .....	25
5.2.2	LAN_X_SET_TURNOUT с Q=1 .....	27

5.3	LAN_X_TURNOUT_INFO.....	28
6	ДЕКОДЕР ДЛЯ ЧТЕНИЯ И ЗАПИСИ РЕЗЮМЕ .....	29
6.1	LAN_X_CV_READ .....	29
6.2	LAN_X_CV_WRITE .....	29
6.3	LAN_X_CV_NACK_SC .....	29
6.4	LAN_X_CV_NACK .....	30
6.5	LAN_X_CV_RESULT .....	30
6.6	LAN_X_CV_POM_WRITE_BYTE .....	31
6.7	LAN_X_CV_POM_WRITE_BIT .....	31
6.8	LAN_X_CV_POM_READ_BYTE .....	32
6.9	LAN_X_CV_POM_ACCESSORY_WRITE_BYTE .....	33
6.10	LAN_X_CV_POM_ACCESSORY_WRITE_BIT .....	33
6.11	LAN_X_CV_POM_ACCESSORY_READ_BYTE .....	34
6.12	LAN_X_MM_WRITE_BYTE .....	35

Документ версии 1.04

25.03.2014

5/48

**Roco**

Спецификация протокола локальной сети Z21

7	R-BUS ОБРАТНОЙ СВЯЗИ .....	36
7.1	LAN_RMBUS_DATACHANGED .....	36
7.2	LAN_RMBUS_GETDATA .....	36
7.3	LAN_RMBUS ПРОГРАММНЫЕ МОДУЛИ .....	37
8	ЖЕЛЕЗНОДОРОЖНЫЙ КОМ .....	38
8.1	LAN_RAILCOM_DATACHANGED .....	38
8.2	LAN_RAILCOM_GETDATA .....	39
9	ЛОКОНЕТ.....	40
9.1	LAN_LOCONET_Z21_RX .....	41
9.2	LAN_LOCONET_Z21_TX .....	41
9.3	LAN_LOCONET_FROM_LAN .....	41
9.4	LAN_LOCONET_DISPATCH_ADDR .....	42
9.5	LAN_LOCONET_DETECTOR .....	43
	ANHANG A - BEFEHLSÜBERSICHT .....	46
	Client an Z21 .....	46
	Z21 клиент .....	47
	КАТАЛОГ ИЛЛЮСТРАЦИЙ .....	48
	ТАБЛИЧНЫЙ КАТАЛОГ .....	48



Спецификация протокола локальной сети Z21

# 1 Основы

## 1.1 Общение

Связь с Z21 осуществляется по протоколу UDP через порты 21105 или 21106 .

Управляющие приложения на клиенте (ПК, приложение, ...) должны в первую очередь использовать порт 21105.

Связь всегда осуществляется асинхронно, т.е. между запросом и соответствующим ответом могут возникать, например, широковещательные сообщения.

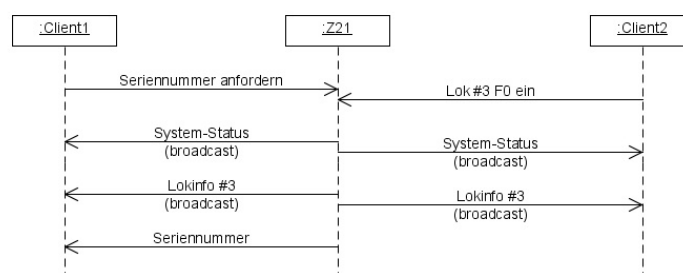


Рисунок 1 Пример связи последовательностей

Ожидается, что каждый клиент будет связываться с Z21 один раз в минуту, иначе он будет удален из списка активных участников. Если возможно, клиент должен выйти из системы с помощью команды LAN\_LOGOFF в центральном офисе при выходе.

## 1.2 Z21 Набор данных

### 1.2.1 Конструкция

Запись Z21, то есть запрос или ответ, построена следующим образом:

**DataLen (2 Byte)      Header (2 Byte)      Data (n Bytes)**

- **Порядок байтов (порядок байтов):**  
общая длина по всему набору данных, включая данные, заголовки и данные, т.е. данные =  $2+2+n$ .
- **Заголовок (прямой порядок байтов):**

- описывает команду и порядок байтов соответственно. группа протоколов.  
**Данные:**  
Состав и количество зависят от команды. Точное описание см. В соответствующей команде.

Если не указано иное, порядок байтов имеет порядок байтов с прямым порядком байтов, то есть сначала нижний байт, а затем верхний байт.



## Спецификация протокола локальной сети Z21

### 1.2.2 Туннелирование протокола X-BUS

Заголовок Z21 LAN **0x40 (LAN\_X\_XXX)** используется для передачи запросов и ответов, которые основаны на протоколе X-BUS. При этом имеется в виду только протокол, поскольку эти команды не имеют ничего общего с физической шиной X Z21, а предназначены исключительно для клиентов локальной сети и, соответственно, для других систем. Z21 направлен.

Затем фактическая команда X-BUS находится в поле **Data** в наборе данных Z21. Последний байт является контрольной суммой и рассчитывается как XOR с помощью команды X-BUS. Пример:

Данные	Заголовок		Data		DB0	DB1	XOR-Byte
0x08	0x00	<b>0x40</b>	0x00	<b>h</b>	<b>x</b>	<b>y</b>	h XOR x XOR y

### 1.2.3 Туннелирование LocoNet

**Начиная с Z21 FW версии 1.20.**

Используя заголовок LAN Z21 **0xA0 и 0xA1 (LAN\_LOCONET\_Z21\_RX, LAN\_LOCONET\_Z21\_TX)**, сообщения, полученные и отправленные Z21 на шине LocoNet соответственно, перенаправляются клиенту локальной сети. Для этого клиент локальной сети должен быть подписан на сообщения LocoNet с помощью

**2.16 LAN\_SET\_BROADCASTFLAGS.**

Используя заголовок LAN Z21 **0xA2 (LAN\_LOCONET\_FROM\_LAN)**, клиент локальной сети может записывать сообщения на шину LocoNet.

Это позволяет использовать Z21 в качестве **шлюза Ethernet / LocoNet**, при этом Z21 является одновременно мастером локонета, который управляет слотами обновления и генерирует пакеты DCC.

Фактическое сообщение LocoNet в каждом случае находится в поле **Data** в наборе данных Z21.

Пример сообщения локальной сети OPC\_MOVE\_SLOTS <0><0> („DISPATCH\_GET“) было получено от Z21:

Данные	Заголовок		Data		ARG1	ARG2	CKSUM
0x08	0x00	<b>0xA0</b>	0x00	<b>OPC</b>	<b>0xBA</b>	<b>0x00</b>	<b>0x45</b>

Подробнее о шлюзе LocoNet см. В разделе **9 LocoNet**.



Спецификация протокола локальной сети Z21

### 1.3 Объединение записей в один UDP-пакет

В полезной нагрузке UDP-пакета несколько независимых друг от друга записей Z21 также могут быть отправлены вместе получателю. Каждый получатель должен иметь возможность интерпретировать эти комбинированные UDP-пакеты.

#### Пример

следующего комбинированного UDP-пакета...

#### UDP-пакета

IP Header	UDP Header	Полезная нагрузка UDP		
		<b>Z21 запись 1</b>	<b>Z21 набор данных 2</b>	<b>Z21 запись 3</b>
		LAN_X_GET_TOURNOUT_INFO #4	LAN_X_GET_TOURNOUT_INFO #5	LAN_RMBUS_GETDATA #0

... эквивалентен этим трем UDP-пакетам, отправленным подряд:

#### UDP пакет 1

IP Header	UDP Header	Полезная нагрузка UDP
		<b>Набор данных Z21</b>
		LAN_X_GET_TOURNOUT_INFO #4

#### UDP пакет 2

IP Header	UDP Header	Полезная нагрузка UDP
		<b>Набор данных Z21</b>
		LAN_X_GET_TOURNOUT_INFO # 5

#### UDP пакет 3

IP Header	UDP Header	Полезная нагрузка UDP
		<b>Набор данных Z21</b>
		LAN_RMBUS_GETDATA #0



## 2 Система, состояние, версии

### 2.1 LAN\_GET\_SERIAL\_NUMBER

Считывание серийного номера Z21.

Требование к Z21:

Данные	Заголовок	Data
0x04    0x00	<b>0x10</b> 0x00	-

Ответ от Z21:

Данные	Заголовок	Data
0x08    0x00	<b>0x10</b> 0x00	Серийный номер 32 бит (прямой порядок байтов)

### 2.2 LAN\_LOGOFF

Выход клиента из системы Z21.

Требование к Z21:

Данные	Заголовок	Data
0x04    0x00	<b>0x30</b> 0x00	-

Ответ Z21:

нет

используйте тот же номер порта при выходе из системы, что и при входе в систему.

**Примечание:** вход в систему выполняется неявно с помощью первой команды клиента (например LAN\_SYSTEM\_STATE\_GETDATA, ...).

### 2.3 LAN\_X\_GET\_VERSION

Следующая команда может использоваться для считывания версии X-Bus Z21.

Требование к Z21:

Данные	Заголовок	Data			
0x07    0x00	<b>0x40</b> 0x00	<b>X-Header</b> <b>0x21</b>	<b>DB0</b> <b>0x21</b>	<b>XOR-Byte</b> 0x00	

Ответ от Z21:

Данные	Заголовок	Data	DB0	DB1	DB2	XOR-Byte
0x09    0x00	<b>0x40</b> 0x00	<b>X-Header</b> 0x63	<b>0x21</b>	<b>0x30</b>	<b>0x12</b>	0x60

DB1 ... X-шина версии 3.0

DB2 ... идентификатор центрального блока, 0x12 = Z21

### 2.4 LAN\_X\_GET\_STATUS

С помощью этой команды можно запросить статус центра.

Требование к Z21:

Данные		Заголовок		Data		
				X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x21</b>	<b>0x24</b>	0x05

Ответ Z21:

см. 2.12 LAN\_X\_STATUS\_CHANGED

Это центральное состояние идентично центральному состоянию, которое отображается в системном состоянии, см.

2.18 LAN\_SYSTEMSTATE\_DATACHANGED.

## 2.5 LAN\_X\_SET\_TRACK\_POWER\_OFF

Эта команда отключит натяжение гусеницы.

Требование к Z21:

Данные		Заголовок		Data		
				X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x21</b>	<b>0x80</b>	0xa1

Ответ Z21:

см. 2.7 LAN\_X\_BC\_TRACK\_POWER\_OFF

## 2.6 LAN\_X\_SET\_TRACK\_POWER\_ON

С помощью этой команды напряжение дорожки включается, соответственно, аварийная остановка или режим программирования завершается.

Требование к Z21:

Данные		Заголовок		Data	X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x21</b>		<b>0x81</b>	0xa0

Ответ Z21:

см. 2.8 LAN\_X\_BC\_TRACK\_POWER\_ON



Спецификация протокола локальной сети Z21

## 2.7 LAN\_X\_BC\_TRACK\_POWER\_OFF

Следующий пакет отправляется с Z21 зарегистрированным клиентам, если

- клиент отправил команду 2.5 LAN\_X\_SET\_TRACK\_POWER\_OFF
- с помощью другого устройства ввода (мульти-мышь) напряжение дорожки было отключено.
- рассматриваемый клиент включил соответствующую трансляцию, см. 2.16 LAN\_SET\_BROADCASTFLAGS , Flag 0x00000001

Z21 клиент:

Данные	Заголовок	Data
--------	-----------	------

0x07	0x00	<b>0x40</b>	0x00	<b>X-Header</b> <b>0x61</b>	<b>DB0</b> <b>0x00</b>	<b>XOR-Byte</b> 0x61
------	------	-------------	------	--------------------------------	---------------------------	-------------------------

## 2.8 LAN\_X\_BC\_TRACK\_POWER\_ON

Следующий пакет будет отправлен с Z21 зарегистрированным клиентам, если

- клиенту Z21 LAN\_X\_SET\_TRACK\_POWER\_ON.
- с помощью другого устройства ввода (мульти-мышь) напряжение дорожки было включено.
- рассматриваемый клиент включил соответствующую трансляцию, см. **2.16 LAN\_SET\_BROADCASTFLAGS** , Flag 0x00000001

Z21 клиент:

Данные	Заголовок	Data	X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x61</b>	<b>0x01</b>

## 2.9 LAN\_X\_BC\_PROGRAMMING\_MODE

Следующий пакет будет отправлен Z21 зарегистрированным клиентам, если Z21 был переведен в режим программирования резюме с помощью 6.1 LAN\_X\_CV\_READ или 6.2 LAN\_X\_CV\_WRITE и соответствующий клиент активировал соответствующую широковещательную рассылку, см.

