

«Синхроком»

Система протоколов взаимодействия по сети Ethernet

Описание протокола

Синхроком-Дата (SCD)

Версия: 18-10-2015



## **Аннотация**

Настоящий документ описывает сетевой протокол стека протоколов «Синхроком» под названием Синхроком-Дата (SCD). SCD - протокол уровня L6 модели OSI/ISO, определяющий способ представления данных в пользовательских сообщениях.

## СОДЕРЖАНИЕ

Введение .....	5
1.1 Мотивация.....	5
1.2. Сфера действия протокола .....	5
1.3 Интерфейсы .....	6
1.4 Работа протокола .....	6
2. Обзор .....	7
2.1. Связь с другими протоколами.....	7
2.2. Модель работы протокола .....	7
2.3. Функциональное описание .....	8
2.3.1 Функция инициализации .....	8
2.3.2 Функция encode .....	8
2.3.3 Функция decode .....	8
2.3.4 Словари .....	8
3. Спецификация.....	8
3.1 Типы данных и правила кодирования.....	8
3.1.1 Передача битовых последовательностей .....	8
3.1.2 Пусто .....	9
3.1.3 Логический .....	9
3.1.4 Резерв .....	9
3.1.5 Беззнаковые.....	9
3.1.6 Целые.....	9
3.1.7 Числа с плавающей точкой .....	10
3.1.8 Строки.....	10
3.1.9 Дата и время .....	10
3.1.10 MAC-адреса Ethernet.....	11
3.1.11 IP-адреса.....	11
3.1.12 Блок данных .....	11

3.1.13 Индексы базовых типов .....	11
3.2 Словарь данных .....	12
3.2.1 Назначение словаря .....	12
3.2.2 Структура элементов словаря .....	12
3.2.3 Разделы словаря.....	13
3.3 Коммуникационные объекты.....	15
3.3.1 Служебные объекты данных .....	16
3.3.2 Объекты данных процесса.....	16
3.3.3 Правила отображение ПОД на СОД.....	16

# Введение

## 1.1 Мотивация

Настоящий документ описывает сетевой протокол стека протоколов «Синхроком» под названием Синхроком-Дата (SCD). SCD - протокол уровня L6 модели OSI/ISO определяющий способ представления данных в пользовательских сообщениях:

- коммуникационные объекты;
- типы передачи данных;
- правила кодирования;
- словари данных.

Протокол SCD не накладывает ограничения на протоколы более низкого уровня по которому могут передаваться пользовательские сообщения. Теоретически, протокол любого уровня стека «Синхроком» могут использоваться для передачи пользовательских сообщений: SCA, SCTP, SCS/IP, TCP/IP, UDP/IP и т.д.

## 1.2. Сфера действия протокола

Протокол SCD ограничивается форматом пользовательского сообщения и относится к уровню 6 модели OSI/ISO.

Протокол позволяет упростить разработку протокола уровня L7 и его реализацию различными подрядчиками, т.к. L7 в плане сообщений будет представлять собой спецификацию классов объектов участвующих в сетевом обмене. Тем самым прикладной уровень протокола L7 освобождается от формальных вопросов форматов и циклов обмена, что зачастую является большой проблемой при разработке сетевых устройств несколькими производителями. Производителям сетевого оборудования, при наличии стека протоколов «Синхроком» заказчика, остается лишь объектно-ориентированно описать в рамках L7 свое устройство и правила обмена (интерфейсы), основываясь на строгой типизации и расширении библиотеки SCD заказчика.

Стек протоколов «Синхроком» относится к L3-L6 слоям модели OSI/ISO.

№	Уровни модели OSI/ISO	Уровни стека протоколов «Синхроком»
7	Прикладной	Прикладные службы
6	Представления	Синхроком-Дата (SCD) - протокол представления данных
5	Сеансовый	Синхроком-Сеанс (SCS)- протокол туннелирования. Приложения, неограниченные требованием жесткого реального времени, могут использовать различные существующие сетевые протоколы на базе IP, которые туннелируются в сети «Синхроком», например: UDP, RTP и т.д.
4	Транспортный	Синхроком-Транспорт (SCTP) - двухфазный протокол

		<p>передачи данных по сети SCA, где:</p> <p>синхронная фаза SCTPS - фаза передачи данных в режиме жесткого реального времени;</p> <p>асинхронная фаза SCTPA - фаза передачи данных в режиме мягкого реального времени.</p>
3	Сетевой	Синхроком-Адрес (SCA)
2	Канальный	IEEE 802.2 на платформе микроядра L4
1	Физический	

Рисунок 1 - Стек протоколов «Синхроком»

### ***1.3 Интерфейсы***

Этот протокол вызывается протоколами взаимодействия L7 и сам вызывает функции сетевых протоколов более низкого уровня для передачи и получения специально форматированных сообщений.

