Общество с ограниченной ответственностью

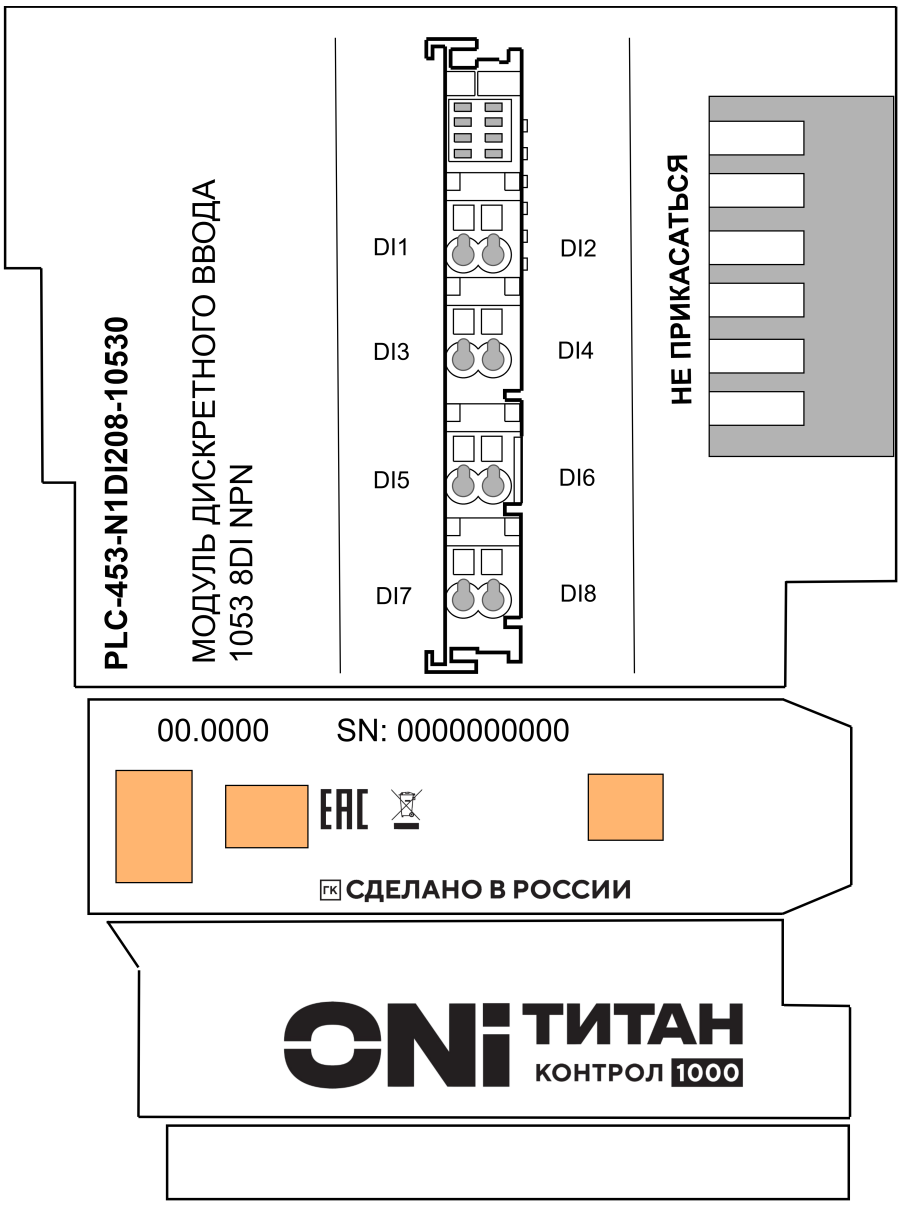
«ГК МФМК»

# КОМПЛЕКС ПРОГРАММНО-ТЕХНИЧЕСКИЙ «ТИТАН КОНТРОЛ 1000»

## Модуль дискретного ввода N1DI208

## Инструкция по настройке и эксплуатации

**Листов 75**




2026

Литера

# 1 Введение

Настоящий документ предназначен для специалистов, занимающихся подключением, конфигурированием и вводом в эксплуатацию модуля дискретного ввода N1DI208, функционирующего по протоколу CANopen в соответствии со стандартом CiA 401 (Device Profile for Generic I/O Modules).

Модуль N1DI208 предназначен для приема дискретного сигнала напряжением 24 В постоянного тока по отрицательному полюсу. Он поддерживает 8 каналов дискретного ввода типа Sourcing input и обеспечивает гальваническую развязку полевых цепей через оптические изоляторы. Типовые устройства, подключаемые к входам модуля, — это контактные датчики и NPN-транзисторы с открытым коллектором (Sinking).

Документ также содержит ссылки на стандартные объекты объектного словаря CiA 401, примеры настройки и рекомендации по использованию фильтрации и индикации входных сигналов.

## 2 Начало работы с CANopen

CAN (Controller Area Network) − это промышленная шина, которая используется для обмена данными между электронными устройствами. В ПТК «ТИТАН КОНТРОЛ 1000» по ней общаются ведущий модуль (вычислительный или интерфейсный) и подключаемые к нему модули расширения (N1DI208 − один из них).

Важно:

- Данные передаются в шестнадцатеричном формате (HEX).
- Каждое сообщение имеет идентификатор (COB-ID), который определяет, какому устройству оно адресовано и какой тип данных передаётся.
- Для работы с CANopen используются SDO-сообщения (Service Data Objects) − они позволяют читать и записывать параметры модуля.

Каждое сообщение в CAN-шине называется кадром.

Структура стандартного CAN-кадра SDO-сообщения состоит из следующих элементов:

| can0 605 [8] 4f 01 10 00 01 00 41 05 | |
|---|---|
| Фрагмент кадра | Описание |
| can0 | Имя CAN-шины |
| 605 | COB-ID сообщения (Сумма Function Code + Node ID) |
| [8] | Количество байт данных в кадре (всегда 8 для SDO) |
| 4f | SDO-команда. Определяет тип операции |
| 01 10 | Индекс объекта |
| 00 | Подындекс объекта |
| 01 00 41 05 | Данные |

В CANopen все 16-битные и 32-битные значения передаются в формате Little-Endian (младшим байтом вперёд). Это означает, что при передаче числа

сначала идёт его младший байт, затем байты по возрастанию, а затем − старший.

| Индекс в кадре | 01 10 | 01 00 41 05 |
|---|---|---|
| Реальное значение | 0x1001 | 0x05410001 |

Ниже более подробно описаны некоторые фрагменты кадра.

**COB-ID**

COB-ID − это полный идентификатор CAN-кадра, который объединяет в себе Function Code и Node-ID. Формируется по формуле:

| COB-ID = Function Code + Node ID |
|---|

Например, если модуль имеет Node-ID = 05, а Function Code для SDO-запроса равен 0x600, то итоговый COB-ID будет равен 0x605 (в формате HEX). Этот идентификатор передаётся в CAN-кадре и позволяет устройствам на шине понять, от кого пришло сообщение, кому оно адресовано и какого типа.

**Node-ID** модуля назначается автоматически в зависимости от порядкового номера его подключения в системе. Например, если модуль подключён пятым по счёту, ему будет присвоен Node-ID = 05.

**Function Code** − это идентификатор типа сообщения в CANopen. Он определяет, какую функцию выполняет передаваемый кадр. Например, все SDO-запросы имеют Function Code = 0x600, а все Heartbeat-сообщения — 0x700. Перечень всех Function Codes приведен в приложении 1.

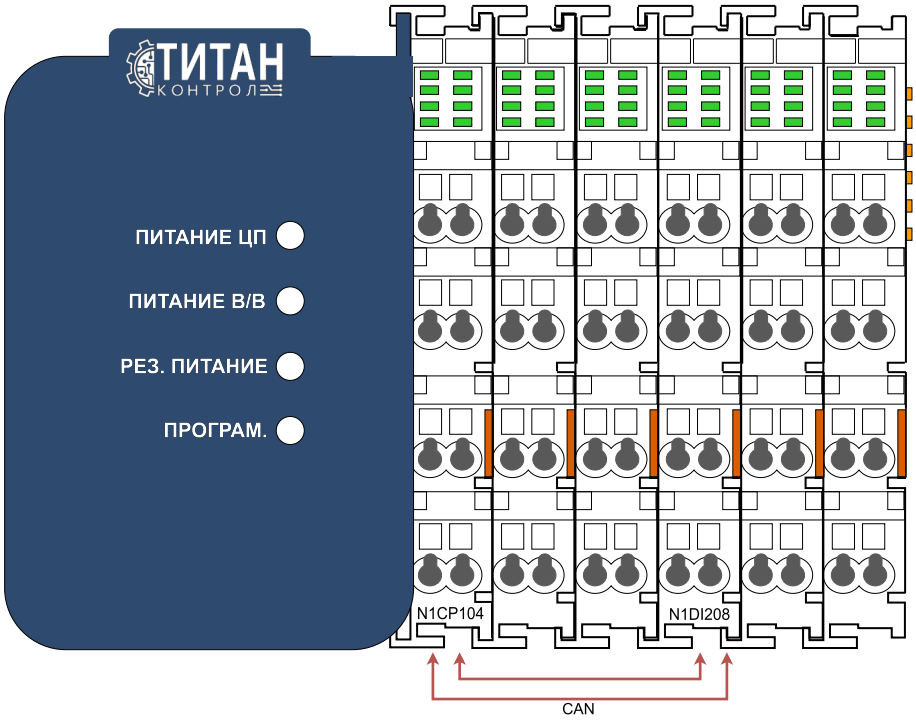
**1. SDO-команда**

SDO-команда (первый байт) − это код операции, который определяет, что именно делает сообщение.