**2.16 LAN\_SET\_BROADCASTFLAGS** , Flag 0x00000001

Z21 клиент:

Данные	Заголовок	Data	X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x61</b>	<b>0x02</b>

## 2.10 LAN\_X\_BC\_TRACK\_SHORT\_CIRCUIT

Следующий пакет будет отправлен с Z21 зарегистрированным клиентам, если произошло короткое замыкание и рассматриваемый клиент включил соответствующую широковещательную рассылку, см.

**2.16 LAN\_SET\_BROADCASTFLAGS** , Flag 0x00000001

Z21 клиент:

Данные	Заголовок	Data	X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x61</b>	<b>0x08</b>



## 2.11 LAN\_X\_UNKNOWN\_COMMAND

Следующий пакет отправляется с Z21 клиенту в ответ на недопустимый запрос.

Z21 клиент:

Данные	Заголовок	Data	X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0x61</b>	<b>0x82</b>

## 2.12 LAN\_X\_STATUS\_CHANGED

Следующий пакет отправляется с Z21 клиенту, если клиент явно запросил состояние с помощью 2.4 LAN\_X\_GET\_STATUS.

Z21 клиент:

Данные	Заголовок	Data	X-Header	DB0	DB1	XOR-Byte
--------	-----------	------	----------	-----	-----	----------

DB1 ... Центральное состояние

```
Битовые маски для центрального состояния:
#define csEmergencyStop          0x01 // Включена аварийная остановка
#define csTrackVoltageOff        0x02 // Напряжение дорожки отключено
#define csShortCircuit           0x04 // Короткое замыкание
#define csProgrammingModeActive 0x20 // Режим программирования активен
```

Это центральное состояние идентично состоянию системы.CentralState, см.  
2.18 LAN\_SYSTEMSTATE\_DATACHANGED.



Спецификация протокола локальной сети Z21

2.13 LAN\_X\_SET\_STOP

С помощью этой команды активируется аварийная остановка, то есть локомотивы останавливаются. но  
путевое напряжение остается включенным.

Требование к Z21:					
Данные		Заголовок		Data	XOR-Byte
				X-Header	
0x06	0x00	0x40	0x00	0x80	0x80

Ответ Z21:  
см. 2.14 LAN\_X\_BC\_STOPPED

2.14 LAN\_X\_BC\_STOPPED

Следующий пакет будет отправлен с Z21 зарегистрированным клиентам, если

- клиент отправил команду 2.13 LAN\_X\_SET\_STOP.
- аварийная остановка была вызвана другим устройством ввода (мульти-мышь).
- рассматриваемый клиент включил соответствующую трансляцию, см. **2.16 LAN\_SET\_BROADCASTFLAGS** , Flag 0x00000001

Z21 клиент:					
Данные		Заголовок		Data	XOR-Byte
				X-Header	DB0
0x07	0x00	0x40	0x00	0x81	0x000x81

## 2.15 LAN\_X\_GET\_FIRMWARE\_VERSION

Эта команда может использоваться для считывания версии прошивки Z21.

Требование к Z21:

Данные		Заголовок		Data X-Header	DB0	XOR-Byte
0x07	0x00	<b>0x40</b>	0x00	<b>0xF1</b>	0x0A	0xFB

Ответ от Z21:

Данные		Заголовок		Data				
				X-Header	DB0	DB1	DB2	XOR-Byte
0x09	0x00	<b>0x40</b>	0x00	0xF3	0x0A	<b>V MSB</b>	<b>V LSB</b>	XOR-байт

**DB1** ... Байт с более высоким значением версии прошивки

**DB2** ... Байт с более низким значением версии прошивки

Версия указана в формате BCD.

Пример:

0x09 0x00 0x40 0x00 0xF3 0x0A **0x01 0x23** 0xdb означает: „Прошивка версии **1.23**“



Спецификация протокола локальной сети Z21

## 2.16 LAN\_SET\_BROADCASTФЛАГИ

Установка флагов вещания в Z21. Эти флаги устанавливаются для каждого клиента (т.е. для каждого IP + номера порта)

и должны быть снова установлены при следующем входе в систему.

Требование к Z21:

Данные	Заголовок	Data
0x08	0x00	<b>0x50</b> 0x00 Broadcast-Flags 32 Bit (little endian)

Broadcast Flags - это OR-соединение следующих значений:

0x00000001 Автоматически генерируемые трансляции и сообщения, касающиеся вождения и переключения передач, доставляются зарегистрированному клиенту.

Здесь подписаны следующие сообщения:

**2.7** LAN\_X\_BC\_TRACK\_POWER\_OFF

**2.8** LAN\_X\_BC\_TRACK\_POWER\_ON

**2.9** РЕЖИМ

**ПРОГРАММИРОВАНИЯ 2.10** LAN\_X\_BC\_TRACK\_SHORT\_CIRCUIT

**2.14** LAN\_X\_BC\_STOPPED

**4.4** LAN\_X\_LOCO\_INFO (рассматриваемый локальный адрес также должен быть подписан)

**5.3** LAN\_X\_TURNOUT\_INFO

0x00000002 Изменения датчиков обратной связи на шине R отправляются автоматически.

Широковещательное сообщение Z21 см. **B 7.1** LAN\_RMBUS\_DATACHANGED

0x00000004 Изменения в данных RailCom отправляются автоматически (будущее расширение)

0x00000100 Изменения состояния системы Z21 отправляются автоматически.

Широковещательное сообщение Z21 см. **B 2.18** LAN\_SYSTEMSTATE\_DATACHANGED

### Ab Z21 FW Version 1.20:

0x00010000 Дополняет флаг 0x00000001; клиент теперь подписывается на LAN\_X\_LOCO\_INFO без предварительной подписки на соответствующие адреса локомотивов, т.е. для всех управляемых локомотивов! Этот флаг разрешен только полноценным пользователям из-за высокого сетевого трафика. Контроллеры ПК используются и ни в коем случае не предназначены для мобильных ручных контроллеров. V1.20 до V1.23: LAN\_X\_LOCO\_INFO отправляется для **всех** локомотивов. Начиная с **FW V1.24**: LAN\_X\_LOCO\_INFO будет отправлен для **всех измененных** локомотивов.

0x01000000 Пересылка сообщений с шины LocoNet на клиент LAN без каких-либо указателей и указателей.

0x02000000 Пересылать специфичные для Lokсообщения локонета клиенту локальной сети: OPC\_LOCO\_SPD, OPC\_LOCO\_DIRF, OPC\_LOCO\_SND, OPC\_LOCO\_F912,

0x04000000 OPC\_EXP\_CMD  
Пересылка сообщений LocoNet, специфичных для коммутатора, клиенту локальной сети:  
OPC\_SW\_REQ, OPC\_SW\_REP, OPC\_SW\_ACK, OPC\_SW\_STATE  
Смотрите также главу 9 LocoNet.

### Ab Z21 FW Version 1.22:

0x08000000 Отправлять сообщения о состоянии с детекторов загруженности путей на шине LocoNet клиенту  
Локальной сети.

Документ версии 1.04

25.03.2014

15/48



Спецификация протокола локальной сети Z21

### LAN\_LOCONET\_DETECTOR

Ответ Z21:  
нет

При настройке флагов вещания также учитывайте влияние на загрузку сети. Это особенно верно для флагов вещания 0x00010000, 0x02000000 и 0x04000000! IP-пакеты могут быть удалены маршрутизатором в случае перегрузки, а UDP не предоставляет никаких механизмов обнаружения для этого! Например, с флагом 0x00000100 (состояние системы) стоит подумать, не является ли 0x00000001 более разумной альтернативой с соответствующими широковещательными сообщениями LAN\_X\_BC\_xxx. Потому что не каждое приложение всегда должно быть в курсе самых последних значений напряжения, тока и температуры, поступающих из центрального устройства, вплоть до мельчайших подробностей.

### 2.17 LAN\_GET\_BROADCASTФЛАГИ

Считывание флагов трансляции в Z21.

Требование к Z21:

Данные	Заголовок	Data
0x04 0x00	0x51 0x00	-

Ответ от Z21:

Данные	Заголовок	Data
0x08 0x00	0x51 0x00	Broadcast-Flags 32 Bit (little endian)

Флаги трансляции см. Выше.

### 2.18 LAN\_SYSTEMSTATE\_DATACHANGED

Сообщить клиенту об изменении состояния системы с Z21.

Это сообщение будет асинхронно сообщаться Z21 клиенту, если последний

- включил соответствующую трансляцию, см. 2.16 LAN\_SET\_BROADCASTFLAGS , флаг 0x00000100
- сделал явный запрос о состоянии системы, см. 2.19 LAN\_SYSTEMSTATE\_GETDATA ниже.

Z21 клиент:

Данные	Заголовок	Data
0x14 0x00	0x84 0x00	SystemState (16 Bytes)

SystemState построен следующим образом (16-битные значения имеют порядок байтов):

Byte	Offset	Название типа		
0	INT16	MainCurrent	mA	Ток на магистральном пути
2	INT16	Постоянный		ток mA на программном треке

4	INT16	Отфильтрованная токовая мощность	Внутренний ток на магистральном пути
6	INT16	Температура	°C внутренняя температура в центральном
офисе 8	UINT16	Напряжение питания мВ	Напряжение питания
	Внутреннее		напряжение, как напряжение на гусеничном ходу
12	UINT8		Битовую маску центрального состояния см. Ниже
13	UINT8		Битовую маску CentralStateEx см. Ниже
14	UINT8	reserved	
15	UINT8	reserved	

Документ версии 1.04

25.03.2014

16/48



## Спецификация протокола локальной сети Z21

### Битовые маски для CentralState:

```
#define csEmergencyStop      0x01    // Включена аварийная остановка
#define csTrackVoltageOff    0x02    // Напряжение дорожки отключено
#define csShortCircuit       0x04    // Короткое замыкание
#define csProgrammingModeActive 0x20  // Режим программирования активен
```

### Битовые маски для CentralStateEx:

```
#define cseHighTemperature   0x01    // слишком высокая температура
#define csePowerLost         0x02    // слишком низкое входное напряжение
#define cseShortCircuitExternal 0x04  // на выходе внешнего усилителя
#define cseShortCircuitInternal 0x08  // на главном пути или пути программирования
```



Спецификация протокола локальной сети Z21

## 2.19 LAN\_SYSTEMSTATE\_GETDATA

Запрос текущего состояния системы.

Требование к Z21:

Данные	Заголовок	Data
0x04 0x00	0x85 0x00	-

Ответ от Z21:

См. Выше 2.18 LAN\_SYSTEMSTATE\_DATACHANGED

## 2.20 LAN\_GET\_HWINFO

Начиная с Z21 FW версии 1.20 и SmartRail FW версии V1.13.

Эта команда может использоваться для считывания типа оборудования и версии прошивки Z21.