### ***1.4 Работа протокола***

Работа протокола аналогична работе кодека - это кодирование и декодирование строго типизированных объектов сообщений на основе стандартных типов и библиотеки классов коммуникационных объектов.

## 2. Обзор

### 2.1. Связь с другими протоколами

На Рисунок 2 показаны связи стека протоколов «Синхроком» с другими протоколами и приложениями.

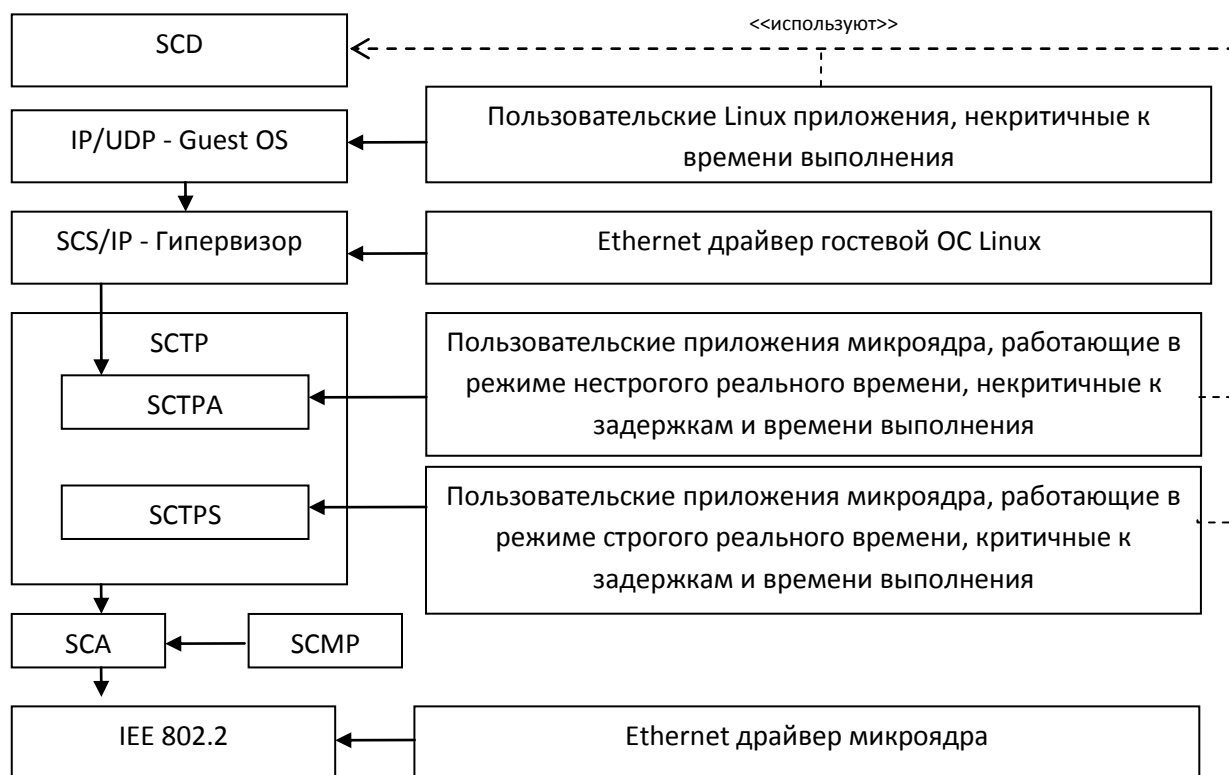


Рисунок 2 - Стек протоколов «Синхроком»

### 2.2. Модель работы протокола

Пользовательское приложение обменивается сообщениями через протокол стека «Синхроком» используя протокол уровня L7, который определяет формат данных сообщений собственной библиотекой коммуникационных объектов, хранящихся в библиотеке SCD.