| Значение | Что означает |
|---|---|
| 0x40 | Запрос на чтение объекта |
| 0x4F | Ответ на запись (успешно) |
| 0x60 | Запрос на запись объекта |
| 0x80 | Ответ с ошибкой |

## 3 Подключение модуля

Модуль N1DI208 устанавливается на DIN-рейку в корзину с вычислительным или интерфейсным модулем. Соединение с вычислительным модулем осуществляется с помощью протокола CANopen.




Процедура установки модуля:

1. Подвести модуль к правой свободной стороне уже собранной части ПТК. Совместить направляющие защелки, расположенные на левой стороне корпуса нового модуля, с ответными пазами на правом торце уже установленного модуля.

2. Опустить модуль на рейку до характерного щелчка, свидетельствующего о зацеплении защелки.

3. Подвести к модулю ЦПУ провода питания и Ethernet.

После завершения сборки необходимо подключиться к вычислительному модулю по SSH:

1. В меню пуск ввести «PowerShell» и выбрать соответствующую программу.
2. Ввести команду: ssh root@192.168.152.100
3. Ввести пароль согласно таблице соответствия серийных номеров и паролей вычислительных модулей.

## 4 Работа с шиной can0

В CAN-шине отображаются все сообщения, которыми обмениваются ведущее устройство и модули расширения. Чтобы увидеть этот трафик и начать работу, необходимо подключиться к вычислительному модулю по SSH.

Проверить рабочие ноды на шине возможно следующей командой:

| Команда | Результат |
|---|---|
| candump can0 | Непрерывный вывод всех сообщений на CAN-шине |

Каждый модуль в рабочем состоянии автоматически отправляет в шину CAN короткие Heartbeat-сообщения по умолчанию один раз в секунду. Если Heartbeat от модуля самопроизвольно перестал приходить — модуль отключился, завис или пропало питание.

Структура Heartbeat-сообщения:

| can0 705 [1] 05 | |
|---|---|
| Фрагмент кадра | Описание |
| can0 | Имя CAN-шины |
| 705 | COB-ID сообщения (Node-ID = 5) |
| [1] | Количество байт данных в кадре |
| 05 | Данные (Режим работы модуля) |

Возможные состояния модуля (подробнее см. пункт 6):

| Бит | Режим работы модуля | Что означает |
|---|---|---|
| 0x05 | Operational | Рабочий режим (обмен данными возможен) |
| 0x7F | Pre-operational | Режим настройки |
| 0x04 | Stopped | Остановлен |

При работе с несколькими модулями на шине поток сообщений может быть большим. Чтобы отфильтровать сообщения только от нужного модуля, используйте команду grep вместе с candump.

```
# Показать только сообщения от модуля с Node-ID 5
candump can0 | grep "605\|585\|705\|85"
# Показать только сообщения от модуля с Node-ID 3
candump can0 | grep "603\|583\|703\|83"
```

## 5 Как понимать и использовать команды

Все команды в руководстве описаны по одному принципу. Пример:

| |
|---|
| cansend can0 X#40.00.60.01.00.00.00.00<br>где X = 600h + Node-ID |

Команды зависят от того, на каком месте модуль стоит на CAN-шине (т. е. от его Node ID). Одна и та же команда будет выглядеть по-разному для модуля N1DI208 на первой позиции и для модуля N1DI208 на пятой.

Чтобы правильно адаптировать команду:

1. Определите Node ID модуля (см. пункт 2 руководства).
2. Переведите Node ID в шестнадцатеричную систему (HEX).

| |
|---|
| Для Node ID = 5 → HEX = 5<br>Для Node ID = 10 → HEX = A |

3. Прибавьте полученное значение к указанному в команде. Например:

| |
|---|
| Для Node ID = 5: 600 + 5 = 605<br>Для Node ID = 10: 600 + A = 60A |

4. Подставьте результат вместо X в команду.

| |
|---|
| Для Node ID = 5: cansend can0 605#40.00.60.01.00.00.00.00<br>Для Node ID = 10: cansend can0 60A#40.00.60.01.00.00.00.00 |

## 6 Настройка режима работы

После включения питания модуль N1DI208 автоматически выполняет инициализацию и переходит в состояние Pre-operational. В этом режиме доступны конфигурация устройства через SDO и основные сервисные функции, однако передача PDO-сообщений заблокирована.

Для начала передачи данных от дискретных входов по сети необходимо перевести модуль в режим Operational путем отправки команды Start Node (0x01) с указанием Node-ID устройства.

| cansend can0 000#01.X,<br>где X = $00_h$ + Node-ID устройства (напр. 0A для Node-ID 10) |
|---|

В ответ устройство раз в секунду будет слать в шину строку:

| can0 X [1] 05,<br>где X = $700_h$ + Node-ID устройства (напр. 70A для Node-ID 10) |
|---|

В режиме Operational модуль готов к работе.

**Дополнительно**

При необходимости модуль может быть возвращен в режим Pre-operational командой:

| cansend can0 000#80.X,<br>где X = $00_h$ + Node-ID устройства (напр. 0A для Node-ID 10) |
|---|

В ответ устройство раз в секунду начнет слать в шину строку:

| can0 X [1] 7F,<br>где X = $700_h$ + Node-ID устройства (напр. 70A для Node-ID 10) |
|---|

Также модуль может быть переведен в режим Stopped для полной остановки обмена данными:

| cansend can0 000#02.X,<br>где X = $00_h$ + Node-ID устройства (напр. 0A для Node-ID 10) |
|---|

В ответ устройство раз в секунду начнет слать в шину строку:

| |
|---|
| can0 X [1] 04,<br>где X = $700_h$ + Node-ID устройства (напр. 70A для Node-ID 10) |

Полный сброс модуля в состояние инициализации выполняется командой Reset Node (0x81). После этого модуль автоматически переходит в режим Pre-operational.

| |
|---|
| cansend can0 000#81.X,<br>где X = $00_h$ + Node-ID устройства (напр. 0A для Node-ID 10) |

В ответ устройство один раз отправит сообщение вида:

| |
|---|
| can0 X [1] 00,<br>где X = $700_h$ + Node-ID устройства (напр. 70A для Node-ID 10) |

И затем начнет слать строку, соответствующую режиму Pre-operational.

Сброс только коммуникационного стека модуля выполняется командой Reset Communication (0x82). После этого модуль автоматически переходит в режим Pre-operational.

| |
|---|
| cansend can0 000#82.X,<br>где X = $00_h$ + Node-ID устройства (напр. 0A для Node-ID 10) |

В ответ устройство один раз отправит сообщение вида:

| |
|---|
| can0 X [1] 00,<br>где X = $700_h$ + Node-ID устройства (напр. 70A для Node-ID 10) |

И затем начнет слать строку, соответствующую режиму Pre-operational.

## 7 Обязательные настройки

### Считать состояние дискретных входов (Объект $6000_h$)

Объект $6000_h$ Read input 8-bit считывает состояние дискретных входов модуля. Необходимо выставить единицы на месте тех каналов, которые должны быть включены, и перевести полученное двоичное число в шестнадцатеричную систему:

11111111 — все каналы выдают значения
Переводим в HEX → FF

00000000 — ноль каналов выдает значение
Переводим в HEX → 00

| Подындекс объекта | 1 |
|---|---|
| Назначение | Состояние дискретных входов (8 шт.) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.00.60.01.00.00.00.00,<br>где X = 600h + Node-ID |

### Дополнительно

В этом же объекте хранится информация, сколько групп каналов по 8 штук в каждой поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных групп по 8 входов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.00.60.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Включить/отключить дискретные входы (Объект $6003_h$)

Объект $6003_h$ Filter constant input 8-bit включает и отключает каналы модуля. Необходимо выставить единицы на месте тех каналов, которые должны быть включены, и перевести полученное двоичное число в шестнадцатеричную систему:

11111111 — все каналы включены
Переводим в HEX → FF

00000000 — ни один канал не включен
Переводим в HEX → 00

| Подындекс объекта | 1 |
|---|---|
| Назначение | Включить/отключить каналы (8 шт.) |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FF |
| Команда для чтения | cansend can0 X#40.03.60.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.60.01.YY.00.00.00,<br>где X = 600h + Node-ID<br>YY = какие каналы включены |

**Дополнительно**

В этом же объекте хранится информация, сколько групп каналов по 8 штук в каждой поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных групп по 8 входов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.03.60.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Разрешить передачу данных (Объект $6005_h$)

Объект $6005_h$ Global interrupt enable digital 8-bit регулирует передачу входного значения в CAN. Если установить этот объект в 00, передача входных значений происходить **не будет**. Объекты 6006h, 6007h и 6008h напрямую влияют на разрешение передачи в CAN.