Требование к Z21:

Данные	Заголовок	Data
0x04 0x00	0x1A 0x00	-

Ответ от Z21:

Данные	Заголовок	Data
0x0C 0x00	0x1A 0x00	HwType 32 Bit (little endian)      FW Version 32 Bit (little endian)

**HwType:**

```
#define D_HWT_Z21_OLD      0x00000200    // Z21 (Hardware-Variante ab 2012)
#define D_HWT_Z21_NEW      0x00000201    // Z21 (аппаратный вариант с 2013 года)
#define D_HWT_SMARTRAIL    0x00000202    // SmartRail (с 2012 г.)
#define D_HWT_z21_SMALL    0x00000203    // вариант стартового набора z21 (с 2013 г.)
```

Версия **FW** указана в формате BCD.

Пример:

**0x0C 0x00 0x1A 0x00 0x00 0x02 0x00 0x00 0x00 0x20 0x01 0x00 0x00**

означает: „Аппаратное оборудование версии прошивки 1.20“

В качестве альтернативы, чтобы прочесть версию более старой прошивки, используйте команду **2.15 LAN\_X\_GET\_FIRMWARE\_VERSION** . Это относится к более старым версиям прошивки:

- Версия 1.10 ... Z21 (аппаратный вариант с 2012 года)
- V1.11 ... Z21 (аппаратный вариант с 2012 года)
- B1.12 ... SmartRail (с 2012 г.)



## 3 Настройки

Следующие настройки, описанные здесь, будут постоянно сохраняться в Z21.

Эти настройки могут быть сброшены пользователем до заводских настроек, удерживая нажатой кнопку STOP на Z21. будет до тех пор, пока светодиоды не начнут мигать фиолетовым.

### 3.1 LAN\_GET\_LOCOMODE

Чтение выходного формата для заданного локального адреса.

В Z21 выходной формат (DCC, MM) для каждого локального адреса может быть сохранен постоянно. Можно разместить максимум 256 различных локальных адресов. Любой адрес  $\geq 256$  автоматически становится DCC.

Требование к Z21:

Данные	Заголовок	Data
0x06 0x00	0x60 0x00	Локальный адрес 16 бит ( <b>прямой порядок байтов</b> )

Ответ от Z21:

Данные	Заголовок	Data
0x07 0x00	0x60 0x00	Локальный адрес 16 бит ( <b>прямой порядок байтов</b> )8

-битный адрес ~~16 бит~~ **прямой порядок байтов** с прямым порядком байтов, т.е. сначала верхний байт, затем нижний байт.

Режим 0 ... Формат DCC  
Формат 1 ... MM

### 3.2 LAN\_SET\_LOCOMODE

Установка выходного формата для заданного локального адреса. Формат постоянно сохраняется в Z21 .

Требование к Z21:

Данные	Заголовок	Data
0x07 0x00	0x61 0x00	Локальный адрес 16 бит ( <b>прямой порядок байтов</b> )8 бит

Ответ Z21:

значения, указанные выше, не имеют значения.

**Примечание:** каждый локальный адрес  $\geq 256$  автоматически является и остается „Форматом DCC“.

**Примечание:** уровни вождения (14, 28, 128) также постоянно сохраняются в центральном офисе. Это происходит автоматически по команде драйва, см. **4.2 LAN\_X\_SET\_LOCO\_DRIVE**.

### 3.3 LAN\_GET\_TURNOUTMODE

Чтение настроек для данного адреса функционального декодера („функциональный декодер“ в значении

„Вспомогательный декодер“ RP-9.2.1).

В Z21 выходной формат (DCC, MM) для каждого адреса функционального декодера может быть сохранен постоянно. Он может хранить максимум 256 различных адресов декодера функций. Любой адрес  $\geq 256$  автоматически становится DCC.

Требование к Z21:

Данные	Заголовок	Data
0x06 0x00	0x70 0x00	Адрес декодера функции 16 бит ( <b>прямой порядок байтов</b> )

Ответ от Z21:

Данные	Заголовок	Data
0x07 0x00	0x70 0x00	Адрес декодера функции 16 бит ( <b>порядок байтов</b> ) Режим 8-битный

адрес декодера функции 2 байта, **порядок байтов** с прямым порядком байтов, т.е. сначала верхний байт, затем нижний байт.

Режим 0 ... Формат DCC  
Формат 1 ... MM

В интерфейсе локальной сети и в Z21 адреса декодера функций адресуются начиная с 0, но в визуализации в приложениях или на мультимаусе они адресуются начиная с 1. Это просто является решением визуализации. Пример: мульти-мышь указывает адрес # 3, соответствующий адресу 2 в локальной сети и Z21.

### 3.4 LAN\_SET\_TURNOUTMODE

Установка выходного формата для заданного адреса декодера функции. Формат постоянно сохраняется в Z21.

Требование к Z21:

Данные	Заголовок	Data
0x07 0x00	0x71 0x00	Адрес декодера функции 16 бит ( <b>порядок байтов</b> ) Режим 8 бит

Ответ Z21:

значения, указанные выше, не имеют значения.

Декодеры функций MM поддерживаются прошивкой Z21, начиная с версии прошивки 1.20.

Декодеры функций MM не поддерживаются SmartRail.

**Примечание:** любой адрес декодера функции  $\geq 256$  автоматически является и остается „Форматом DCC“.



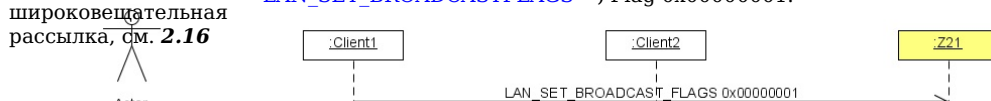
Спецификация протокола локальной сети Z21

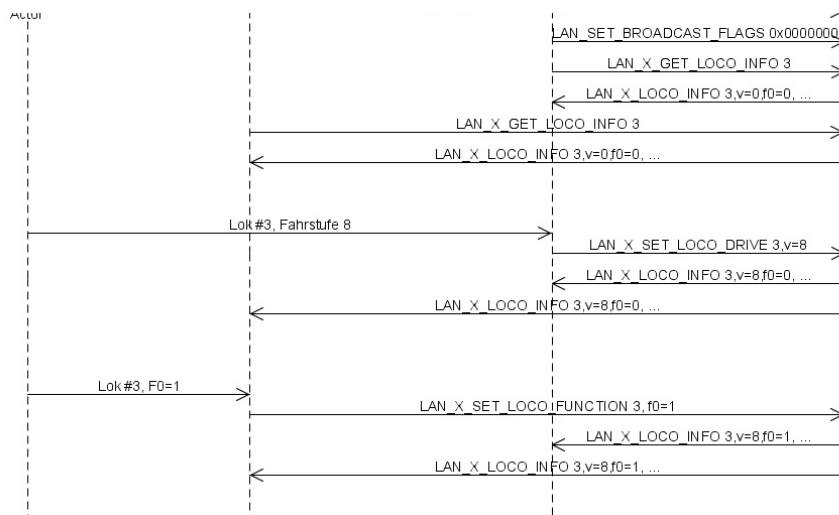
## 4 Вождение

В этой главе рассматриваются сообщения, необходимые для работы с декодерами локомотивов

.

Клиент может подписаться на Lok Info с помощью 4.1 LAN\_X\_GET\_LOCO\_INFO, чтобы автоматически получать уведомления о любых будущих изменениях этого локального адреса, вызванных другими клиентами или ручными регуляторами. Кроме того, для клиента также должна быть включена соответствующая широковещательная рассылка, см. **2.16** LAN\_SET\_BROADCAST\_FLAGS, flag 0x00000001.





**Рисунок 2 Пример последовательности управления локомотивом**

Чтобы поддерживать сетевой трафик в разумных пределах, можно подписаться максимум на 16 локальных адресов для каждого клиента (FIFO). Хотя он не возражает против продолжения „пыльцы“ после этого, но это следует делать только с учетом использования сети: IP-пакеты могут быть удалены маршрутизатором в случае перегрузки, а UDP не предоставляет никаких механизмов обнаружения для этого!

#### 4.1 LAN\_X\_GET\_LOCO\_INFO

Статус локомотива может быть запрошен с помощью следующей команды. В то же время он использует его для „подписки“ на информацию о местоположении для этого локального адреса клиентом.

Требование к Z21:

Данные	Заголовок	Data	X-Header	DB0	DB1	DB2	XOR-Byte
0x09	0x00	<b>0x40</b>	0x00	<b>0xE3</b>	<b>0xF0</b>	<b>Adr_MSB</b>	<b>Adr_LSB</b>
							XOR-байт

Применяется: адрес локомотива = (**Adr\_MSB** и 0x3F) << 8 + **Adr\_LSB**

С адресами локомотива два высших бита в DB1 должны быть установлены равными 1:

**DB1 = (0xC0 | Adr\_MSB)**. Для локальных адресов < 128 эти два высших бита не имеют значения.

Ответ Z21:

см. 4.4 LAN\_X\_LOCO\_INFO



#### 4.2 LAN\_X\_SET\_LOCO\_DRIVE

С помощью следующей команды можно переключить одну функцию декодера Lok.

Требование к Z21:

Данные	Заголовок	Data	X-Header	DB0	DB1	DB2	DB3	XOR-Byte
0x0A	0x00	<b>0x40</b>	0x00	<b>0xE4</b>	<b>0x1S</b>	<b>Adr_MSB</b>	<b>Adr_LSB</b>	<b>RVVVVVVV</b>
								XOR-байт

Применяется: адрес локомотива = (**Adr\_MSB** и 0x3F) << 8 + **Adr\_LSB**

С адресами локомотива два высших бита в DB1 должны быть установлены равными 1:

**DB1 = (0xC0 | Adr\_MSB)**. Для локальных адресов < 128 эти два высших бита не имеют значения.

**0x1S** S=0 или 1: 14 уровней вождения  
S=2: 28 уровней вождения  
S= 3: 128 ступеней движения

**PBBBBBBBB** ... Направление: 1=вперед

V ... Скорость: в зависимости от уровней движения, кодировка, как в DCC

Ответ от Z21:  
нет ответа по умолчанию, 4.4 LAN\_X\_LOCO\_INFO для клиентов с подпиской.

**Примечание:** уровни вождения автоматически сохраняются на постоянной основе в центральном офисе.

### 4.3 LAN\_X\_SET\_LOCO\_FUNCTION

С помощью следующей команды можно переключить одну функцию декодера Lok.

Требование к Z21:

Данные	Заголовок	Data						
		X-Header	DB0	DB1	DB2	DB3	XOR-Byte	
0x0A	0x00	0x40	0x00	0xE4	0xF8	Adr_MSB	Adr_LSB	TTNNNNN XOR-Byte

Применяется: адрес локомотива = (Adr\_MSB и 0x3F) << 8 + Adr\_LSB

С адресами локомотива 128 два высших бита в DB1 должны быть установлены равными 1:

**DB1 = (0xC0 | Adr\_MSB).** Для локальных адресов < 128 эти два высших бита не имеют значения.

**Тип переключения:** 00=выкл., 01= вкл., 10= переключение, 11=не разрешено

**NNNNNN** Индекс функции, 0 = F0 (свет), 1 = F1 и т. Д.

Ответ от Z21:

нет ответа по умолчанию, 4.4 LAN\_X\_LOCO\_INFO для клиентов с подпиской.