Пример последовательности работы протокола приведен ниже.

1. Пользовательское приложение при запуске инициализирует кодек SCD с пользовательской библиотекой, в которой поставлены в соответствие друг с другом коммуникационные и информационные объекты.
2. Пользовательское приложение получает сообщение.
3. Вызывается функция декодирования протокола SCD и возвращается результат в виде набора информационных объектов.
4. Пользовательское приложение подготавливает некоторое сообщение в соответствии со своим протоколом уровня L7, которое содержит набор информационных объектов.

5. Вызывается функция кодирования, с входящими параметрами в виде информационных объектов приложения, которая возвращает сообщение в соответствии с протоколом SCD и готовое к отправке.
6. Сообщение отправляется.

## **2.3. Функциональное описание**

### **2.3.1 Функция инициализации**

Для работы протокола SCD необходим словарь. На этапе инициализации программе SCD передается указатель на файл со словарем.

### **2.3.2 Функция encode**

Функция генерирует сообщение. На вход подается коллекция информационных объектов приложения. Результат работы - ссылка на буфер с текстом сообщения.

### **2.3.3 Функция decode**

Функция декодирует входящее сообщение. На вход подается коллекция информационных объектов приложения, которые обновляются значениями из пришедшего сообщения.

### **2.3.4 Словари**

Словари представляют собой бинарные конфигурационные файлы. Способ работы со словарем не отличается от стандартной работы с конфигурационным файлом. Для корректного функционирования протокола необходимо, чтобы словари были идентичными у всех участников коммуникации.

## **3. Спецификация**

### **3.1 Типы данных и правила кодирования**

Данный раздел описывает типы данных и правила кодирования с целью кроссплатформенности и реализации строгой типизации.

#### **3.1.1 Передача битовых последовательностей**

Правила кодирования определяют способы представления различных типов данных при пересылке по сети. Значения данных представляются битовыми последовательностями. Бит может принимать значение 0 или 1. Битовой последовательностью называется упорядоченный набор битов произвольной длины. Может рассматриваться также пустая битовая последовательность (обозначаемая  $\epsilon$ ), длина которой – 0 бит.

Примеры битовых последовательностей: 1001010, 1001010, 1001010

Битовые последовательности разбиваются для передачи в сети на последовательности байтов. Для многобайтовых целочисленных типов данных запись начинается с младшего байта и заканчивается старшим (little-endian).

Пример: число 0xA1B2C3D4 будет записано в виде 0xD4, 0xC3, 0xB2, 0xA1.



Рекомендуется выравнивать переменные на границу байтов. Пусть у нас имеется битовая последовательность длиной N. Определим такое целое положительное число K, что  $8*(K-1) < N \leq 8*K$ . Тогда получается, что указанная битовая последовательность будет передана за K октетов упорядоченных. Биты старшего октета, номер которых превышает N-1 (то есть номер старшего бита в последовательности), не содержат битов из передаваемой последовательности, в общем случае они не определены.

Пример: Битовая последовательность из десяти битов 0011 1000 01, представляющее беззнаковое целое число, и имеющее значение 0x021C будет передана в виде двух последовательных байтов: первый 0x1C второй 0x02.

### 3.1.2 Пусто

Данные базового типа Пусто представляют пустые множества битов и имеют длину 0.

### 3.1.3 Логический

Данные базового типа Логический принимают значения Истина или Ложь. Значения этой переменной представляются битовой последовательностью длиной 1. Значению Истина соответствует битовая последовательность 1, значению Ложь – 0.

Логический тип имеет числовое обозначение.

### 3.1.4 Резерв

Данные типа Резерв представляются битовой последовательностью длиной N бит. Значения данных для типа Резерв N не определяются. Биты в последовательности, для типа Резерв N должны быть либо определены явно, либо помечены как “Любые”. Данные типа Резерв N могут использоваться либо для зарезервированных полей, либо для выравнивания переменных в составных типах данных на границу байтов.