01 — прерывания включены 00 — прерывания не включены

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Разрешить передачу данных |
| Доступ | Чтение и запись |
| Формат | BOOLEAN |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.05.60.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#22.05.60.00.YY.00.00.00,<br>где X = 600h + Node-ID<br>YY = разрешение/запрет прерываний |

### Включить отправку TPDO-1 (Объекты $1800_h$ и $1A00_h$)

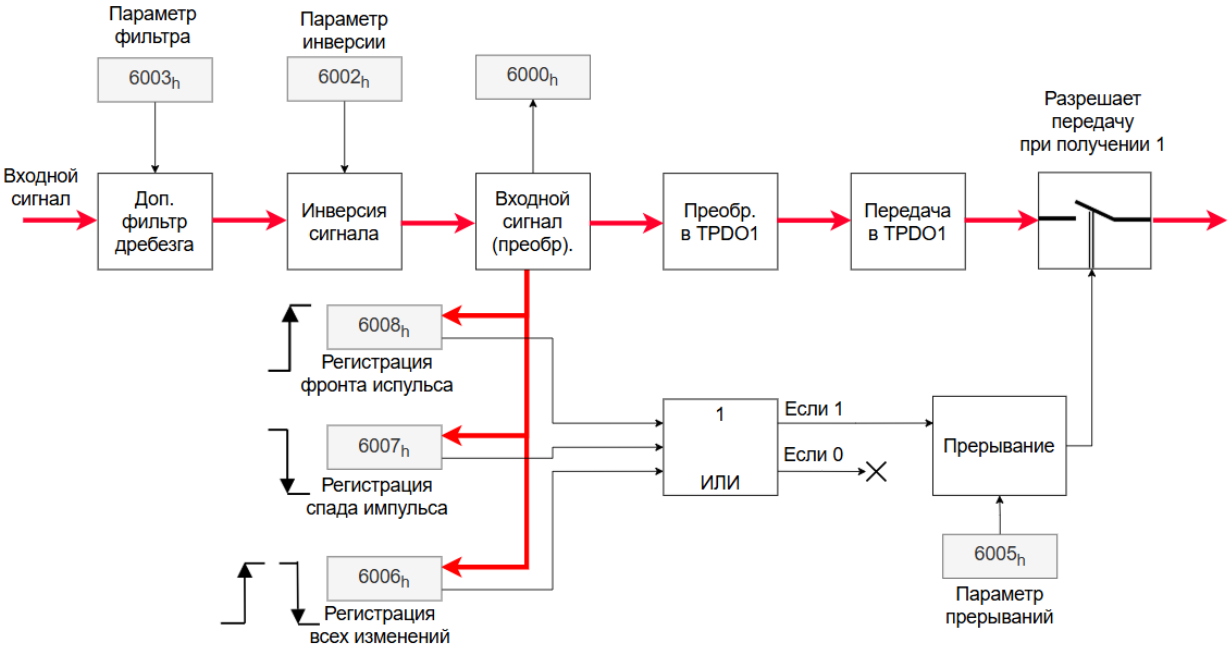
TPDO (Transmit Process Data Object) − это способ передачи данных от модуля к мастеру. С помощью объектов 1800h TPDO communication parameter и 1A00h TPDO mapping parameter настроим модуль на автоматическую отправку состояния дискретных входов (объект 6000h) с заданной периодичностью.

| Шаг | Команда |
|---|---|
| Перевести модуль в режим Pre-operational | cansend can0 000#80.X,<br>где X = $00_h$ + Node-ID |
| Отключить объект $1800_h$ | cansend can0 X#23.00.18.01.82.01.00.80,<br>где X = 600h + Node-ID |
| Отключить объект $1A00_h$ | cansend can0 X#2f.00.1a.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Настроить передачу: объект 6000h, подындекс 01, длина 08 бит. Таким образом формируем строку: 60000108 | |
| Записать сформированную строку в объект 1A00h | cansend can0 X#23.00.1a.01.08.01.00.60,<br>где X = 600h + Node-ID |
| Включить объект $1A00_h$ | cansend can0 X#2f.00.1a.00.01.00.00.00,<br>где X = 600h + Node-ID |
| Настроить тип передачи асинхронный, по изменению или по таймеру | cansend can0 X#2f.00.18.02.fe.00.00.00,<br>где X = 600h + Node-ID |
| Настроить таймер отправки 1 раз в секунду, т.е 1 раз в 1000 мс (0x03E8) | cansend can0 X#2b.00.18.05.e8.03.00.00,<br>где X = 600h + Node-ID |
| Включить объект $1800_h$ | cansend can0 X#23.00.18.01.82.01.00.40,<br>где X = 600h + Node-ID |
| Перевести модуль в режим Operational | cansend can0 000#01.X,<br>где X = $00_h$ + Node-ID |

После выполнения всех шагов модуль начнёт автоматически отправлять состояние своих входов (объект $6000_h$) раз в секунду. Если состояние входов изменится раньше, чем пройдёт секунда, модуль тоже отправит обновление (по изменению).

## 8 Настройка входного дискретного сигнала

По умолчанию модуль поддерживает 8-битный формат входных данных. На рисунке 2 показана взаимосвязь между объектами цифрового ввода при 8-битном доступе.



### Настроить полярность дискретных входов (Объект $6002_h$)

Объект $6002_h$ Polarity input 8-bit инвертирует входной сигнал каналов дискретного модуля. Необходимо выставить единицы на месте тех каналов, которые должны быть инвертированы, и перевести полученное двоичное число в шестнадцатеричную систему:

11111111 — все входы инвертированы
Переводим в HEX → FF

00000000 — ни один вход не инвертирован
Переводим в HEX → 00

| Подындекс объекта | 1 |
|---|---|
| Назначение | Полярность дискретных входов (8 шт.) |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.00.60.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.02.60.01.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = значение полярности в HEX |

**Дополнительно**

В этом же объекте хранится информация, сколько групп каналов по 8 штук в каждой поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных групп по 8 входов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.02.60.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Уведомлять по любому изменению сигнала (Объект $6006_h$)

Объект $6006_h$ Interrupt mask any change 8-bit настраивает, для каких каналов модуль будет отправлять уведомление при любом изменении сигнала (как при переходе с 0 на 1, так и с 1 на 0). Необходимо выставить единицы на месте тех каналов, которые уведомляют о смене фронта сигнала, и перевести полученное двоичное число в шестнадцатеричную систему:

11111111 — уведомления активны
Переводим в HEX → FF

00000000 — уведомления неактивны
Переводим в HEX → 00

| | |
|---|---|
| Подындекс объекта | 1 |
| Назначение | Включить/отключить уведомления (8 шт.) |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FF |
| Команда для чтения | cansend can0 X#40.06.60.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.06.60.01.YY.00.00.00,<br>где X = 600h + Node-ID<br>YY = разрешение/запрет уведомлений |

**Дополнительно**

В этом же объекте хранится информация, сколько групп каналов по 8 штук в каждой поддерживает данный объект.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных групп по 8 входов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.06.60.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Уведомлять о восходящем фронте сигнала (Объект $6007_h$)

Объект $6007_h$ Interrupt mask low-to-high 8-bit настраивает, для каких каналов модуль будет отправлять уведомление при обнаружении восходящего фронта сигнала (переход с 0 на 1). Необходимо выставить единицы на месте тех каналов, которые уведомляют о смене фронта сигнала, и перевести полученное двоичное число в шестнадцатеричную систему:

11111111 — уведомления активны
Переводим в HEX → FF

00000000 — уведомления неактивны
Переводим в HEX → 00

| | |
|---|---|
| Подындекс объекта | 1 |
| Назначение | Включить/отключить уведомления (8 шт.) |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.07.60.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.07.60.01.YY.00.00.00,<br>где X = 600h + Node-ID<br>YY = разрешение/запрет уведомлений |

**Дополнительно**

В этом же объекте хранится информация, сколько групп каналов по 8 штук в каждой поддерживает данный объект.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных групп по 8 входов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.07.60.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Уведомлять о нисходящем фронте сигнала (Объект $6008_h$)

Объект $6008_h$ Interrupt mask high-to-low 8-bit настраивает, для каких каналов модуль будет отправлять уведомление при обнаружении нисходящего фронта сигнала (переход с 1 на 0). Необходимо выставить единицы на месте тех каналов, которые уведомляют о смене фронта сигнала, и перевести полученное двоичное число в шестнадцатеричную систему:

11111111 — уведомления активны
Переводим в HEX → FF

00000000 — уведомления неактивны
Переводим в HEX → 00

| | |
|---|---|
| Подындекс объекта | 1 |
| Назначение | Включить/отключить уведомления (8 шт.) |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.08.60.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.08.60.01.YY.00.00.00,<br>где X = 600h + Node-ID<br>YY = разрешение/запрет уведомлений |

**Дополнительно**

В этом же объекте хранится информация о том, сколько информационных полей поддерживает данный модуль.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных групп по 8 входов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.08.60.00.00.00.00.00,<br>где X = 600h + Node-ID |

## 9 Сохранение и загрузка данных

### Сохранить настройки в память (Объект $1010_h$)

Объект $1010_h$ Store parameters сохраняет текущие настройки модуля в энергонезависимую память. После сохранения настройки не сбрасываются при отключении питания.