Спецификация протокола локальной сети Z21

### 4.4 LAN\_X\_LOCO\_INFO

Это сообщение отправляется Z21 клиентам в ответ на команду

4.1 LAN\_X\_GET\_LOCO\_INFO. Но он также отправляется клиентам без запроса, если

- состояние Lok было изменено одним из клиентов или ручными регуляторами
- рассматриваемый клиент включил соответствующую трансляцию, см. **2.16 LAN\_SET\_BROADCASTFLAGS** , Flag 0x00000001
- и рассматриваемый клиент подписался на локальный адрес с помощью 4.1 LAN\_X\_GET\_LOCO\_INFO

Z21 клиент:

Данные	Заголовок	Data						
		X-Header	DB0	...	...	DBn	XOR-Byte	
7 + n	0x00	0x40	0x00	0xEF	Информация о локомотиве			XOR-байт

Текущая длина пакета может варьироваться в зависимости от фактических отправленных данных

Данные для **получения информации о локомотиве** построены следующим образом:

Положение	Данные	Важность
DB0	Adr_MSB	Два старших бита в Adr_MSB следует игнорировать.
DB1	Adr_LSB	Адрес локомотива = (Adr_MSB и 0x3F) << 8 + Adr_LSB
DB2	0000BKKK	B= 1 ... локомотив управляется („занят“) другим устройством (ручным регулятором X-BUS).
		KKK ... Информация о лифте: 0=14, 2=28, 4=128
DB3	PBBBBBBB	P ... Направление: 1=вперед
		V ... Скорость: в зависимости от уровней движения, кодирование, как в DCC
DB4	0DSLFGHJ	D ... Двойная тяга: 1= Локомотив, включенный в двойную тягу.
		S ... Smartsearch
		L ... F0 (легкий)
		F ... F4
		G ... F3

		<b>H ... F2</b>
		<b>J ... F1</b>
DB5	F5-F12	Функция F5 равна биту 0 (LSB)
DB6	F13-F20	Функция F13 равна биту 0 (LSB)
DB7	F21-F28	Функция F21 равна биту 0 (LSB)
DBn		необязательно, для будущих расширений



Спецификация протокола локальной сети Z21

## 5 Переключение передач

В этой главе рассматриваются сообщения, необходимые для переключения функциональных декодеров в значении

„Вспомогательный декодер“ RP-9.2.1(т.е. программный декодер, ...).

Визуализация номера указателя на пользовательском

интерфейсе решается по-разному во многих системах DCC и может значительно отличаться от фактического адреса

и порта декодера доступа, используемых на дорожке. Согласно DCC, на адрес декодера доступа приходится четыре порта с двумя

выходами в каждом. На порт может быть подключен один переключатель. Обычно для визуализации

номера указателя на дорожке используется комбинация следующих портов декодера (ESU, Uhlenbrock, ...)

Мягкий #1: DCC Addr=1 порт=0; Мягкий #5: DCC Addr=2 порта=0; Мягкий #6: DCC Addr=2 порта=1

2. Нумерация от 1 с адресом DCC, начинающимся с 0, по 4 порта в каждом (**Roco**, Lenz).

Мягкий #1: DCC Addr=0 порт=0; Мягкий #5: DCC Addr=1 порт=0; Мягкий #6: DCC Addr=1 порт=1

3. Виртуальный номер указателя со свободно настраиваемым адресом и портом DCC (двойной центр).

4. Представление адреса / порта DCC (Zimo)

Ни один из этих способов визуализации нельзя назвать „неправильным“. Однако пользователю необходимо привыкнуть

к тому, что один и тот же переключатель управляется на центральной панели ESU под номером 1

, в то время как на мультимаусе Roco с Z21 он переключается под номером 5

(„сдвиг на 4“).

Чтобы иметь возможность реализовать в вашем приложении визуализацию по вашему выбору, полезно знать, как

Z21 преобразует входные параметры для команд переключения (**FAdr\_MSB**, **FAdr\_LSB**, **A**, **P**, см. Ниже) в соответствующую команду DCC Accessory:

DCC Basic Accessory Decoder Packet Format: {preamble} 0 10AAAAAA 0 1aaaCDDd 0 EEEEEEEE 1

UINT16 FAdr = (**FAdr\_MSB** << 8) + **FAdr\_LSB**;

UINT16 Dcc\_Addr = FAdr >> 2;

AAAAAA = (~Dcc\_Addr & 0x1C0) | (Dcc\_Addr & 0x003F); // DCC адрес

C = **A**; // Включение или отключение выхода

DD = FAdr & 0x03; // Порт

d = **P**; // Отклонитесь влево или вправо

Пример:  
FAdr=0 дает адрес DCC=0 порт=0;  
FAdr=3 дает адрес DCC=0 порт=3;  
FAdr=4 дает адрес DCC=1 порт=0; и т. д  
С другой стороны, для формата MM: FAdr начинается с 0, т.е. FAdr=0: MM-Addr=1; FAdr=1: MM-Addr=2; ...

Клиент может подписаться на информацию о функциях, чтобы автоматически получать уведомления об изменениях в декодерах функций, также вызванных другими клиентами или ручными регуляторами.

Для этого у клиента должна быть включена соответствующая трансляция, см. **2.16**

LAN\_SET\_BROADCASTFLAGS,

Платформенное положение указателя зависит от проводки и, возможно, также от конфигурации в клиентском приложении. Центральный офис не может знать об этом, поэтому в приведенном ниже описании намеренно опущены обозначения „прямой“ и „ответвление“.

Документ версии 1.04

25.03.2014

24/48



Спецификация протокола локальной сети Z21

## 5.1 LAN\_X\_GET\_TURNOUT\_INFO

С помощью следующей команды можно изменить состояние мягкого (или мягкого) переключателя соответственно. функция переключения) может быть запрошена.

Требование к Z21:

Данные	Заголовок	Data	DB0	DB1	XOR-Byte
0x08	0x00	<b>0x40</b>	<b>0x00</b>	<b>0x43</b>	<b>FAdr_MSB</b>
					<b>FAdr_LSB</b>
					XOR-байт

Применяется: адрес функции = (FAdr\_MSB << 8) + FAdr\_LSB

Ответ Z21:

см. 5.3 LAN\_X\_TURNOUT\_INFO

## 5.2 LAN\_X\_SET\_TURNOUT

Переключатель можно переключить с помощью следующей команды.

Требование к Z21:

Данные	Заголовок	Data	DB0	DB1	DB2	XOR-Byte
0x09	0x00	<b>0x40</b>	<b>0x00</b>	<b>0x53</b>	<b>FAdr_MSB</b>	<b>FAdr_LSB</b>
					<b>10Q0A00P</b>	XOR-байт

Es gilt: Funktions-Adresse = (FAdr\_MSB << 8) + FAdr\_LSB

**1000A00P** A=0 ... Отключение выхода стрелочного перевода  
A=1 ... Активация выхода стрелочного переводчика  
P=0 ... Выберите выход 1 из мягкого переключателя  
P=1 ... Выберите выход 2 переключателя  
Q=0 ... немедленно выполните команду  
Q= 1 ... **начиная с Z21 FW V1.24:** вставьте команду стрелочного перевода в очередь в Z21 и выведите ее на дорожку в следующий возможный момент.

Ответ Z21:

нет ответа по умолчанию , 5.3 LAN\_X\_TURNOUT\_INFO для клиентов с подпиской.

**Начиная с Z21 FW V1.24,** был введен флаг Q („Очередь“).

### 5.2.1 LAN\_X\_SET\_TURNOUT с Q=0

Если **Q=0**, то Z21 ведет себя совместимо с предыдущими версиями: команда поворота немедленно выводится на трассу, смешиваясь с текущими командами движения.  
**Активация (A=1) будет выдаваться до тех пор, пока соответствующий деактиватор не будет отправлен клиентом локальной сети. Одновременно может быть активна только одна команда переключения передач.** Такое поведение аналогично, например, нажатию и отпусанию кнопок мультимуса.

Обратите внимание, что при  $Q=0$  необходимо строго соблюдать правильную последовательность команд переключения (т.е. активировать , за которым следует деактивировать). В противном случае это может привести к неопределенным конечным положениям в зависимости от используемого вами декодера наложения курсора. **Правильная сериализация и выбор времени переключения являются обязанностью клиента локальной сети!**



## Спецификация протокола локальной сети Z21

### **Неправильно:**

Включить софт #5/A2 (4,0x89); Включить софт #6/A2 (5,0x89);  
Включить софт #3/A1 (2,0x88); Отключить софт #3/A1 (2,0x80);  
Отключить софт #5/A2 (4,0x81); Отключить софт #6/A2 (5,0x81);

### **Правильно:**

Включить софт #5/A2 (4,0x89); ожидание 100 мс; Отключить софт #5/A2 (4,0x81); ожидание 50 мс;  
Включить софт #6/A2 (5,0x89); ожидание 100 мс; Отключить софт #6/A2 (5,0x81); 50 мс ожидание;  
Включить софт #3/A1 (2,0x88); ждать 100 мс; Отключить софт #3/A1 (2,0x80); ждать 50 мс;

### Игра с топором:

включить софт # 7 / A2 (6,0x89); подождать 150 мс; Отключить софт # 7 / A2 (6,0x81)

**Рисунок 3 DCC нюхает дорожку при  $Q = 0$**

### 5.2.2 LAN\_X\_SET\_TURNOUT с Q= 1

Если **Q = 1**, то это приводит к следующему поведению: команда переключения сначала помещается в Z21 во внутреннюю очередь (FIFO). При генерации сигнала дорожки эта очередь постоянно проверяется на наличие команды переключения для вывода. Эта команда переключения затем, при необходимости, извлекается из очереди и выдается четыре раза на треке. Это освобождает клиента локальной сети от ранее обязательной сериализации, то есть команды переключения могут быть отправлены на Z21 в смешанном виде при Q = 1 (проезжие части!). Клиенту локальной сети просто нужно больше заботиться о сроках деактивации. В зависимости от вашего DCC-декодера, деактивация может быть даже исключена при определенных обстоятельствах. Но ни в коем случае

Пример: не следует отказываться от MM, потому что, например, k83 и более старые приводы стрелочных переводов не имеют софта #25 / A2 (24, 0xA9); Включить софт #5 / A2 (4, 0xA9);  
подождать 150 мс;  
Отключить софт #25 / A2 (24, 0xA1)

Рисунок 4 DCC нюхает дорожку при Q = 1

Ни в коем случае не смешивайте в своем приложении команды переключения с Q=0 и команды переключения с Q=1.



5.3 LAN\_X\_TURNOUT\_INFO

Это сообщение отправляется Z21 клиентам в ответ на команду LAN\_X\_GET\_TURNOUT\_INFO. Но он также отправляется клиентам без запроса, если

- состояние функции было изменено одним из клиентов или ручным регулятором;
- и соответствующий клиент включил соответствующую широковещательную рассылку, см. 2.16 LAN\_SET\_BROADCASTFLAGS , Flag 0x00000001

Z21 клиент:								
Данные		Заголовок		Data		XOR-Byte		
0x09	0x00	0x40	0x00	0x43	FAdr_MSB	FAdr_LSB	000000ZZ	XOR-байт

Применяется: адрес функции = (FAdr\_MSB << 8) + FAdr\_LSB

000000ZZ      ZZ=00 ... Софт еще не переключен  
 ZZ=01 ... Стрелочный перевод выполняется в соответствии с командой переключения „P = 0“,  
 ZZ=10 LAN\_X\_SET\_TURNOUT выполняется в соответствии с командой переключения „P = 1“,  
 ZZ=11 LAN\_X\_SET\_TURNOUT выполняется в соответствии с командой переключения „P = 1“,

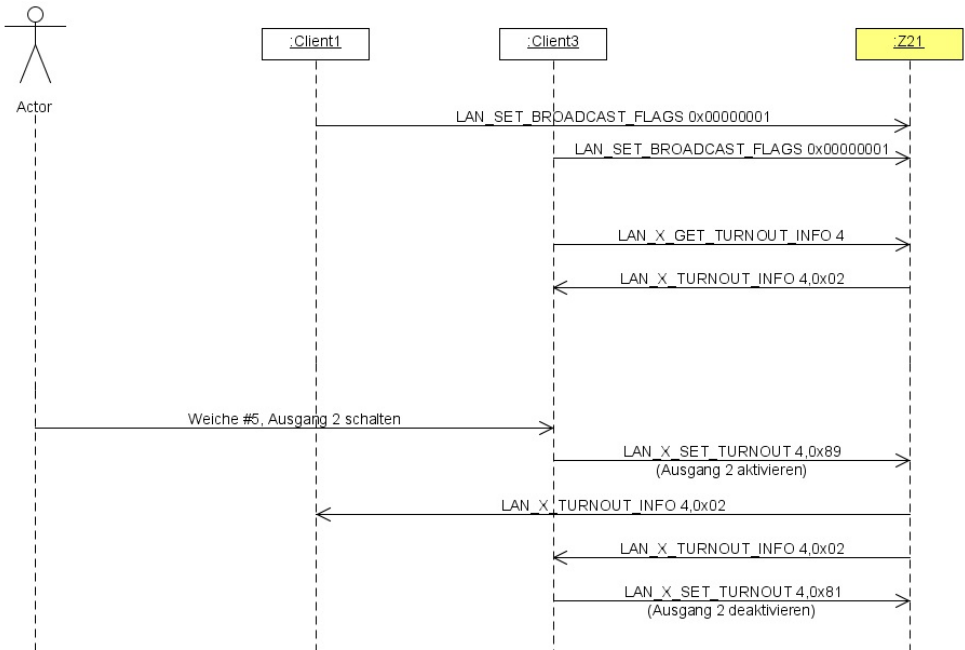


Рисунок 5 Пример последовательности мягкого переключения передач



6 Декодер для чтения и записи резюме

В этой главе рассматриваются сообщения, необходимые для чтения и записи РЕЗЮМЕ декодера (переменная конфигурации, RP-9.2.2, RP-9.2.3).