### 3.1.5 Беззнаковые

Данные типа Беззнаковый N имеют значения неотрицательных целых чисел. Диапазон принимаемых значений 0, . . . , 2 N -1. Данные представляются битовой последовательностью длиной N, при этом все разряды кода используются для представления значения числа. Согласно стандарту ISO/IEC 9899:2011, в программах на языке C, беззнаковые типы обозначаются как uintN\_t. Рекомендуется использовать следующие беззнаковые типы:

Название	Тип ISO/IEC 9899:2011	Длина (байт)
Беззнаковый 8	scd_uint8_t	1
Беззнаковый 16	scd_uint16_t	2
Беззнаковый 32	scd_uint32_t	4

### 3.1.6 Целые

Данные типа Целый N имеют значения целых чисел. Диапазон принимаемых значений от -2N-1 до 2N-1 -1. Данные представляются битовой последовательностью длиной N. Знак кодируется в

старшем разряде числа: 0 соответствует положительным, а 1 отрицательным числам.

Рекомендуется использовать следующие типы:

Название	Тип ISO/IEC 9899:2011	Длина (байт)
Знаковый 8	scd_int8_t	1
Знаковый 16	scd_int16_t	2
Знаковый 32	scdint32_t	4

### 3.1.7 Числа с плавающей точкой

Данные базовых типов вещественный 32 и вещественный 64 принимают значения вещественных чисел.

Данные типа вещественный 32 представляются битовой последовательностью длиной 32 бита. Кодирование значений соответствует стандарту ANSI/IEEE 754-1985 для чисел с плавающей точкой одинарной точности.

Данные типа вещественный 64 представляются битовой последовательностью длиной 64 бита. Кодирование значений соответствует стандарту ANSI/IEEE 754-1985 для чисел с плавающей точкой двойной точности.

Данные этих типов могут либо иметь конечное, в том числе нулевое, вещественное значение либо значение NaN (not a number).

Название	Тип ISO/IEC 9899:2011	Длина (байт)
Вещественный 32	scd_float_t	4
Вещественный 64	scd_double_t	8

### 3.1.8 Строки

Строка длиной L символов представляет собой массив байтов uint8\_t [L] в кодировке Unicode UTF-8.

### 3.1.9 Дата и время

Тип Сетевое время предназначен для меток времени и соответствует стандарту IEEE 1588. Время хранится в секундах и наносекундах, прошедших с момента 01-01-1970 00:00.

```
struct SCDDTime {  
  
    scd_int32_t seconds  
  
    scd_int32_t nanoseconds  
  
}
```

Тип Интервал времени содержит такие же поля, что и тип Время, разница хранится в виде разностей в секундах и в наносекундах.

### 3.1.10 MAC-адреса Ethernet

scd\_mac\_t - тип MAC-Адрес представляет собой массив из 6 байт: scd\_uint8\_t [6].

### 3.1.11 IP-адреса

scd\_ip\_t - тип IP-адрес представляет собой массив из 4 байт: scd\_uint8\_t [4].

### 3.1.12 SCA-адреса

scd\_sca\_t - тип IP-адрес представляет собой scd\_uint8\_t.

### 3.1.13 Порт

scd\_port\_t - тип порт представляет собой scd\_uint8\_t.

### 3.1.14 CRC сумма

scd\_crc\_t - тип CRC суммы представляет собой scd\_uint16\_t

### 3.1.15 Размер данных

scd\_size\_t тип размера данных scd\_uint16\_t

### 3.1.16 Размер идентификатора протокола

scd\_protocol\_t тип размера данных scd\_uint8\_t

### 3.1.17 Код ошибки

scd\_err\_t тип размера данных scd\_int8\_t

### 3.1.18 Блок данных

Блок данных служит для передачи произвольного большого блока данных от клиента к серверу и наоборот. Структура внутреннего содержимого таких блоков информации определяется приложением и не рассматривается в этом документе.

### 3.1.19 Индексы базовых типов

Базовые типы данных имеют числовые обозначения (индексы). Они перечислены в таблице:

Тип	Индекс
Логический	1
Целый 8	2
Целый 16	3
Целый 32	4
Беззнаковый 8	5
Беззнаковый 16	6
Беззнаковый 32	7

Вещественный 32	8
Строка	10
Интервал времени	13
Блок данных	15
Вещественный 64	17
Целый 64	21
Беззнаковый 64	27
MAC-адрес	1025
IP-адрес	1026
Сетевое время	1027

## 3.2 Словарь данных

### 3.2.1 Назначение словаря

Словарь данных описывает каждый объект служебных данных в системе. Каждый объект словаря имеет уникальный 16-битный индекс. Таким образом, словарь данных может содержать информацию о 65335 объектах.