Сохраняются следующие типы параметров:

- Все параметры;
- Параметры калибровки ($2000_h$ – $5FFF_h$).

Для защиты от случайной записи сохранение происходит только при отправке специальной команды-подписи save (в виде строки байт: 65 76 61 73 — символы e, v, a, s).

| Подындекс объекта | 1 |
|---|---|
| Назначение | Сохранить все параметры |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 01 00 00 00 |
| Команда для чтения | cansend can0 X#40.10.10.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.10.10.01.73.61.76.65,<br>где X = 600h + Node-ID |

| Подындекс объекта | 4 |
|---|---|
| Назначение | Сохранить параметры калибровки |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 01 00 00 00 |
| Команда для чтения | cansend can0 X#40.10.10.04.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.10.10.04.73.61.76.65,<br>где X = 600h + Node-ID |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 04 |
| Команда для чтения | cansend can0 X#40.10.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Загрузить настройки из памяти (Объект $1011_h$)

Объект $1011_h$ Restore default parameters выгружает сохраненные настройки модуля из энергонезависимой памяти. После сохранения настройки не сбрасываются при отключении питания.

Сбрасываются следующие типы параметров:

- Все параметры;
- Параметры калибровки ($2000_h$ – $5FFF_h$).

Для защиты от случайной записи сброс происходит только при отправке специальной команды-подписи load (в виде строки байт: 6C 6F 61 64 — символы l, o, a, d).

| Подындекс объекта | 1 |
|---|---|
| Назначение | Сбросить все параметры |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 01 00 00 00 |
| Команда для чтения | cansend can0 X#40.11.10.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.11.10.01.6C.6F.61.64,<br>где X = 600h + Node-ID |

| Подындекс объекта | 4 |
|---|---|
| Назначение | Сбросить параметры калибровки |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 01 00 00 00 |
| Команда для чтения | cansend can0 X#40.11.10.04.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.11.10.04.6C.6F.61.64,<br>где X = 600h + Node-ID |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 04 |
| Команда для чтения | cansend can0 X#40.11.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

## 10 Возможные ошибки

### Определить наличие ошибки прямо сейчас (Объект $1001_h$)

Объект $1001_h$ Error register сигнализирует о возникновении ошибки.

Таблица соответствия бита ошибки ее значению представлена ниже.

| Бит | Статус | Значение |
|---|---|---|
| 0 | Обязательный | Неизвестная ошибка |
| 1 | Опциональный | Ошибка по току |
| 2 | Опциональный | Ошибка по напряжению |
| 3 | Опциональный | Ошибка по температуре |
| 4 | Опциональный | Ошибка связи |
| 5 | Опциональный | Ошибка дискретных входов |
| 6 | Опциональный | Зарезервировано (всегда 0) |

| Подындекс объекта | 1 |
|---|---|
| Назначение | Тип ошибки |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.01.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Посмотреть историю ошибок (Объект $1003_h$)

Объект $1003_h$ Pre-defined error field хранит историю ошибок, произошедших на устройстве.

Подындекс 0 содержит суммарное количество ошибок, сохраненных в данном объекте. Запись значения «0» в подындекс 0 очищает всю историю ошибок.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Количество произошедших ошибок |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.03.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

Каждая новая ошибка записывается в подындекс 1, а предыдущие сдвигаются вниз по списку, таким образом формируя журнал. В каждом подындексе хранится код произошедшей ошибки.

| Подындекс объекта | 1-16 |
|---|---|
| Назначение | Код ошибки |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | — |
| Команда для чтения | cansend can0 X#40.03.10.AA.00.00.00.00,<br>где X = 600h + Node-ID,<br>AA = номер подындекса для чтения |

### Узнать ID сообщений об ошибках (Объект $1014_h$)

Объект $1014_h$ COB-ID EMCY показывает идентификатор аварийного сообщения (emergency) устройства в сети CANopen. В нем содержится значение COB-ID, которое используется для передачи аварийных сообщений.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Узнать ID сообщений об ошибках |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 80h + Node-ID |
| Команда для чтения | cansend can0 X#40.14.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Ограничить частоту отправки сообщений об ошибках (Объект $1015_h$)

Объект $1015_h$ Inhibit time EMCY задаёт время (Inhibit Time), ограничивающее частоту отправки аварийных сообщений.

Значение времени должно быть кратно 100 мкс. Запись «0» отключает ограничение частоты.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Ограничить частоту отправки EMCY-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.15.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.15.10.00.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время задержки (в 100-микросекундных интервалах) |

## 11 Информация о модуле

### Определить тип модуля (Объект $1000_h$)

Объект $1000_h$ Device type предназначен для идентификации типа устройства и его функциональности.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Узнать тип устройства |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 91 01 01 00 |
| Команда для чтения | cansend can0 X#40.00.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

## Определить уникальные характеристики модуля (Объект $1018_h$)

Объект $1018_h$ Identity предназначен для хранения данных о производителе, модели и версии. Структура объекта состоит из нескольких подындексов.

| Подындекс объекта | 1 |
|---|---|
| Назначение | ID производителя (Vendor ID) |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | E8 03 00 00 |
| Команда для чтения | cansend can0 X#40.18.10.01.00.00.00.00,<br>где X = 600h + Node-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Номер модели (Product code) |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 0D 00 00 00 |
| Команда для чтения | cansend can0 X#40.18.10.02.00.00.00.00,<br>где X = 600h + Node-ID |

| Подындекс объекта | 3 |
|---|---|
| Назначение | Номер ревизии (Revision number) |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | Индивидуально для каждого модуля |
| Команда для чтения | cansend can0 X#40.18.10.03.00.00.00.00,<br>где X = 600h + Node-ID |

| Подындекс объекта | 4 |
|---|---|
| Назначение | Серийный номер (Serial number) |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | Индивидуален для каждого модуля |
| Команда для чтения | cansend can0 X#40.18.10.04.00.00.00.00,<br>где X = 600h + Node-ID |

**Дополнительно**

В этом же объекте хранится информация о том, сколько информационных полей поддерживает данный модуль.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных информационных полей) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 04 |
| Команда для чтения | cansend can0 X#40.18.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

## 12 Проверка статуса устройства (Heartbeat)

### Настроить частоту отправки heartbeat-сообщения (Объект $1017_h$)

Объект $1017_h$ Producer heartbeat time задаёт, как часто модуль будет отправлять в сеть сообщение-оповещение о своем состоянии.

Если записать в объект значение 0, модуль перестанет отправлять heartbeat-сообщения.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Частота отправки heartbeat-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | E8 03 |
| Команда для чтения | cansend can0 X#40.17.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.17.10.00.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = новая частота отправки |

**Настроить время ожидания heartbeat-сообщения других устройств (Объект $1016_h$)**

Объект $1016_h$ Consumer heartbeat time задаёт время ожидания этого сигнала от других модулей. Если за указанное время статус активности от соседнего модуля не пришёл, устройство фиксирует ошибку связи — значит, модуль отключился или завис.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Количество отслеживаемых устройств |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 08 |
| Команда для чтения | cansend can0 X#40.16.10.00.00.00.00.00,<br>где X = 600h + Node-ID |

Значение подындексов 1-8 представляет собой 32-битное число:

| Биты 31-24 | Биты 23-16 | Бит 15-0 |
|---|---|---|
| 0 | Node ID отслеживаемого модуля | Время ожидания (мс) |

Полученное двоичное число необходимо перевести в Little-Endian.

| Пример записи: | | |
|---|---|---|
| 00000000 | 00000001 | 0000001111101000 |
| Переводим в Little-Endian −> e8 03 01 00 | | |

| | |
|---|---|
| Подындекс объекта | 1 − 8 |
| Назначение | Отслеживаемые устройства 1 − 8 |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.16.10.AA.00.00.00.00,<br>где X = 600h + Node-ID,<br>AA = номер подындекса |
| Команда для записи | cansend can0 X#23.16.10.AA.YY.YY.YY.YY,<br>где X = 600h + Node-ID, |

| | |
|---|---|
| | AA = номер подындекса,<br>YY.YY.YY.YY = значение |

## 13 Настройка обмена данными (SDO-сообщения)

Для обмена данными с другими устройствами модуль использует SDO-каналы:

1. Канал для приема команд и отправки ответов (объект$1200_h$).
2. Канал для отправки команд и приема ответов (объект $1280_h$).