Происходит ли доступ к декодеру побитовым или побайтовым способом, зависит от настроек в Z21.

### 6.1 LAN\_X\_CV\_READ

Резюме может быть прочитано в режиме Direct с помощью следующей команды

Требование к Z21:

Данные	Заголовок		Data					XOR-Byte
			X-Header	DB0	DB1	DB2		
0x09	0x00	0x40	0x00	0x23	0x11	CVAdr_MSB	CVAdr_LSB	XOR-Byte

Es gilt: CV-Adresse = (CVAdr\_MSB << 8) + CVAdr\_LSB, sowie 0=CV1., 1=CV2, 255=CV256, usw.

Ответ от Z21:

2.9 LAN\_X\_BC\_PROGRAMMING\_MODE для клиентов с подпиской, а также результат  
6.3 LAN\_X\_CV\_NACK\_SC, 6.4 LAN\_X\_CV\_NACK или 6.5 LAN\_X\_CV\_RESULT.

### 6.2 LAN\_X\_CV\_WRITE

Используя следующую команду, РЕЗЮМЕ может быть перезаписано в режиме Direct.

Требование к Z21:

Данные	Заголовок		Data					XOR-Byte
			X-Header	DB0	DB1	DB2	DB3	
0x0A	0x00	0x40	0x00	0x24	0x12	CVAdr_MSB	CVAdr_LSB	Value
								XOR-байт

Es gilt: CV-Adresse = (CVAdr\_MSB << 8) + CVAdr\_LSB, sowie 0=CV1., 1=CV2, 255=CV256, usw.

Ответ от Z21:

2.9 LAN\_X\_BC\_PROGRAMMING\_MODE для клиентов с подпиской, а также результат  
6.3 LAN\_X\_CV\_NACK\_SC, 6.4 LAN\_X\_CV\_NACK или 6.5 LAN\_X\_CV\_RESULT.

### 6.3 LAN\_X\_CV\_NACK\_SC

Если программирование было выполнено с ошибкой из-за короткого замыкания на гусенице, это сообщение автоматически отправляется клиенту, который инициировал программирование с помощью 6.1

LAN\_X\_CV\_READ или  
6.2 LAN\_X\_CV\_WRITE.

Z21 клиент:

Данные	Заголовок		Data			XOR-Byte
			X-Header	DB0		
0x07	0x00	0x40	0x00	0x61	0x12	0x73

Документ версии 1.04

25.03.2014

29/48



Спецификация протокола локальной сети Z21

### 6.4 LAN\_X\_CV\_NACK

Если ACK не попадает в декодер, это сообщение автоматически отправляется клиенту, который инициировал программирование через 6.1 LAN\_X\_CV\_READ или 6.2 LAN\_X\_CV\_WRITE.

При байтовом доступе при чтении время до LAN\_X\_CV\_NACK может занять очень много времени.

Z21 клиент:

Данные	Заголовок		Data			XOR-Byte
			X-Header	DB0		
0x07	0x00	0x40	0x00	0x61	0x13	0x72

### 6.5 LAN\_X\_CV\_RESULT

Это сообщение одновременно является „положительным подтверждением“ и автоматически отправляется клиенту, который инициировал программирование через 6.1 LAN\_X\_CV\_READ или 6.2 LAN\_X\_CV\_WRITE.  
 При байтовом доступе при чтении время до LAN\_X\_CV\_RESULT может занять очень много времени.  
 Z21 клиент:

Данные	Заголовок		Data						XOR-Byte
0x0A	0x00	0x40	0x00	X-Header	DB0	DB1	DB2	DB3	
0x0A	0x00	0x40	0x00	0x64	0x14	CVAdr_MSB	CVAdr_LSB	Value	XOR-байт

Es gilt: CV-Adresse = (CVAdr\_MSB << 8) + CVAdr\_LSB, sowie 0=CV1., 1=CV2, 255=CV256, usw.

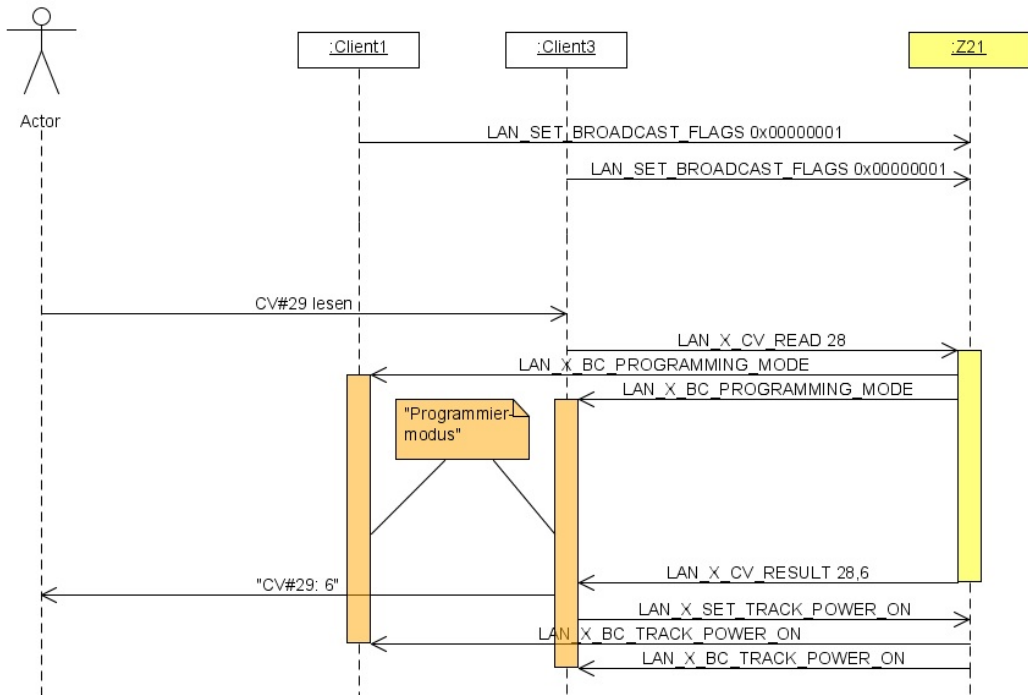


Рисунок 6 Чтения резюме последовательности



Спецификация протокола локальной сети Z21

## 6.6 LAN\_X\_CV\_POM\_WRITE\_BYTE

Используя следующую команду, CV локального декодера (многофункционального цифрового декодера в NMRA S-9.2.1 Abschnitt C; Configuration Variable Access Instruction - Long Form) на главном треке (POM „Программирование на главном“). Это происходит в обычном режиме работы, т. е. напряжение дорожки должно быть включено, обычный режим программирования не включен. Обратной связи нет.

Требование к Z21:

Данные	Заголовок		Data						XOR-Byte
0x0C	0x00	0x40	0x00	X-Header	DB0	DB1	DB2	DB3	
0x0C	0x00	0x40	0x00	0xE6	0x30	Параметры POM			XOR-байт

Данные для параметров POM построены следующим образом:

Положение	Данные	Важность
DB1	Adr_MSB	
DB2	Adr_LSB	Адрес локомотива = (Adr_MSB и 0x3F) << 8 + Adr_LSB
DB3	111011MM	Option ... 0xEC MM ... CVAdr_MSB
DB4	CVAdr_LSB	CV-Adresse = (MM << 8) + CVAdr_LSB (0=CV1., 1=CV2, 255=CV256, usw.)
DB5	Value	neuer CV-Wert

Ответ Z21:  
нет

## 6.7 LAN\_X\_CV\_POM\_WRITE\_BIT

Следующая команда позволяет записать бит **РЗКОМ** в локального декодера (многофункциональные цифровые декодеры в соответствии с разделом С NMRA S-9.2.1; Инструкция по доступу к переменной конфигурации - длинная форма) путь (POM). Это делается в обычном режиме работы, т.е. напряжение на пути должно быть включено, обычный режим программирования не включен. Обратной связи нет

Требование к Z21:

Данные	Заголовок	Data									
0x0C	0x00	0x40	0x00	X-Header	DB0	DB1	DB2	DB3	DB4	DB5	XOR-Byte
		0xE6		0x30	Параметры POM						XOR-байт

Данные для **параметров POM** построены следующим образом:

Положение	Данные	Важность
DB1	Adr_MSB	
DB2	Adr_LSB	Адрес локомотива = (Adr_MSB и 0x3F) << 8 + Adr_LSB
DB3	111010MM	Option ... 0xE8 MM ... CVAdr_MSB
DB4	CVAdr_LSB	CV-Adresse = (MM << 8) + CVAdr_LSB (0=CV1., 1=CV2, 255=CV256, usw.)
DB5	0000VPPP	PPP ... Bit-Position in CV V ... neuer Bit-Wert

Ответ Z21:  
нет



Спецификация протокола локальной сети Z21

## 6.8 LAN\_X\_CV\_POM\_READ\_BYTE

Начиная с Z21 FW версии 1.22.

Используя следующую команду, резюме локального декодера (многофункционального цифрового декодера в соответствии с разделом С NMRA S-9.2.1; Инструкция по доступу к переменной конфигурации - длинная форма) может быть прочитано на главном пути (POM). Это происходит в обычном режиме работы, т.е. напряжение дорожки должно быть включено, обычный режим программирования не включен. RailCom должен быть включен в Z21 . Считываемый автомобильный декодер должен знать RailCom, CV28 бит 0 и 1, а также CV29 бит 3 . Требования к Z21 обновлены на 1 в локальном декодере (Zimo).

Данные	Заголовок	Data									
0x0C	0x00	0x40	0x00	X-Header	DB0	DB1	DB2	DB3	DB4	DB5	XOR-Byte
		0xE6		0x30	Параметры POM						XOR-байт

Данные для **параметров POM** построены следующим образом:

Положение	Данные	Важность
DB1	Adr_MSB	
DB2	Adr_LSB	Адрес локомотива = (Adr_MSB и 0x3F) << 8 + Adr_LSB
DB3	111010MM	Option ... 0xE4 MM ... CVAdr_MSB
DB4	CVAdr_LSB	CV-Adresse = (MM << 8) + CVAdr_LSB (0=CV1., 1=CV2, 255=CV256, usw.)
DB5	0	neuer CV-Wert

Ответ от Z21:

6.4 LAN\_X\_CV\_NACK или 6.5 LAN\_X\_CV\_RESULT .



Спецификация протокола локальной сети Z21

## 6.9 LAN\_X\_CV\_POM\_ACCESSORY\_WRITE\_BYTE

Начиная с Z21 FW версии 1.22.

