Інтерфейсні модулі ION

Опис протоколу Modbus/RTU

В даному документі наведено короткий опис протоколу MODBUS/RTU з передачею даних по послідовному інтерфейсу для інтерфейсних модулів ION

06.02.2021

ЗМІСТ

ФІЗИЧНИЙ РІВЕНЬ

Як середовище передачі даних використовується двопровідний (напівдуплексний) інтерфейс RS-485.

Вимоги до параметрів середовища передачі даних наведено в стандарті ANSI/TIA/EIA-485-A-98.

Кабель зазвичай містить 3 провідника в загальному екрані, два з яких представляють собою виту пару, а третій з'єднує загальні ("земляні") виводи всіх інтерфейсів RS-485 в мережі. Загальний провід і екран повинні бути заземлені тільки в одній точці, бажано біля ведучого пристрою. Для невеликих швидкостей (до 19200 біт/с) і відстаней (до 10 м) можливо використовувати кабель з двома провідниками без екрану.

На кожному кінці кабелю повинні бути встановлені резистори (в інтерфейсних модулях присутні перетинки, що дозволяють підключати вбудовані термінальні резистори номіналом 120 Ом) для узгодження лінії передачі, як це вимагає інтерфейс RS-485.

Модулі підтримують швидкості обміну 2400 біт/с, 4800 біт/с, 9600 біт/с, 14400 біт/с, 19200 біт/с, 38400 біт/с, 28800 біт/с и 57600 біт/с. За замовчуванням швидкість обміну дорівнює 9600 біт/с.

КАНАЛЬНИЙ РІВЕНЬ

Канальний рівень забезпечує створення, передачу і прийом кадрів даних. Цей рівень обслуговує запити мережевого рівня і використовує сервіс фізичного рівня для прийому і передачі пакетів.

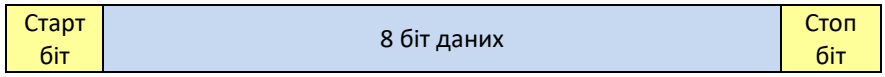
Протокол Modbus є протоколом типу "ведучий-відомий" («головний-підлеглий»), тобто в один і той же час до шини може бути підключений тільки однин ведучий пристрій (master/майстер) і кілька підлеглих (slave/слейв). Передача даних ініціюється завжди майстром (ведучим пристроєм). Слейви (підлеглі пристрої) можуть відповідати лише на запити ведучого.

Ведучий пристрій може ініціювати запити до конкретного підлеглого пристрою (unicast mode) або всім підлеглим пристроям (broadcast mode - широкомовний запит). Підлеглі пристрої не відповідають на широкомовні запити, а тільки приймають їх. Для передачі широкомовних запитів використовується адреса 0.

Дана реалізація протоколу Modbus передбачає використання адрес слейвів в діапазоні 1-247. Кожен пристрій в мережі повинен мати унікальну адресу. Максимальна кількість пристроїв в мережі – 32.

Посилка кожного байта починається з одного старт-біта, після якого йдуть 8 біт даних і один стоп-біт (формат 8N1, рис 1а) або два стоп-біта (формат 8N2, рис 1б) чи один біт паритету (парний паритет – 8E1 або непарний паритет – 8O1, рис 1в, 1г) та стоп-біт.

а) формат 8N1



б) формат 8N2

в) формат 8E1

г) формат 8O1

Рис 1 - Формат даних

За замовчуванням в RTU режимі біт паритету встановлюють рівним 1, якщо кількість двійкових одиниць в байті непарне, і рівним 0, якщо воно парне. Такий паритет називають парним (even parity) і метод контролю називають контролем парності.

При парній кількості двійкових одиниць в байті біт паритету може бути рівним 1. У цьому випадку говорять, що паритет є непарним (odd parity).