Алгоритм работы с данными объектов $1200_h$ и $1280_h$ выглядит следующим образом:

| Шаг | Пример |
|---|---|
| Выделить биты данных из CAN-кадра | 02 06 00 00 |
| Восстановить 32-битное число | 00 00 06 02 |
| Перевести в двоичный вид | 00000000 00000000 00000110 00000010 |

Рассмотрим структуру полученной двоичной строки:

| Бит 31 | Бит 30 | Бит 29 | Бит 28-0 |
|---|---|---|---|
| 0 | 0 | 0 | 00000 00000000 00000110 00000010 |

Бит 31 отвечает за активность канала

| Значение | Состояние |
|---|---|
| 0 | Канал включён, работает |
| 1 | Канал выключен |

Бит 30 регулирует отправку автоматических PDO-сообщений.

PDO − это способ быстрой передачи данных, при котором модуль отправляет информацию без подтверждения от мастера.

| Значение | Состояние |
|---|---|
| 0 | PDO отправляются |
| 1 | PDO не отправляются |

Бит 29 устанавливает длину COB-ID сообщения.

| Значение | Состояние |
|---|---|
| 0 | Стандартный 11-битный формат |
| 1 | Расширенный 29-битный формат |

Биты 28-0 задают сам COB-ID.

### Настроить COB-ID для приёма команд (Объект $1200_h$)

Объект $1200_h$ SDO server parameter содержит настройки COB-ID каналов, через которые модуль принимает команды и отправляет ответы.

Длина COB-ID установлена стандартной 11-битной.

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID для приема команд |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 0x600 + Node-ID |
| Команда для чтения | cansend can0 X#40.00.12.01.00.00.00.00,<br>где X = 600h + Node-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | COB-ID для отправки ответов |
| Доступ | Только чтение |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 0x580 + Node-ID |
| Команда для чтения | cansend can0 X#40.00.12.02.00.00.00.00,<br>где X = 600h + Node-ID |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 02 |
| Команда для чтения | cansend can0 X#40.00.12.00.00.00.00.00,<br>где X = 600h + Node-ID |

### Настроить COB-ID для отправки команд (Объект $1280_h$)

Объект $1280_h$ SDO client parameter содержит настройки COB-ID каналов, через которые модуль отправляет команды другим модулям и принимает ответы.

Длина COB-ID установлена стандартной 11-битной.

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID для приема команд |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 80 |
| Команда для чтения | cansend can0 X#40.80.12.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.80.12.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | COB-ID для отправки ответов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 80 |
| Команда для чтения | cansend can0 X#40.80.12.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.80.12.02.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 3 |
|---|---|
| Назначение | Node-ID модуля, которому отправляем команды |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.80.12.03.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.80.12.03.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = новый Node-ID |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 03 |
| Команда для чтения | cansend can0 X#40.80.12.00.00.00.00.00,<br>где X = 600h + Node-ID |

## 14 Настройка синхронизации (SYNC-сообщения)

### Задать COB-ID для SYNC (Объект 1005h)

Объект $1005_h$ COB-ID SYNC message задаёт COB-ID для SYNC-сообщений и определяет, будет ли устройство само генерировать синхросигналы или только принимать их.

Не все устройства могут быть генераторами SYNC-сообщений. Если ваше устройство не поддерживает эту функцию, попытка включить генерацию вызовет ошибку. Также нельзя менять COB-ID, пока устройство активно генерирует синхросигналы.

Структура объекта представляет собой 32-битное число:

| Бит 31 | Бит 30 | Бит 29 | Бит 28-0 |
|---|---|---|---|
| 0 | 0 | 0 | 00000 00000000 00000110 00000010 |

Бит 31 зарезервирован (всегда 0).

Бит 30 регулирует генерацию SYNC-сообщений.

| Значение | Состояние |
|---|---|
| 0 | Данное устройство ведомое и потребляет SYNC-сообщения. |
| 1 | Данное устройство ведущее и генерирует SYNC-сообщения. |

Бит 29 устанавливает длину COB-ID SYNC-сообщения.

| Значение | Состояние |
|---|---|
| 0 | Стандартный 11-битный формат |
| 1 | Расширенный 29-битный формат |

Биты 28-0 задают сам COB-ID.

| Подындекс объекта | 0 |
|---|---|
| Назначение | COB-ID для SYNC-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 80 00 00 00 |
| Команда для чтения | cansend can0 X#40.05.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.05.10.00.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

### Задать период отправки SYNC-сообщений (Объект $1006_h$)

Объект $1006_h$ Communication cycle period задает интервал времени (в микросекундах) между двумя последовательными SYNC-сообщениями, которые устройство отправляет в сеть.

Значение «0» указывает на то, что генерация SYNC-сообщений отключена.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Период отправки SYNC-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.06.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.06.10.00.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = период отправки |

### Задать максимальное время синхронизации (Объект $1007_h$)

Объект 1007h Synchronous window length задаёт время ожидания (в микросекундах) от отправки SYNC-сигнала до ответных PDO-сообщения от синхронизирующихся устройств. Если ответы не пришли в этот интервал, синхронизация считается просроченной.

Значение «0» означает бесконечное ожидание.

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Максимальное время синхронизации |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.07.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.07.10.00.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = время ожидания (мкс) |

### Задать значение переполнения счётчика SYNC (Объект $1019_{h)}$

Объект 1019h Synchronous counter overflow value задаёт максимальное значение счётчика SYNC-сигналов. При достижении этого значения счётчик сбрасывается. Значение по умолчанию — 0 (функция отключена).

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Максимальное значение счётчика SYNC |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.19.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.19.10.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = граница переполнения |

## 15 Настройка счётчика времени (TIME STAMP)

### Настроить счётчик времени (Объект $1012_h$)

Объект $1012_h$ COB-ID time stamp object задаёт COB-ID для TIME-сообщений и определяет, будет ли устройство само генерировать синхросигналы или только принимать их.

TIME STAMP — это счётчик времени для всех модулей в сети с момента старта работы. Все устройства отсчитывают от общего нуля.

Не все устройства могут быть генераторами TIME-сообщений. Если ваше устройство не поддерживает эту функцию, попытка включить генерацию вызовет ошибку. Также нельзя менять COB-ID, пока устройство активно генерирует сигналы.

Структура объекта представляет собой 32-битное число:

| Бит 31 | Бит 30 | Бит 29 | Бит 28-0 |
|---|---|---|---|
| 0 | 0 | 0 | 00000 00000000 00000110 00000010 |

Бит 31 регулирует потребление TIME-сообщений.

| Значение | Состояние |
|---|---|
| 0 | Устройство не потребляет TIME-сообщения. |
| 1 | Устройство потребляет TIME-сообщения. |

Бит 30 регулирует генерацию TIME-сообщений.

| Значение | Состояние |
|---|---|
| 0 | Устройство не генерирует TIME-сообщения. |
| 1 | Устройство генерирует TIME-сообщения. |

Бит 29 устанавливает длину COB-ID TIME-сообщения.

| Значение | Состояние |
|---|---|
| 0 | Стандартный 11-битный формат |
| 1 | Расширенный 29-битный формат |

Биты 28-0 задают сам COB-ID.

| Подындекс объекта | 0 |
|---|---|
| Назначение | COB-ID для TIME-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 01 00 00 |
| Команда для чтения | cansend can0 X#40.12.10.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.12.10.00.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

## 16 Настройка параметров связи RPDO ($1400_h$ - $1403_h$ и $1600_h$ - $1603_h$)

RPDO (Receive Process Data Object) — это канал для приёма данных от мастера. В этом разделе описаны объекты, которые настраивают, по какому адресу модуль слушает RPDO-команды, как часто и при каких условиях он их обрабатывает.

У модуля N1DI208 доступны 4 RPDO-канала и 4 RPDO-маппинга: объекты $1400_h$, $1401_h$, $1402_h$, $1403_h$ и $1600_h$, $1601_h$, $1602_h$, $1603_h$ соответственно.