Используя следующую команду, РЕЗЮМЕ вспомогательного декодера (согласно разделу D NMRA S-9.2.1, Базовый адрес пакета вспомогательного декодера для программирования режима работы) может быть записано на основной канал (POM). Это происходит в обычном режиме работы, т.е. напряжение дорожки должно быть включено, обычный режим программирования не включен. Обратной связи нет.

Требование к Z21:

Данные	Заголовок	Data							
		X-Header	DB0	DB1	DB2	DB3	DB4	DB5	XOR-Byte
0x0C	0x00	0x40	0x00	0xE6	0x31	Параметры POM			XOR-байт

Данные для параметров POM построены следующим образом:

Положение	Данные	Важность
DB1	aaaaa	Decoder Adresse MSB
DB2	AAAACDDD	Применяется: $aaaaaAAAACDDD = ((\text{адрес декодера} \& 0x1FF) \ll 4)   CDDD$ ; Если CDDD=0000, то РЕЗЮМЕ относится ко всему декодеру. Если C=1, то DDD - это номер выходного сигнала, который необходимо
DB3	111011MM	Оформить. MM ... CVAdr_MSB
DB4	CVAdr_LSB	CV-Adresse = $(MM \ll 8) + CVAdr\_LSB$ (0=CV1., 1=CV2, 255=CV256, usw.)
DB5	Value	neuer CV-Wert

Ответ Z21:

нет

## 6.10 LAN\_X\_CV\_POM\_ACCESSORY\_WRITE\_BIT

Начиная с Z21 FW версии 1.22.

Используя следующую команду, бит CV вспомогательного декодера (в соответствии с разделом D NMRA S-9.2.1, Базовый адрес пакета вспомогательного декодера для программирования режима работы) может быть записан на основной канал (POM). Это происходит в обычном режиме работы, т.е. напряжение дорожки должно быть включено, обычный режим программирования не включен. Обратной связи нет

Требование к Z21:

Данные	Заголовок	Data							
		X-Header	DB0	DB1	DB2	DB3	DB4	DB5	XOR-Byte

0x0C	0x00	<b>0x40</b>	0x00	<b>0xE6</b>	<b>0x31</b>	<b>Параметры POM</b>	XOR-байт
Данные для <b>параметров POM</b> построены следующим образом:							
<b>Положение</b>	<b>Данные</b>	<b>Важность</b>					
DB1	<b>aaaaa</b>	Decoder_Adresse MSB					
DB2	<b>AAAACDDD</b>	Применяется: <b>aaaaaAAAACDDD</b> = ((адрес декодера & 0x1FF) << 4)   CDDD; Если <b>CDDD</b> =0000, то РЕЗЮМЕ относится ко всему декодеру. Если <b>C</b> =1, то <b>DDD</b> - это номер выходного сигнала, который необходимо					
DB3	<b>111010MM</b>	<b>Option ... 0xE4</b> программировать. <b>MM</b> ... CVAdr_MSB					
DB4	<b>CVAdr_LSB</b>	CV-Adresse = ( <b>MM</b> << 8) + <b>CVAdr_LSB</b> (0=CV1., 1=CV2, 255=CV256, usw.)					
DB5	<b>0000VPPP</b>	<b>PPP</b> ... Bit-Position in CV <b>V</b> ... neuer Bit-Wert					

Dokumentenversion 1.04

25.03.2014

33/48



Спецификация протокола локальной сети Z21

Ответ Z21:  
нет

## 6.11 LAN\_X\_CV\_POM\_ACCESSORY\_READ\_BYTE

**Начиная с Z21 FW версии 1.22.**

Используя следующую команду, РЕЗЮМЕ вспомогательного декодера (в соответствии с разделом D NMRA S-9.2.1,

Адрес пакета базового вспомогательного декодера для программирования режима работы) может быть прочитано на главном канале

с помощью POM). Это происходит в обычном режиме работы, т.е. напряжение дорожки должно быть включено, обычный режим программирования не включен. RailCom должен быть включен в Z21 Требуется декодер Z21, который нужно прочитать, должен быть опытным в RailCom.

Данные		Заголовок		Data						XOR-Byte	
				X-Header	DB0	DB1	DB2	DB3	DB4	DB5	XOR-Byte
0x0C	0x00	0x40	0x00	0xE6	0x31	Параметры POM					XOR-байт

Данные для **параметров POM** построены следующим образом:

Положение	Данные	Важность
DB1	<b>aaaaa</b>	Decoder_Adresse MSB
DB2	<b>AAAACDDD</b>	Применяется: <b>aaaaaAAAACDDD</b> = ((адрес декодера & 0x1FF) << 4)   CDDD; Если <b>CDDD</b> =0000, то РЕЗЮМЕ относится ко всему декодеру. Если <b>C</b> =1, то <b>DDD</b> - это номер рассматриваемого вывода.
DB3	<b>111010MM</b>	<b>Option ... 0xE4</b> <b>MM</b> ... CVAdr_MSB
DB4	<b>CVAdr_LSB</b>	CV-Adresse = ( <b>MM</b> << 8) + <b>CVAdr_LSB</b> (0=CV1., 1=CV2, 255=CV256, usw.)
DB5	<b>0</b>	neuer CV-Wert

Ответ от Z21:  
6.4 LAN\_X\_CV\_NACK или 6.5 LAN\_X\_CV\_RESULT .



Спецификация протокола локальной сети Z21

## 6.12 LAN\_X\_MM\_WRITE\_BYTE

Начиная с Z21 FW версии 1.23.

Следующая команда может быть использована для перезаписи регистра декодера Motorola на треке программирования.

Требование к Z21:

Данные	Заголовок	Data					
		X-Header	DB0	DB1	DB2	DB3	XOR-Byte
0x0A 0x00	0x40 0x00	0x24	0xFF 0		RegAdr	Value	XOR-байт

Это верно для **RegAdr**: 0=Register1, 1=Register2, ..., 78=Register79.

Es ~~применяется~~ **Value** < 255, но некоторые декодеры принимают значения только от 0 до 80.

Ответ от Z21:

2.9 LAN\_X\_BC\_PROGRAMMING\_MODE для клиентов с подпиской, а также результат

6.3 LAN\_X\_CV\_NACK\_SC или 6.5 LAN\_X\_CV\_RESULT .

**Примечание:** Программирование декодеров Motorola не было предусмотрено в исходном формате Motorola . Следовательно, не существует стандартизированной и обязательной процедуры программирования для программирования декодеров Motorola. Для программирования декодеров Motorola в Z21 был реализован введенный позже так называемый „режим программирования 6021“. Это позволяет записывать значения, но не позволяет считывать их. Точно так же невозможно проверить успешность операции записи (за исключением обнаружения короткого замыкания). Эта процедура программирования работает для многих декодеров ESU, Zimo и Märklin, но не является обязательной для всех декодеров MM. Например , декодеры Motorola не могут быть запрограммированы с помощью DIP-переключателей. Некоторые декодеры принимают в программировании Motorola от декодера не поступает никаких отзывов об успешности операции записи, только сообщение LAN\_X\_CV\_RESULT (см. Описание декодера). Поэтому LAN\_X\_CV\_RESULT следует понимать просто как „MM операция программирования завершена“, а не как „MM операция программирования выполнена успешно“.

Пример:

**0x0A 0x00 0x40 0x00 0x24 0xFF 0x00 0x00 0x05 0xDE**

означает: „Измените адрес локального декодера (регистр 51)

## 7 Шина обратной связи R

Модули обратной связи (номер заказа 10787) на шине R можно считывать и настраивать с помощью следующих команд.

### 7.1 LAN\_RMBUS\_DATACHANGED

Сообщить об изменении в шине обратной связи с Z21 клиенту.

Это сообщение будет асинхронно сообщаться Z21 клиенту, если последний

- включил соответствующую трансляцию, см. **2.16** [LAN\\_SET\\_BROADCASTFLAGS](#) , флаг 0x00000002
- или явно запросил статус обратной связи, см. Ниже 7.2 LAN\_RMBUS\_GETDATA.

Z21 клиент:

Данные	Заголовок	Data
0x0F 0x00	<b>0x80</b> 0x00	<b>Gruppenindex</b> (1 Byte) <b>Статус обратной связи</b> (10 байт)

**Индекс группы:** 0 ... Модули обратной связи с адресами от 1 до 10  
1 . Модули обратной связи с адресами от 11 до 20

**состояний обратной связи:** 1 байт на обратную связь, 1 бит на вход.  
Отображение адреса обратной связи и позиции байта статически возрастающее.

Например:

индекс группы = 1 и статус обратной связи = 0x01 0x00 0xC5 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00  
означает „Обратная связь 11, контакт на входе 1; Обратная связь 13, контакт на входе 8,7,3 и 1“

### 7.2 LAN\_RMBUS\_GETDATA

Запрос текущего статуса обратной связи.

Требование к Z21:

Данные	Заголовок	Data
0x05 0x00	<b>0x81</b> 0x00	<b>Gruppenindex</b> (1 Byte)

**Групповой индекс:** см. Выше

Ответ от Z21:

См. Выше 7.1 LAN\_RMBUS\_DATACHANGED



### 7.3 LAN\_RMBUS\_ПРОГРАММНЫЕ МОДУЛИ

Изменение адреса обратной связи.

Требование к Z21:

Данные	Заголовок	Data
0x05	0x00	<b>0x82</b> 0x00    Адрес (1 байт)

**Адрес:** новый адрес для программируемого модуля обратной связи.  
Поддерживаемый диапазон значений: 0 и 1 ... 20.

Ответ Z21:

нет

Команда программирования будет выдаваться на шине R до тех пор, пока эта команда не будет повторно отправлена на Z21 с адресом= 0.

В процессе программирования на шине R не должно быть другого модуля обратной связи.

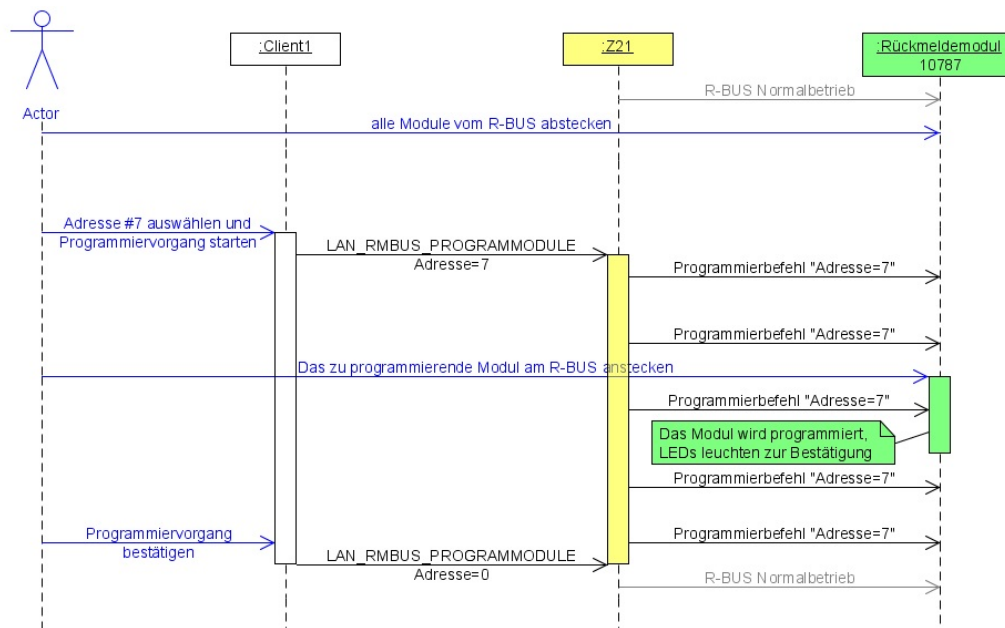


Рисунок 7 Пример Программирования модуля обратной связи последовательности

## 8 RailCom

Поскольку стандартизация RailCom постоянно развивается, с нашей стороны ведется работа над более гибкой передачей данных. Это расширение протокола Z21 LAN в настоящее время находится в стадии разработки и, как ожидается, будет реализовано в виде новых дополнительных команд связи в

новой версии прошивки, а также в свое время будет описано в новой версии „  
Спецификации Z21 LAN“ на данном этапе команд вы все равно можете один раз  
ознакомиться с темой RailCom.