Контроль парності може бути відсутнім взагалі. У цьому випадку замість біта паритету може використовуватися другий стоповий біт (формат 8N2). Для забезпечення максимальної сумісності з іншими продуктами рекомендується використовувати можливість заміни біта паритету на другий стоповий біт.

У модулях встановлений за замовчуванням формат передачі даних 8N2 (старт-біт + 8 біт даних + 2 стоп-біта).

Обмін даними по протоколу проводиться фреймами (пакетами даних).

Адреса	Функція	Дані	Контрольна сума
1 байт	1 байт	0 … 124 байт	2 байта

Рис 2 - Структура фрейма

Фрейм розпочинається з посилки адреси пристрою (1 байт), до якого відправляється запит (або адреса пристрою, який формує відповідь). Діапазон можливих значень адрес: 0-247. Адреса 0 є широкомовною і призначена для передачі інформації всім пристроям в мережі. Запит з нульовою адресою не передбачає відповіді від слейва.

Після передачі адреси слідує код функції (1 байт), що визначає функціональну належність запиту (відповіді). Коли слейв відповідає майстру, він використовує поле коду функції для фіксації помилки. У разі нормальної відповіді слейв повторює оригінальний код функції. Якщо має місце помилка, повертається код функції із встановленим в 1 старшим бітом.

Наприклад, повідомлення майстра «прочитати стан дискретних входів» має код функції 0000 0010 (0x02). Якщо слейв виконав запит без помилки, він повертає такий же код. Якщо має місце помилка, то він повертає 1000 0010 (0x82). У цьому випадку крім зміни коду функції, слейв розміщує в поле даних унікальний код (див. коди помилок), який повідомляє майстру, яка саме помилка сталася (або причину помилки).

Після передачі коду функції слідує передача даних, яка здійснюється побайтно. Кількість переданих байт – 0…124 (загальна довжина пакету – до 128 байт). Поле даних в повідомленні від майстра до слейва містить додаткову інформацію, яка необхідна слейву для виконання зазначеної функції. Воно може містити адреси регістрів або виходів, їх кількість, розмір переданих байтів даних.

Після передачі даних слідує контрольна сума (2 байта), що призначена для перевірки цілісності інформації, яка приймається в форматі [MSB (старший байт) | LSB (молодший байт)].

Контрольна сума підраховується і додається в кінець фрейму пристроєм, що передає інформацію і порівнюється пристроєм одержувача з контрольною сумою, що підрахована за прийнятими даними. У підрахунку контрольної суми використовуються всі байти кадру, починаючи з першого (адреси).

ПРИКЛАДНИЙ РІВЕНЬ

Прикладний рівень Modbus базується на запитах за допомогою кодів функцій, які вказують підлеглому пристрою, яку операцію потрібно виконати.

Стандартом Modbus передбачені три категорії кодів функцій: встановлені стандартом, функції, що задаються користувачем та зарезервовані.

Коди функцій є числами в діапазоні від 1 до 127 (0x01 … 0x7F). Коди в діапазоні від 65 до 72 та від 100 до 110 (0x41-0x48 и 0x64-0x6E) відносяться до заданих користувачем функцій, в діапазоні від 128 до 255 (0x80-0xFF) зарезервовані для пересилання кодів помилок у відповідному повідомленні. Код «0» не використовується.

ПЕРЕЛІК ВИКОРИСТОВУВАНИХ ФУНКЦІЙ ПРОТОКОЛУ MODBUS/RTU

Тип адресації	Режим	Опис функції	Псевдонім	Код функції (hex)
Бітова адресація	Читання	Зчитування стану релейних виходів	READ_COILS	0x01
		Зчитування стану дискретних входів	READ_DISCRETE_INPUT	0x02
	Запис	Запис стану одного релейного виходу	WRITE_COIL	0x05
		Запис стану декількох релейних виходів	WRITE_MULTIPLE_COILS	0x0F
16-бітна адресація	Читання	Зчитування регістрів параметрів	READ_HOLDING_REGISTERS	0x03
		Зчитування регістрів даних	READ_INPUT_REGISTERS	0x04
	Запис	Запис одного регістра параметрів	WRITE_REGISTER	0x06
		Запис декількох регістрів параметрів	WRITE_MULTIPLE_REGISTERS	0x10
	Діагностика		DIAGNOSTIC	0x08
	Зчитування відомостей про пристрій		REPORT_SLAVE_ID	0x11