Маппинг − это объект, настраивающий, какие именно объекты и их подындексы будут поступать на модуль через RPDO-канал. RPDO-маппинг работает в паре с RPDO-каналом. Номер пары определяется по последней цифре: например, $1401_h$ работает с $1601_h$, $1402_h$ − с $1602_h$ и так далее. Каналы и маппинги идентичны и отличаются друг от друга только номером.

Поскольку все каналы и маппинги устроены одинаково, в этом разделе подробно разобрана работа с парой **$1400_h$ / $1600_h$**. Остальные пары ($1401_h$ / $1601_h$, $1402_h$ / $1602_h$, $1403_h$ / $1603_h$) настраиваются по тому же принципу — команды для них приведены в соответствующих разделах.

**RPDO-канал $1400_h$**

RPDO-канал $1400_h$ содержит в подындексах следующие параметры:

- COB-ID принимаемого RPDO-сообщения (подындекс 1);
- Тип передачи (подындекс 2);
- Таймер приема регулярных сообщений (подындекс 5);
- Количество параметров (служебный подындекс 0).

Запрещено изменять параметры связи, если RPDO активен.

**COB-ID** представляет собой 32-битное число:

| Бит 31 | Бит 30 | Бит 29 | Бит 28-0 |
|---|---|---|---|
| 0 | 0 | 0 | 00000 00000000 00000110 00000010 |

Бит 31 отвечает за активность канала.

| Значение | Состояние |
|---|---|
| 0 | Канал включён, RPDO обрабатываются. |
| 1 | Канал выключен |

Бит 30 зарезервирован, всегда 0.

Бит 29 устанавливает длину COB-ID RPDO-сообщения.

| Значение | Состояние |
|---|---|
| 0 | Стандартный 11-битный формат |
| 1 | Расширенный 29-битный формат |

Биты 28-0 задают сам COB-ID.

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID RPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $80000200_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.00.14.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.00.14.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

**Тип передачи** определяет, при каких условиях и как часто устройство обрабатывает принятые (RPDO) данные.

- Тип передачи 0-240: модуль принимает данные при приёме следующего SYNC-сигнала.
- Тип передачи 254-255: модуль принимает данные мгновенно, по их поступлению.

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.00.14.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.00.14.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = новый тип передачи |

**Таймер приема регулярных сообщений** определяет, как часто устройство получает регулярные RPDO-сообщения. Значение указывается в микросекундах (мкс).

| Подындекс объекта | 5 |
|---|---|
| Назначение | Таймер приема RPDO-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.00.14.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.00.14.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между приемом сообщений |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 05 |
| Команда для чтения | cansend can0 X#40.00.14.00.00.00.00.00,<br>где X = 600h + Node-ID |

**RPDO-маппинг $1600_h$**

В маппинг можно записывать только те объекты, которые допустимо изменять через RPDO (поддерживают запись).

Подындекс 0 содержит суммарное количество объектов, задействованных в данном маппинге. Запись значения «0» в подындекс 0 отключает маппинг и разрешает редактировать объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Количество получаемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.00.16.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.00.16.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

Каждый объект, получаемый по RPDO, записывается в подындексы 1-8, начиная с первого. В каждом подындексе хранится номер объекта, номер подындекса и длина данных. Общая длина всех объектов в одном RPDO не может превышать 8 байт (64 бита) — это ограничение размера CAN-кадра.

Пример формирования строки объекта, который хотим получать:

| Исходные данные | Номер объекта | Номер подындекса | Длина данных |
|---|---|---|---|
| | 6003 | 01 | 8 |
| Собираем все в одну строку | 60 03 01 08 | | |
| Переводим в Little-Endian | 08 01 03 60 | | |

Список объектов, которые можно получать по RPDO:

- Настроить полярность дискретных входов $6002_h$
- Включить/отключить дискретные входы $6003_h$
- Разрешить передачу данных $6005_h$
- Уведомлять по любому изменению сигнала $6006_h$
- Уведомлять о восходящем фронте сигнала $6007_h$
- Уведомлять о нисходящем фронте сигнала $6008_h$

| Подындекс объекта | 1 − 8 |
|---|---|
| Назначение | Хранение объектов, получаемых по RPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.00.16.YY.00.00.00.00,<br>где X = 600h + Node-ID,<br>YY = номер подындекса |
| Команда для записи | cansend can0 X#23.00.16.AA.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>AA = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

## Алгоритм работы с RPDO-парой $1400_h$ / $1600_h$

| Шаг | Комментарий |
|---|---|
| Перевести модуль в режим Pre-operational | См. раздел 6 «Настройка режима работы» |
| Отключить канал $1400_h$ | См. раздел «RPDO-канал 1400h», подындекс 1 |
| Отключить маппинг $1600_h$ | См. раздел «RPDO-маппинг 1600h», подындекс 0 |
| Записать в подындексы маппинга объекты, получаемые по RPDO | См. раздел «RPDO-маппинг 1600h», подындексы 1-8 |
| Записать в маппинг $1600_h$ количество передаваемых объектов | См. раздел «RPDO-маппинг 1600h», подындекс 0 |
| Настроить тип передачи | См. раздел «RPDO-канал 1400h», подындекс 2 |
| Настроить таймер получения (при необходимости) | См. раздел «RPDO-канал 1400h», подындекс 5 |
| Включить объект $1400_h$ | См. раздел «RPDO-канал 1400h», подындекс 1 |
| Перевести модуль в режим Operational | См. раздел 6 «Настройка режима работы» |

## RPDO-канал $1401_h$

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID RPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $80000300_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.01.14.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.01.14.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.01.14.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.01.14.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = новый тип передачи |

| Подындекс объекта | 5 |
|---|---|
| Назначение | Таймер приема RPDO-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.01.14.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.01.14.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между приемом сообщений |

**Дополнительно**

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 05 |
| Команда для чтения | cansend can0 X#40.01.14.00.00.00.00.00,<br>где X = 600h + Node-ID |

**RPDO-маппинг $1601_h$**

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Количество получаемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.01.16.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.01.16.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

| | |
|---|---|
| Подындекс объекта | 1 − 8 |
| Назначение | Хранение объектов, получаемых по RPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.01.16.YY.00.00.00.00,<br>где X = 600h + Node-ID,<br>YY = номер подындекса |
| Команда для записи | cansend can0 X#23.01.16.AA.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>AA = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

**RPDO-канал $1402_h$**

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID RPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $80000400_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.02.14.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.02.14.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.02.14.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.02.14.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = новый тип передачи |

| Подындекс объекта | 5 |
|---|---|
| Назначение | Таймер приема RPDO-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.02.14.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.02.14.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между приемом сообщений |

### Дополнительно

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 05 |
| Команда для чтения | cansend can0 X#40.02.14.00.00.00.00.00,<br>где X = 600h + Node-ID |

### RPDO-маппинг $1602_h$

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Количество получаемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.02.16.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.02.16.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

| | |
|---|---|
| Подындекс объекта | 1 − 8 |
| Назначение | Хранение объектов, получаемых по RPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.02.16.YY.00.00.00.00,<br>где X = 600h + Node-ID,<br>YY = номер подындекса |
| Команда для записи | cansend can0 X#23.02.16.AA.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>AA = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

**RPDO-канал $1403_h$**

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID RPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $80000500_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.03.14.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.03.14.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.03.14.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.14.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = новый тип передачи |

| Подындекс объекта | 5 |
|---|---|
| Назначение | Таймер приема RPDO-сообщений |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.03.14.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.03.14.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между приемом сообщений |

**Дополнительно**

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 05 |
| Команда для чтения | cansend can0 X#40.03.14.00.00.00.00.00,<br>где X = 600h + Node-ID |

**RPDO-маппинг $1603_h$**

| | |
|---|---|
| Подындекс объекта | 0 |
| Назначение | Количество получаемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.03.16.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.16.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

| | |
|---|---|
| Подындекс объекта | 1 − 8 |
| Назначение | Хранение объектов, получаемых по RPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.03.16.YY.00.00.00.00,<br>где X = 600h + Node-ID,<br>YY = номер подындекса |
| Команда для записи | cansend can0 X#23.03.16.AA.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>AA = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

## 17 Настройка параметров связи TPDO ($1800_h$ - $1803_h$ и $1A00_h$ – $1A03_h$)

TPDO (Transmit Process Data Object) — это канал для передачи данных от модуля мастеру. В этом разделе описаны объекты, которые настраивают, по какому адресу модуль отправляет TPDO-сообщения, как часто и при каких условиях он их передаёт.

У модуля N1DI208 доступны 4 TPDO-канала и 4 TPDO-маппинга: объекты $1800_h$, $1801_h$, $1802_h$, $1803_h$ и $1A00_h$, $1A01_h$, $1A02_h$, $1A03_h$ соответственно.