Обратите внимание, что сначала декодер должен быть готов к работе с RailCom, что само собой  
разумеется, а также CV28 и CV29 должны быть правильно настроены (см.  
Руководство по декодированию от производителя). Наконец, конечно, в настройках Z21 все еще должна быть  
включена опция „RailCom“

### 8.1 LAN\_RAILCOM\_DATACHANGED

Это сообщение отправляется Z21 клиенту, который явно запросил данные RailCom  
, см. Ниже 8.2 LAN\_RAILCOM\_GETDATA.

Z21 клиент:

Данные	Заголовок	Data
len	0x00	0x88 0x00 массив railcomdata[n]

**Данные:** Длина **данных len** зависит от количества обнаруженных RailCom декодеры.  
См. Примечание ниже.

**n:** Количество обнаруженных декодеров RailCom;

Структура **Railcomdata** построена следующим образом (16-битное и 32-битное значения имеют порядок

Byte Offset	Имя типа	
0	UINT16 LocoAddress	Адрес обнаруженного декодера
2	UINT32 ReceiveCounter	Счетчик приема в Z21
6	UINT32 ErrorCounter	Счетчик ошибок приема в Z21
10	UINT8 Reserved1	экспериментально, см. Примечание
11	UINT8 Reserved2	экспериментально, см. Примечание
12	UINT8 Reserved3	экспериментально, см. Примечание

**Примечание:** Это относится к **примечанию**

- применяется **len**; и  $len = 4 + (n * 13)$ , а также  $n = (len - 4) / 13$
- *Reserved1* ... Скорость передачи данных RailCom (идентификатор типа сообщения 3 „скорость / нагрузка“, не каждый декодер должен уметь это делать)
- *Reserved2*... Options (experimentell)  
Битовые маски для Options:  
#define rcoSpeed 0x01 // „Скорость“ Railcom была определена декодером mind. отправленный  
опция 0x02
- *Reserved3* ... Температура данных RailCom (идентификатор типа сообщения 8 „Температура“, не каждый декодер должен уметь это делать)



### 8.2 LAN\_RAILCOM\_GETDATA

Запрос данных RailCom от Z21.

Требование к Z21:

Данные	Заголовок	Data
0x04 0x00	0x89 0x00	-

Ответ от Z21:

См. Выше 8.2 LAN\_RAILCOM\_DATACHANGED



Спецификация протокола локальной сети Z21

## 9 LocoNet

**Начиная с Z21 FW версии 1.20.**

Как упоминалось ранее во введении, Z21 можно использовать в качестве **мюза Ethernet** / LocoNet, при этом Z21 одновременно является мастером локонета, который управляет слотами обновления и генерирует пакеты DCC.

Чтобы клиент локальной сети получал сообщения из локальной сети, он должен был подписаться на соответствующие сообщения локальной сети с помощью **2.16** LAN\_SET\_BROADCASTFLAGS. Сообщения, которые Z21 получает на шине LocoNet, передаются клиенту LAN с использованием заголовка **LAN\_LOCONET\_Z21\_RX**.

Сообщения, которые сам Z21 записывает на шину LocoNet, также *передаются клиенту LAN с заголовком* LAN\_LOCONET\_Z21\_TX.

С помощью команды LAN Z21 LAN LAN\_LOCONET\_FROM\_LAN клиент локальной сети может сам записывать сообщения на шину LocoNet. Если одновременно с этим появятся другие клиенты локальной сети с подпиской на LocoNet, они также будут уведомлены сообщением LAN\_LOCONET\_FROM\_LAN. Только фактический отправитель больше не будет получать уведомления в процессе.





Обратите внимание, что эта функциональность шлюза Ethernet/LoCoNet была создана в первую очередь для контроллеров ПК в качестве средства связи с датчиками обратной связи LoCoNet и т. Д.

40/48

**Roco**

**Начиная с Z21 FW версии 1.20.**

- включил соответствующую трансляцию, см. **2.16** LAN\_SET\_BROADCASTFLAGS, флаги 0x01000000, 0x02000000 и 0x04000000 соответственно.
- и от Z21 было получено сообщение на шине LocoNet.

Данные		Заголовок		Data
0x04+n	0x00	0xA0	0x00	Сообщение LocoNet, включая CKSUM n байтов

**Начиная с Z21 FW версии 1.20.**

- включил соответствующую трансляцию, см. **2.16** LAN\_SET\_BROADCASTFLAGS, флаги 0x01000000, 0x02000000 и 0x04000000 соответственно.
- и от Z21 было отправлено сообщение на шину LocoNet.

Данные	Заголовок	Data
--------	-----------	------

9.3 LAN\_LOCONET\_FROM\_LAN

Начиная с Z21 FW версии 1.20.

Это сообщение позволяет клиенту локальной сети записать сообщение на шину LocoNet.

- Это сообщение также будет асинхронно сообщаться Z21 клиенту, если у последнего
- включена соответствующая трансляция, см. **2.16** LAN\_SET\_BROADCASTFLAGS, флаги 0x01000000, 0x02000000 и 0x04000000 соответственно.
  - и **другой** клиент локальной сети, через который Z21 написал сообщение на шину LocoNet.

Клиент локальной сети на Z21, соответственно. Z21 для клиента локальной сети:

Данные	Заголовок	Data
0x04+n	0x00	0xA2    0x00    Сообщение LocoNet, включая CKSUM n байтов



Спецификация протокола локальной сети Z21

9.4 LAN\_LOCONET\_DISPATCH\_ADDR

Начиная с Z21 FW версии 1.20.

Подготовить локальный адрес для отправки в локонет.

Это сообщение позволяет клиенту локальной сети подготовить определенный локальный адрес для отправки в локонет. Это эквивалентно „DISPATCH\_PUT“ и означает, что при следующем „DISPATCH\_GET“ (запускаемом ручными регуляторами) с Z21 слот, принадлежащий этому адресу локомотива, будет сообщен обратно. При необходимости Z21 автоматически займет для этого свободный слот.

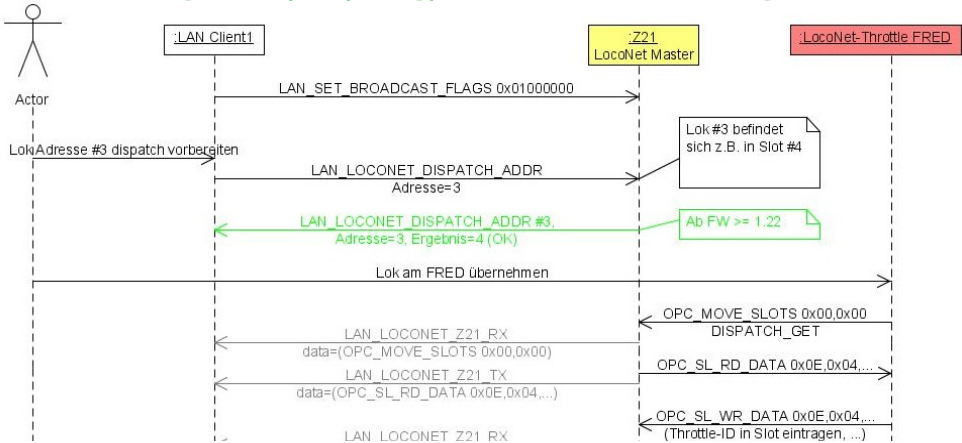
Требование к Z21:

Данные	Заголовок	Data
0x06	0x00	0xA3    0x00    Lok-Adresse 16 bit (little endian)

Ответ от Z21:  
Z21 FW версия < 1.22: нет  
в ответе Z21 FW ≥

Z21 клиент:			
Данные	Заголовок	Data	Результат 8 бит
0x07	0x00	0xA3    0x00    Lok-Adresse 16 bit (little endian)	

- результат    0    Не удалось выполнить „DISPATCH\_PUT“ для данного адреса. Это может произойти, если, например, Z21 работает как ведомое устройство LocoNet, а мастер LocoNet отклонил запрос на отправку, потому что этот локальный адрес уже назначен другому ручному контроллеру.
- >0    „DISPATCH\_PUT“ выполнен успешно. Адрес локомотива теперь может быть записан на ручном контроллере (например, FRED). Значение Result равно текущему номеру слота LocoNet для данного адреса Lok.



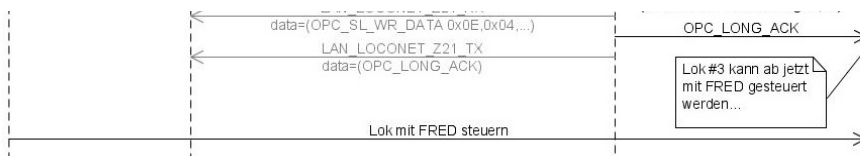


Рисунок 9 Пример 1 Sequenz LocoNet Dispatch per LAN-Client



## Спецификация протокола локальной сети Z21

### 9.5 LAN\_LOCONET\_DETECTOR

Начиная с Z21 FW версии 1.22.

Если приложение в клиенте локальной сети хочет поддерживать детектор заполнения пути LocoNet, для этого есть два варианта. Первым было бы использование [LAN\\_LOCONET\\_Z21\\_RX](#) для получения пакетов LocoNet и самостоятельной обработки соответствующих сообщений LocoNet. Но это требует соответствующего точного знания протокола LocoNet.

Поэтому была создана следующая альтернатива, которая позволяет как клиенту локальной сети запрашивать статус занятого, так и получать асинхронное уведомление об изменении статуса занятого без необходимости углубляться в глубины протокола LocoNet.

**Информация:** пожалуйста, обратите внимание на следующее существенное различие между модулем обратной связи Roco R-шина обратной связи LocoNet, детекторы рельсовых путей: 10787 на шине R (см. 7

- 10787 основан на переключающих контактах с механическим приводом, которые могут замыкаться и размыкаться на каждую ось поезда, идущего по нему.
- Детекторы заполнения пути LocoNet обычно основаны на точном измерении тока на контролируемом участке пути или на передовых технологиях (транспондеры, инфракрасный, RailCom, ..), чтобы иметь возможность надежно определять состояние заполнения пути. В идеале при нормальной работе генерируется только одно сообщение об изменении состояния занятости.

С помощью следующей команды можно запросить состояние одного или нескольких датчиков заполнения пути.

Требование к Z21:

Данные	Заголовок	Data
0x07 0x00	0xA4 0x00	Typ 8 bit Reportadresse 16 bit (little endian)

Запрос типа 0x80 с использованием „Стационарного запроса с поддержкой (SIC)“ процедурой Digitrax. Этот процесс также можно использовать с датчиками квитанций в электронике Блюхера. Адрес отчета здесь равен 0 (не беспокойтесь).

**0x81** Запрос с помощью так называемого **адреса отчета** для детекторов занятости Уленброка. Этот адрес отчета может быть настроен пользователем, например, в UB63320 через LNCV 17 в детекторе занятости. Значение по умолчанию там равно 1017. Адрес отчета используется только для запросов типа 0x81, и его не следует путать с адресом обратной связи.  
**Примечание:** На шине LocoNet этот запрос реализован с помощью команд указателя, поэтому значение **должно быть передано с 0x07-0x00 0xA4 0x00 0x81 0xF8 0x03** (адрес отчета = 1017 = **0x03F8 + 1 = 1016 + 1**)“

**0x82** Запрос статуса для LISSY от Z21 FW версии 1.23  
У Уленброка ЛИСИИ, однако здесь адрес отчета снова совпадает с адресом обратной связи. Характер последующей обратной связи во многом зависит от настроенного режима работы приемника LISSY. Вы можете ознакомиться с обширными возможностями настройки приемника LISSY в руководстве по эксплуатации LISSY.