### 3.2.2 Структура элементов словаря

Формат элемента словаря данных показан в таблице:

Индекс	Класс объекта	Имя объекта	Тип объекта	Атрибуты доступа	Обязательность
--------	---------------	-------------	-------------	------------------	----------------

Устройство может опционально предоставлять доступ к чтению структур поддерживаемых комплексных типов данных, по соответствующему индексу. Подиндекс 0, при этом содержит количество элементов в этом индексе, а последующие подиндексы содержат закодированные числом UNSIGNED16, типы в соответствии с таблицей. Если элемент(индекс) Словаря содержит несколько подэлементов(подиндексов), то подиндекс 0 содержит количество следующих за ним подиндексов, остальные(за исключением подиндекса 255) описывают подэлементы и представляются как беззнаковый 8.

Столбец Индекс объекта обозначает положение данного элемента в словаре. Индекс играет роль адреса при обращении к желаемому полю данных объекта. Подиндекс используется для обращения к полям данных в составных объектах, таких как массив(ARRAY), или запись(RECORD), и имеет тип UNSIGNED8. Столбец Класс объекта является по сути классификатором объекта в соответствии с таблицей:

Код	Класс объекта	Описание
-----	---------------	----------

0	NULL	Элемент словаря не содержит полей данных.
2	DOMAIN	Простая переменная произвольной длины, например, содержащая исполняемый программный код
5	DEFTYPE	Обозначает что в данном элементе определен базовый тип данных, такой как логический, беззнаковый, вещественный и т.п.
6	DEFSTRUCT	Обозначает что определяется новый тип записи (составной тип)
7	VAR	Отдельная переменная базового типа
8	ARRAY	Объект с множественными полями данных, где каждый элемент переменная простого типа, того же самого что и остальные переменные, другими словами массив. Элемент с подиндексом 0 имеет тип Беззнаковый 8 и не является частью массива
9	RECORD	Запись. Объект с множественными полями данных, где каждый элемент может иметь индивидуальный тип (обязательно простой тип). Элемент с подиндексом 0 имеет тип Беззнаковый 8 и не является частью данных этого объекта

Столбец Имя объекта содержит простое текстовое описание функций данного объекта.

Столбец Тип даёт информацию о типе объекта. Может быть одним из базовых типов или типом, определённых производителем устройства. Не допустимы такие составные типы как записи записей или массивы записей.

Столбец атрибуты доступа определяет возможные варианты доступа к объекту:

Атрибут доступа	Описание
rw	разрешены чтение и запись
wo	только запись
ro	только чтение
const	только чтение, значение никогда не меняется

Столбец Обязательность определяет, является ли наличие объекта в составе Объектного словаря обязательным или нет.

### 3.2.3 Разделы словаря

Множество индексов разделено на разделы, как показано в таблице.

Индексы	Объекты данных
0000	Не используется
0001-001f	Статические (простые) типы данных

0020-003f	Комплексные (составные) типы данных
0040-005f	Комплексные типы данных, определенные производителем
0060-007f	Статические типы данных определяемые профилем устройства.
0080-009f	Комплексные типы данных определяемые профилем устройства.
00a0-0fff	Зарезервировано
1000-1fff	Область параметров связи
2000-5fff	Область профилей устройств, определяемых изготовителем.
6000-9fff	Область профилей стандартизованных устройств.
a000-bfff	Область стандартизованных интерфейсов профилей.
cfff-ffff	Зарезервировано.

Элементы с индексами 01h-1fh содержат определение стандартных типов данных.

Элементы с индексами 40h-5fh оставлены для определения типов данных, специфических для разрабатываемой системы.

Диапазон с индексами 60h-7fh содержит определение типов, используемых стандартизованными классами устройств, диапазон с индексами 80h-9fh содержит определение комплексных типов данных, используемых стандартизованными классами устройств.

Диапазон a0h-25fh зарезервирован для многофункциональных устройств, может содержать определения типов данных, аналогичные элементам 60h-9fh, для устройств другого профиля.

Элементы с индексами 260h-ffff, зарезервированы для возможного будущего расширения.

Диапазон с индексами 1000h-1ffff включает элементы с параметрами коммуникации. Данные элементы общие для всех типов устройств, независимо от профиля к которому они относятся.