Маппинг — это объект, настраивающий, какие именно объекты и их подындексы будут отправляться модулем через TPDO-канал. TPDO-маппинг работает в паре с TPDO-каналом. Номер пары определяется по последней цифре: например, $1801_h$ работает с $1A01_h$, $1802_h$ − с $1A02_h$ и так далее. Каналы и маппинги идентичны и отличаются друг от друга только номером.

Поскольку все каналы и маппинги устроены одинаково, в этом разделе подробно разобрана работа с парой **$1800_h$ / $1A00_h$**. Остальные пары ($1801_h$ / $1A01_h$, $1802_h$ / $1A02_h$, $1803_h$ / $1A03_h$) настраиваются по тому же принципу, команды для них приведены в соответствующих разделах.

**TPDO-канал $1800_h$**

TPDO-канал $1800_h$ содержит в подындексах следующие параметры:

- COB-ID отправляемого TPDO-сообщения (подындекс 1);
- Тип передачи (подындекс 2);
- Минимальное время между отправкой TPDO (подындекс 3);
- Event timer (подындекс 5);
- Минимальное количество SYNC-сигналов перед первой отправкой TPDO (подындекс 6);
- Количество параметров (служебный подындекс 0).

Запрещено изменять параметры связи, если TPDO активен.

**COB-ID** представляет собой 32-битное число:

| Бит 31 | Бит 30 | Бит 29 | Бит 28-0 |
|---|---|---|---|
| 0 | 0 | 0 | 00000 00000000 00000110 00000010 |

Бит 31 отвечает за активность канала.

| Значение | Состояние |
|---|---|
| 0 | Канал включён, TPDO обрабатываются. |
| 1 | Канал выключен |

Бит 30 разрешает или запрещает запросы к модулю на передачу данных.

| Значение | Состояние |
|---|---|
| 0 | Модуль принимает запросы на отправку данных. |
| 1 | Модуль не принимает запросы. |

Бит 29 устанавливает длину COB-ID TPDO-сообщения.

| Значение | Состояние |
|---|---|
| 0 | Стандартный 11-битный формат |
| 1 | Расширенный 29-битный формат |

Биты 28-0 задают сам COB-ID.

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID TPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $40000180_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.00.18.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.00.18.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

**Тип передачи** определяет, при каких условиях и как часто устройство передаёт TPDO-данные.

- Тип передачи 0: передача данных по SYNC-сигналу, только если данные изменились
- Тип передачи 1 – 240: передача данных каждый N-й SYNC-сигнал (где N = тип передачи)
- Тип передачи 252: передача данных только по запросу мастера (RTR) после следующего SYNC-сигнала
- Тип передачи 253: мгновенная передача данных по запросу мастера (RTR)
- Тип передачи 254-255: передача данных по любому изменению

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.00.18.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.00.18.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = тип передачи |

**Минимальное время между отправкой TPDO** определяет минимальное время между двумя последовательными отправками TPDO-сообщений. Значение указывается в 100-микросекундных интервалах. Значение «0» отключает ограничение.

| Подындекс объекта | 3 |
|---|---|
| Назначение | Минимальное время между двумя TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.00.18.03.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.00.18.03.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

**Event timer** определяет интервал времени, по истечении которого модуль обязательно отправит TPDO-сообщение, даже если данные не изменились. Значение указывается в микросекундах. Значение «0» отключает ограничение.

| Подындекс объекта | 5 |
|---|---|
| Назначение | Интервал между двумя равными TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.00.18.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.00.18.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

Минимальное количество SYNC-сигналов (подындекс 6) определяет количество SYNC-сигналов, которое модуль пропускает перед первой отправкой TPDO.

| Подындекс объекта | 6 |
|---|---|
| Назначение | Количество SYNC-сигналов перед первой отправкой |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.00.18.06.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.00.18.06.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество SYNC-сигналов |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 06 |
| Команда для чтения | cansend can0 X#40.00.18.00.00.00.00.00,<br>где X = 600h + Node-ID |

**TPDO-маппинг $1A00_h$**

В маппинг можно записывать только те объекты, которые допустимо передавать через TPDO.

**Важно:** подындекс 1 объекта $1A00_h$ по умолчанию содержит маппинг объекта $6000_h$ (0x60000108). Модификация данных значений без необходимости может нарушить функционирование модуля.

Подындекс 0 содержит суммарное количество объектов, задействованных в данном маппинге. Запись значения «0» в подындекс 0 отключает маппинг и разрешает редактировать объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Количество передаваемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 01 |
| Команда для чтения | cansend can0 X#40.00.1A.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.00.1A.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

Каждый объект, передаваемый по TPDO, записывается в подындексы 1-8, начиная с первого. В каждом подындексе хранится номер объекта, номер подындекса и длина данных. Общая длина всех объектов в одном TPDO не может превышать 8 байт (64 бита) — это ограничение размера CAN-кадра.

Пример формирования строки объекта, который хотим передавать:

| Исходные данные | Номер объекта | Номер подындекса | Длина данных |
|---|---|---|---|
| | 6000 | 01 | 8 |
| Собираем все в одну строку | 60 00 01 08 | | |
| Переводим в Little-Endian | 08 01 00 60 | | |

Список объектов, которые можно передавать по TPDO:

- Считать состояние дискретных входов $6000_h$
- Определить наличие ошибки прямо сейчас $1001_h$
- Настроить COB-ID для приема команд $1200_h$

| Подындекс объекта | 1 − 8 |
|---|---|
| Назначение | Хранение объектов, отправляемых по TPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 (08 01 00 60 для подындекса 1) |
| Команда для чтения | cansend can0 X#40.00.1A.BB.00.00.00.00,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для чтения |
| Команда для записи | cansend can0 X#23.00.1A.BB.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

## Алгоритм работы с TPDO-парой 1800h / 1A00h

| Шаг | Комментарий |
|---|---|
| Перевести модуль в режим Pre-operational | См. раздел 6 «Настройка режима работы» |
| Отключить канал $1800_h$ | См. раздел «TPDO-канал $1800_h$», подындекс 1 |
| Отключить маппинг $1A00_h$ | См. раздел «TPDO-маппинг $1A00_h$», подындекс 0 |
| Записать в подындексы маппинга объекты, отправляемые по TPDO | См. раздел «TPDO-маппинг $1A00_h$», подындексы 1-8 |
| Записать в маппинг $1A00_h$ количество передаваемых объектов | См. раздел «TPDO-маппинг $1A00_h$», подындекс 0 |
| Настроить тип передачи | См. раздел «TPDO-канал $1800_h$», подындекс 2 |
| Настроить Event timer (при необходимости) | См. раздел «TPDO-канал $1800_h$», подындекс 5 |
| Настроить Inhibit time (при необходимости) | См. раздел «TPDO-канал $1800_h$», подындекс 3 |

| | |
|---|---|
| Настроить Sync start value (при необходимости) | См. раздел «TPDO-канал 1800h», подындекс 6 |
| Включить объект $1800_h$ | См. раздел «TPDO-канал $1800_h$», подындекс 1 |
| Перевести модуль в режим Operational | См. раздел 6 «Настройка режима работы» |

### TPDO-канал $1801_h$

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID TPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $40000280_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.01.18.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.01.18.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.01.18.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.01.18.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = тип передачи |

| Подындекс объекта | 3 |
|---|---|
| Назначение | Минимальное время между двумя TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |

| | |
|---|---|
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.01.18.03.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.01.18.03.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

| | |
|---|---|
| Подындекс объекта | 5 |
| Назначение | Интервал между двумя равными TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.01.18.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.01.18.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

| | |
|---|---|
| Подындекс объекта | 6 |
| Назначение | Количество SYNC-сигналов перед первой отправкой |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.01.18.06.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.01.18.06.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество SYNC-сигналов |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| | |
|---|---|
| Подындекс объекта | 0 |

| Назначение | Служебный (количество доступных подындексов) |
|---|---|
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 06 |
| Команда для чтения | cansend can0 X#40.01.18.00.00.00.00.00,<br>где X = 600h + Node-ID |

### TPDO-маппинг $1A01_h$

| Подындекс объекта | 0 |
|---|---|
| Назначение | Количество передаваемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.01.1A.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.01.1A.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

| Подындекс объекта | 1 − 8 |
|---|---|
| Назначение | Хранение объектов, отправляемых по TPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.01.1A.BB.00.00.00.00,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для чтения |
| Команда для записи | cansend can0 X#23.01.1A.BB.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

**TPDO-канал $1802_h$**

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID TPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $40000380_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.02.18.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.02.18.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.02.18.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.02.18.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = тип передачи |

| Подындекс объекта | 3 |
|---|---|
| Назначение | Минимальное время между двумя TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.02.18.03.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.02.18.03.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

| Подындекс объекта | 5 |
|---|---|
| Назначение | Интервал между двумя равными TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.02.18.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.02.18.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

| Подындекс объекта | 6 |
|---|---|
| Назначение | Количество SYNC-сигналов перед первой отправкой |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.02.18.06.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.02.18.06.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество SYNC-сигналов |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 06 |
| Команда для чтения | cansend can0 X#40.02.18.00.00.00.00.00,<br>где X = 600h + Node-ID |