Обратите внимание, что при выполнении одного запроса может быть одновременно



## Спецификация протокола локальной сети Z21

Ответ от Z21:

Z21 клиент:

Данные	Заголовок	Data	Информация[n]
0x07 + <i>n</i>	0x00 0x00 0x00 0x00	Тип 8-битный адрес обратной связи 16-битный (прямой порядок байтов)	

Это сообщение будет асинхронно сообщаться Z21 клиенту, если последний

- включил соответствующую трансляцию, см. **2.16** LAN\_SET\_BROADCASTFLAGS, Flag 0x08000000
- и Z21 получил соответствующее сообщение от детектора занятости пути из-за изменения состояния на его входе или из-за явного запроса со стороны клиента локальной сети с помощью команд, описанных выше.

**Адрес обратной связи** Каждому входу детектора

занятости присваивается свой собственный адрес обратной связи, который может быть настроен пользователем (например, у Уленброка и Блюхера с помощью LNCV) и четко описывает контролируемый блок.

**Информация[n]** массив байтов; содержимое и длина *n* зависят от типа, см. Ниже

**Тип** **0x01** Для типов датчиков занятости, таких как Ulenbrock 63320 или Blucher GBM16XL, которые сообщают только о состоянии „занято“ и „свободно“ (LocoNet OPC\_INPUT\_REP, X=1). *n*=1

Статус входного сигнала, принадлежащего адресу обратной связи, указан в **Info[0]**:  
**Info[0]=0** ... Датчик **ЛО** (“свободный”)  
**Информация[0]=1** ... Датчик **ПРИВЕТ** (“занят”)

**0x02 Transponder Enters Block**

**0x03 Transponder Exits Block**

Для детекторов занятости, таких как Blucher GBM16XN и т. Д., Которые сообщают информацию

(например, адрес местонахождения) об автомобиле в блоке в центральный офис (с помощью OPC\_MULTI\_SENSE) передает так называемый

адрес транспондера в блоке. Адрес транспондера идентифицирует

транспортное средство, находящееся в блоке. В случае GBM16XN это локомотивный адрес, который

определен датчиком квитанций с помощью RailCom.

Адрес транспондера находится в **Info[0]** и **Info[1]**, 16-битный порядок байтов с прямым порядком байтов:

**Info[0]** ... Transponderadresse Low Byte

**Info[1]** ... Transponderadresse High Byte

Примечание: из-за слабости спецификации LocoNet существует возможность

интерпретации диапазона значений OPC\_MULTI\_SENSE, что оставляет производителей детекторов квитанций в неведении.. Следовательно, в случае GBM16XN, исходя из нашего опыта, следует отметить следующее:

- К адресу обратной связи необходимо добавить +1, чтобы перейти на тот адрес обратной связи, который настроен в GBM16XN.
- В зависимости от конфигурации GBM16XN в бите под маской 0x1000 будет закодировано направление движения транспортного средства по рельсам. Мы не рекомендуем эту конфигурацию, поскольку этот бит сталкивается с адресным пространством для длинных локальных адресов!

**0x10 Адрес местонахождения Лисси с Z21 по FW 1.23.**

Это сообщение отправляется на клиент Z21 LAN, когда Уленброк Приемник LISSY сообщает о транспортном средстве, оборудованном передатчиком LISSY , и переводит приемник LISSY в „ÜF (формат передачи) Уленброк“ (LNCV 15) настроен. Кроме того, это сообщение во многом зависит от настроенного режима работы (LNCV2, ...) приемника Lissy. См. Руководство ЛИССИ.

**n= 3**

Адрес местоположения указан в Info[0] и Info[1], 16-битный порядок байтов с прямым порядком байтов:

**Info[0]** ... Lokadresse Low Byte

**Info[1]** ... Локомотивы с большим байтовым

адресом имеют диапазон значений 1 ..9999

вагонов имеют диапазон значений от 10000 до 16382

**Информация[2]** ... Дополнительная информация со следующими битами: 0 **ДИР1** **ДИР0** 0 **К3** **К2** **К1** **К0**

**ДИР1=0:** **ДИР0** игнорировать

**ДИР1=1:** **ДИР0=0** вперед, **ДИР0=1** назад

**К3..К0:** 4 бит Информация о классе, которая была депонирована в передатчике ЛИССИ.

**0x11 Состояние квитанции ЛИССИ, начиная с Z21 FW 1.23.**

Это сообщение отправляется на клиент Z21 LAN, когда Уленброк Приемник ЛИССИ настроен как детектор квитанций. См. Руководство ЛИССИ.

**n=1**

Состояние блока, принадлежащего адресу обратной связи, указано в Info[0]:

**Info[0]=0** ... Блок свободен

**Информация[0]=1** ... Блок занят

**0x12 Скорость Лисси от Z21 до FW 1.23.**

Это сообщение отправляется на клиент Z21 LAN, когда Уленброк Приемник Лисси настроен для измерения скорости. См. Руководство ЛИССИ.

**n=2**

Скорость указана в Info[0] и Info[1], 16-битный прямой порядок байтов:

**Информация[0]** ... Скорость Низкий байт

**Информация[1]** . Скорость Высокий байт

Прим. **тип** в будущем будет расширен дополнительными идентификаторами в зависимости от потребностей.



# Приложение А - Обзор команд

## Client an Z21

Header Parameter			Фамилия
X-Header DB0 Parameter			
0x10	-		LAN_GET_SERIAL_NUMBER
0x1A	-		LAN_GET_HWINFO
0x30	-		LAN_LOGOFF
0x40	0x21	0x21 -	LAN_X_GET_VERSION
0x40	0x21	0x24 -	LAN_X_GET_STATUS
0x40	0x21	0x80 -	LAN_X_SET_TRACK_POWER_OFF
0x40	0x21	0x81 -	LAN_X_SET_TRACK_POWER_ON
0x40	0x23	0x11 CV-Adresse	LAN_X_CV_READ
0x40	0x24	0x12 CV-Adresse, Wert	LAN_X_CV_WRITE
0x40	0x24	Регистр 0xFF, значение	LAN_X_MM_WRITE_BYTE
0x40	0x43	Адрес пункта назначения	LAN_X_GET_TURNOUT_INFO
0x40	0x53	адрес стрелочного перевода, команда	LAN_X_SET_TURNOUT
0x40	0x80	-	LAN_X_SET_STOP
0x40	0xE3	0xF0 Lok-Adresse	LAN_X_GET_LOCO_INFO
0x40	0xE4	0x1s локомотивный адрес, скорость	LAN_X_SET_LOCO_DRIVE
0x40	0xE4	0xF8 локомотивный адрес, функция	LAN_X_SET_LOCO_FUNCTION
0x40	0xE6	0x30 POM-Param, Option 0xEC	LAN_X_CV_POM_WRITE_BYTE
0x40	0xE6	0x30 POM-Param, Option 0xE8	LAN_X_CV_POM_WRITE_BIT
0x40	0xE6	0x30 POM-Param, Option 0xE4	LAN_X_CV_POM_READ_BYTE
0x40	0xE6	0x31 POM-Param, Option 0xEC	LAN_X_CV_POM_ACCESSORY_WRITE_BYTE
0x40	0xE6	0x31 POM-Param, Option 0xE8	LAN_X_CV_POM_ACCESSORY_WRITE_BIT
0x40	0xE6	0x31 POM-Param, Option 0xE4	LAN_X_CV_POM_ACCESSORY_READ_BYTE
0x40	0xF1	0x0A -	LAN_X_GET_FIRMWARE_VERSION
0x50	Флаги трансляции		LAN_SET_BROADCASTFLAGS
0x51	-		LAN_GET_BROADCASTФЛАГИ
0x60	Адрес локомотива		LAN_GET_LOCOMODE
0x61	Адрес локомотива, режим		LAN_SET_LOCOMODE
0x70	Адрес декодера функции		LAN_GET_TURNOUTMODE
0x71	Адрес декодера функции, режим		LAN_SET_TURNOUTMODE
0x81	Групповой индекс		LAN_RMBUS_GETDATA
0x82	Адрес		LAN_RMBUS_ПРОГРАММНЫЕ
МОДУЛИ 0x85			LAN_SYSTEMSTATE_GETDATA
0x89	-		LAN_RAILCOM_GETDATA
0xA2	Сообщение LocoNet		LAN_LOCONET_FROM_LAN
0xA3	Адрес локомотива		LAN_LOCONET_DISPATCH_ADDR
0xA4	тип, адрес отчета		LAN_LOCONET_DETECTOR

Таблица 1 Сообщения от клиента на Z21



## Z21 an Client

Данные заголовка			Фамилия
X-Header DB0			
0x10	Серийный номер		Ответ на LAN_GET_SERIAL_NUMBER
0x1A	HWType, FW Version (BCD)		Ответ на LAN_GET_HWINFO
0x40	0x43	Информация о стрелках	LAN_X_TURNOUT_INFO
0x40	0x61	0x00 -	LAN_X_BC_TRACK_POWER_OFF
0x40	0x61	0x01 -	LAN_X_BC_TRACK_POWER_ON

0x40	0x61	0x08	-	LAN_X_BC_PROGRAMMING_MODE
0x40	0x61	0x12	-	LAN_X_CV_NACK_SC
0x40	0x61	0x13	-	LAN_X_CV_NACK
0x40	0x61	0x82	-	LAN_X_UNKNOWN_COMMAND
0x40	0x62	0x22	Статус	LAN_X_STATUS_CHANGED
0x40	0x63	0x21	Версия XBus, идентификатор	LAN_X_GET_VERSION
0x40	0x64	0x14	РЕЗЮМЕ-Результат	LAN_X_CV_RESULT
0x40	0x81	-		LAN_X_BC_STOPPED
0x40	0xEF		Информация о локомотиве	LAN_X_LOCO_INFO
0x40	0xF3	0x0A	Версия (BCD)	Ответ на LAN_X_GET_FIRMWARE_VERSION
0x51			Флаги трансляции	Ответ на LAN_GET_BROADCASTFLAGS
0x60			Адрес локомотива, режим	Ответ на LAN_GET_LOCOMODE
0x70			Адрес декодера функции	Ответ на LAN_GET_TURNOUTMODE
0x80			индекс группы, статус обратной связи	LAN_RMBUS_DATACHANGED
0x84			Состояние системы	LAN_SYSTEMSTATE_DATACHANGED
0x88			Железнодорожные данные[n]	LAN_RAILCOM_DATACHANGED
0xA0			Сообщение LocoNet	LAN_LOCONET_Z21_RX
0xA1			Сообщение LocoNet	LAN_LOCONET_Z21_TX
0xA2			Сообщение LocoNet	LAN_LOCONET_FROM_LAN
0xA3			Адрес локомотива, результат	LAN_LOCONET_DISPATCH_ADDR
0xA4			Тип, адрес обратной связи, информация	LAN_LOCONET_DETECTOR

**Таблица 2 Сообщения от Z21 клиентам**



## Каталог иллюстраций

Рисунок 1 Пример связи последовательностей .....	7
Рисунок 2 Пример последовательности управления локомотивом	
Рисунок 3.DCC.Нюхает дорожку.в.точке.Q=0.....	21
Рисунок 4 DCC Нюхает дорожку в точке Q=1 .....	27
Рисунок 5 Пример последовательности мягких переключение	
Рисунок 6.Пример.Чтения.резюме.последовательности	28
Рисунок 7.Пример.Программирования.модуля.обратной.связи.последовательности	
Рисунок 8.Пример.последовательности.шлюз Ethernet/LocoNet .....	40
Рисунок 9 Пример Последовательности LocoNet Dispatch для каждого клиента локальной сети .....	42

# Табличный каталог

Таблица 1 Сообщения от клиента на Z21 .....	46
Таблица 2 Сообщения от Z21 клиентам .....	47