Диапазон с индексами 2000h-5ffff для определений особых профилей изготовителя.

Диапазон с индексами 6000h-9ffff, вмещает параметры стандартных профилей устройств.

Диапазон с индексами a000h-ffffh, зарезервирован для будущих применений.

Статические типы данных с индексами 0001-001F содержат определения стандартных типов таких как, логический, целый и т.п. Эти ссылки введены исключительно в справочном порядке и не могут быть ни прочитаны, ни считаны.

Составные типы данных с индексами 0020-003F predetermined структуры состоящие из стандартных типов данных, общие для всех устройств.

Комплексные типы данных, определённые производителем с индексами 0040-005f структуры состоящие из стандартных типов данных, но специфичные для конкретного устройства. Конфигурация устройства может определять дополнительные типы данных специфичные для данного типа устройств.

Статические типы данных, определённые профилем устройства внесены в список с индексами 0060-007f, комплексные с индексами 0080-009f.

Область параметров связи, с индексами 1000-1fff содержит специфические параметры необходимые для обмена по сети. Эти элементы общие для всех устройств.

Область конфигураций стандартизованных устройств с индексами 6000-9fff содержит все объекты данных устройства, общие для класса подобных устройств, которые могут быть записаны или прочитаны через сеть. Конфигурации устройств могут использовать элементы 6000-9fff, чтобы описать параметры устройства и его функциональные возможности. В границах этого диапазона могут быть описаны до 8 отдельных устройств. В этом случае такие устройства называются многофункциональными модулями. Такой модуль вмещает в себя до 8ми сегментов с конфигурациями различных устройств. Эта особенность позволяет создавать устройства с множественной функциональностью. Элементы конфигураций различных устройств сдвинуты относительно друг друга на 800h.

6000h to 67FFh device 0

6800h to 6FFFh device 1

7000h to 77FFh device 2

7800h to 7FFFh device 3

8000h to 87FFh device 4

8800h to 8FFFh device 5

9000h to 97FFh device 6

9800h to 9FFFh device 7

Распределение ПДО, для каждого сегмента многофункционального модуля должно быть осуществлено со смещением 64, то есть первый ПДО второго сегмента начинается с адреса 65. В этой связи в системе поддерживается максимум 8 сегментов. Пространство в словаре с индексами 2000-5fff определено для введённых изготовителем функциональных возможностей.

### ***3.3 Коммуникационные объекты***

Коммуникационные объекты служат для доступа к информационным объектам (переменным) приложения. И описываются протоколами уровня L7 модели OSI/ISO.

Коммуникационные объекты условно делятся на два вида - служебные объекты данных и объекты данных процесса. Протокол уровня L7 может использовать данную классификацию

коммуникационных объектов, либо иметь собственное деление, либо вовсе не детализировать понятие коммуникационного объекта.

Ниже предложена классификация коммуникационных объектов по аналогии с протоколом Ethernet Powerlink (стандарт EPSG DS 301 v.1.10).

### **3.3.1 Служебные объекты данных**

С помощью служебных объектов данных (СОД) обеспечивается доступ для записи и чтения к информационным объектам приложения в асинхронной фазе протокола.

### **3.3.2 Объекты данных процесса**

Передача данных в изохронной фазе протокола осуществляется посредством сообщений, называемых объектами данных процесса (ПОД).

### **3.3.3 Правила отображение ПОД на СОД**

Каждый ПОД может описываться последовательностью мультиплексов, указывающих на элементы служебных объектов данных. Такое описание называется отображением ПОД на СОД. Пользовательские приложения могут поддерживать настройку отображения ПОД.

В том случае, настройка описывается служебными объектами данных, имеющими зарезервированные индексы. Каждый ПОД описывается двумя СОД - параметром связи и параметром отображения. Параметр связи описывает коммуникационные возможности ПОД, структурный параметр определяет содержимое ПОД. Индексы соответствующих объектов вычисляется следующим образом:

- Индекс RPDO параметра связи =  $1400h + \text{Номер ПОД} - 1$ .
- Индекс RPDO структурного параметра =  $1600h + \text{Номер ПОД} - 1$ .
- Индекс TPDO параметра связи =  $1800h$ .
- Индекс TPDO структурного параметра =  $1A00h$ .