## TPDO-маппинг $1A02_h$

| Подындекс объекта | 0 |
|---|---|
| Назначение | Количество передаваемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.02.1A.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.02.1A.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

| Подындекс объекта | 1 − 8 |
|---|---|
| Назначение | Хранение объектов, отправляемых по TPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.02.1A.BB.00.00.00.00,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для чтения |
| Команда для записи | cansend can0 X#23.02.1A.BB.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

**TPDO-канал $1803_h$**

| Подындекс объекта | 1 |
|---|---|
| Назначение | COB-ID TPDO-сообщения |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | $40000480_h$ + Node-ID |
| Команда для чтения | cansend can0 X#40.03.18.01.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#23.03.18.01.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>YY.YY.YY.YY = новый COB-ID |

| Подындекс объекта | 2 |
|---|---|
| Назначение | Тип передачи |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | FE |
| Команда для чтения | cansend can0 X#40.03.18.02.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.18.02.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = тип передачи |

| Подындекс объекта | 3 |
|---|---|
| Назначение | Минимальное время между двумя TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.03.18.03.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.03.18.03.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

| Подындекс объекта | 5 |
|---|---|
| Назначение | Интервал между двумя равными TPDO-сообщениями |
| Доступ | Чтение и запись |
| Формат | UNSIGNED16 |
| Значение по умолчанию | 00 00 |
| Команда для чтения | cansend can0 X#40.03.18.05.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2b.03.18.05.YY.YY.00.00,<br>где X = 600h + Node-ID,<br>YY.YY = время между отправкой сообщений |

| Подындекс объекта | 6 |
|---|---|
| Назначение | Количество SYNC-сигналов перед первой отправкой |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.03.18.06.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.18.06.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество SYNC-сигналов |

**Дополнительно**

В этом же объекте хранится информация, сколько подындексов поддерживает данный объект.

| Подындекс объекта | 0 |
|---|---|
| Назначение | Служебный (количество доступных подындексов) |
| Доступ | Только чтение |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 06 |
| Команда для чтения | cansend can0 X#40.03.18.00.00.00.00.00,<br>где X = 600h + Node-ID |

## TPDO-маппинг $1A03_h$

| Подындекс объекта | 0 |
|---|---|
| Назначение | Количество передаваемых объектов |
| Доступ | Чтение и запись |
| Формат | UNSIGNED8 |
| Значение по умолчанию | 00 |
| Команда для чтения | cansend can0 X#40.03.1A.00.00.00.00.00,<br>где X = 600h + Node-ID |
| Команда для записи | cansend can0 X#2f.03.1A.00.YY.00.00.00,<br>где X = 600h + Node-ID,<br>YY = количество передаваемых объектов |

| Подындекс объекта | 1 − 8 |
|---|---|
| Назначение | Хранение объектов, отправляемых по TPDO |
| Доступ | Чтение и запись |
| Формат | UNSIGNED32 |
| Значение по умолчанию | 00 00 00 00 |
| Команда для чтения | cansend can0 X#40.03.1A.BB.00.00.00.00,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для чтения |
| Команда для записи | cansend can0 X#23.03.1A.BB.YY.YY.YY.YY,<br>где X = 600h + Node-ID,<br>BB = номер подындекса для записи<br>YY.YY.YY.YY = строка объекта |

## 2 Ссылки на используемые стандарты

- CiA 301 CANopen application layer and communication profile

Определяет базовые коммуникационные механизмы CANopen, включая SDO, PDO, NMT, SYNC, HEARTBEAT, объектный словарь и структуру EDS/DCF-файлов.

- CiA 401 CANopen device profile for generic I/O modules

Описывает типовые объекты, поведение и структуру для модулей дискретного и аналогового ввода/вывода. Предоставляет стандартизированную модель объектного словаря для I/O-модулей.

## Приложение 1. Перечень Function Code

| Function Code | Диапазон COB-ID | Тип сообщения | Назначение |
|---|---|---|---|
| 0x000 | 0x000 | NMT | Управление состоянием узлов (запуск, останов, сброс). Глобальный, не содержит Node ID. |
| 0x080 | 0x080 | SYNC | Синхронизация PDO-передач. Глобальный, не содержит Node ID. |
| 0x080 | 0x081 – 0x0FF | EMCY | Сообщения об ошибках. Содержит Node ID. |
| 0x100 | 0x100 – 0x17F | TIME STAMP | Метка времени (опционально). |
| 0x180 | 0x181 – 0x1FF | PDO1 Tx | Первый канал передачи оперативных данных (от узла). |
| 0x200 | 0x201 – 0x27F | PDO1 Rx | Первый канал приёма оперативных данных (к узлу). |
| 0x280 | 0x281 – 0x2FF | PDO2 Tx | Второй канал передачи оперативных данных. |
| 0x300 | 0x301 – 0x37F | PDO2 Rx | Второй канал приёма оперативных данных. |
| 0x380 | 0x381 – 0x3FF | PDO3 Tx | Третий канал передачи оперативных данных. |
| 0x400 | 0x401 – 0x47F | PDO3 Rx | Третий канал приёма оперативных данных. |
| 0x480 | 0x481 – 0x4FF | PDO4 Tx | Четвёртый канал передачи оперативных данных. |
| 0x500 | 0x501 – 0x57F | PDO4 Rx | Четвёртый канал приёма оперативных данных. |
| 0x580 | 0x581 – 0x5FF | SDO Tx | SDO-ответы (от узла к мастеру). |
| 0x600 | 0x601 – 0x67F | SDO Rx | SDO-запросы (от мастера к узлу). |
| 0x700 | 0x701 – 0x77F | Heartbeat | Мониторинг состояния узла (сердцебиение). |
| 0x780 | 0x780 – 0x7FF | Зарезервировано | Для расширенных функций (например, NMT-контроль). |

**Приложение А**

Технические характеристики микроконтроллеров, входящих в состав модулей, представлены в таблице А1.

Таблица А1 – Технические характеристики микроконтроллеров

| Характеристика | Значение |
|---|---|
| Процессор | Не ниже ARM ® Cortex ® -M4 |
| Разрядность | 32-разрядное процессорное ядро |
| Рабочая частота, МГц | Не ниже 100 |
| Системный таймер | 24-разрядный |
| Встроенная память, Мбайт, не менее | 3 |
| Аналого-цифровой преобразователь | Есть |
| Разрешение АЦП, бит | 12 |
| Диапазон входного напряжения АЦП, В | 2,6–3,6 |
| Датчик температуры | Есть |
| Цифроаналоговый преобразователь (ЦАП) | Есть |
| Разрешение ЦАП, бит | 12 |
| Входы-выходы общего назначения, шт, не менее | 100 |
| Внешних линий прерывания | 16 |
| Таймеры | Два 16-разрядных расширенных таймера, десять 16-разрядных общих таймеров |
| Независимые ШИМ каналы, шт | 4 |
| Контроллер интерфейса энкодера | Да, с двумя входами, использующими квадратурный декодер |
| Сторожевой таймер, шт | 2 |
| Часы реального времени (RTC) | 32-разрядный счетчик с программируемым 20-разрядным прескалером, функцией сигнализации и функцией прерывания и пробуждения |

Окончание таблицы А1

| Характеристика | Значение |
|---|---|
| Интерфейс I2C | Не менее 2-х интерфейсов шины I2C, с поддержкой как ведущего, так и ведомого режима, с частотой до 1 МГц |
| Последовательный периферийный интерфейс (SPI) | Не менее 2-х интерфейсов SPI с частотой до 30 МГц и поддержкой как ведущего, так и ведомого режимов, аппаратного расчета CRC и автоматической проверкой ошибок CRC при передаче данных |
| Универсальный синхронный асинхронный приемник-передатчик (USART) | Не менее 2-х |
| Полноскоростной интерфейс устройства с универсальной последовательной шиной (USB) | Один с частотой до 12 Мбит/с, внутренним генератором 48 МГц, поддерживающим работу без кристаллов |
| Локальная сеть контроллера (CAN) | Один интерфейс CAN 2.0 B с частотой связи до 1 Мбит/с |
| Защищенный интерфейс платы цифрового ввода-вывода (SDIO) | Поддержка хост-интерфейса SD2.0/SDIO2.0/MMC4.2 |
| Контроллер внешней памяти (EMC) | SRAM, SRAM-накопитель, ROM и NOR-Flash, NAND-флэш-память, 16-битная шина данных |
| Режим отладки | Последовательный проводной порт отладки JTAG (SWJ-DP) |