0X01 – ЗЧИТУВАННЯ СТАНУ РЕЛЕЙНИХ ВИХОДІВ (READ_COILS)

Ця функція використовується для зчитування стану від 1 до 16 релейних виходів. Нумерація релейних виходів починається з нуля. Значення бітів: 1 – відповідає включеному стану реле, 0 – вимкненому.

Виклик номера, що виходить за межі кількості релейних виходів, повертає помилку. Наприклад, в пристрої, який має 8 виходів (0x0000 ... 0x0007), на запит 9-го номера (0x0008), відповідь поверне помилку.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x01
Початковий номер виходів	2 байта	0x0000 … 0x000F (0 … 15)
Кількість виходів	2 байта	0x0001 … 0x0010 (1 … 16)
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x01
Число байт у відповіді	1 байт	N (1 … 2)
Байти стану виходів	N * 1 байт	N * 0xXX
CRC	2 байта	

N = [Кількість виходів для зчитування]/8, якщо залишок > 0, то приймається N=N+1

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x81
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

Приклад

Запит

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x01
Початковий номер виходів	2 байта	0x0003
Кількість виходів	2 байта	0x000D
CRC	2 байта	0xXXXX

Відповідь

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x01
Число байт у відповіді	1 байт	0x02
Стан виходів 11-4	1 байт	0xAC
Стан виходів 12-11	1 байт	0x40
CRC	2 байта	0xXXXX

У прикладі запитується стан 13-ти виходів (0x0D) пристрою з адресою 0x07, починаючи з 4-го входу (0x03) по 16-ий. У відповіді - два байта даних (0xAC 0x40), що в двійковому поданні є 1010 1100 0000 0010.

Стан виходу	1	0	1	0	1	1	0	0	0	0	0	0	0	0	1	0
Номер виходу	11	10	9	8	7	6	5	4	-	-	-	-	-	-	13	12

0X02 – ЗЧИТУВАННЯ СТАНУ ДИСКРЕТНИХ ВХОДІВ (READ_DISCRETE_INPUT)

Ця функція використовується для зчитування від 1 до 16 дискретних входів. Нумерація дискретних входів починається з нуля (входи 1-32 адресуються як 0-31). Значення бітів: 1 – відповідає замкнутому стану входу, 0 – розімкненому.

Виклик номера, що виходить за межі кількості дискретних входів, повертає помилку. Наприклад, в пристрої, який має 8 входів (0x0000 … 0x0007), на запит 9-го номера (0x0008), відповідь поверне помилку.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x02
Початковий номер входів	2 байта	0x0000 … 0x000F (0 … 15)
Кількість входів	2 байта	0x0001 … 0x0010 (1 … 16)
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x02
Число байт у відповіді	1 байт	N (1 … 2)
Байти стану входів	N * 1 байт	N * 0xXX
CRC	2 байта	

N = [Кількість входів для зчитування]/8, якщо залишок > 0, то приймається N=N+1

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x82
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

Приклад аналогічний попередньому (функція 0x01).

0X03 – ЗЧИТУВАННЯ РЕГІСТРІВ ПАРАМЕТРІВ (READ_HOLDING_REGISTERS)

Ця функція використовується для послідовного зчитування від 1 до 60 регістрів параметрів. Нумерація регістрів починається з нуля. Регістри 16-ти бітні (слово, що складається з 2 байт по 8 біт), беззнакові або знакові (в додатковому коді). 32-х бітні регістри розбиваються на два 16-ти бітних слова в форматі [HIword, LOWword].

Регістри параметрів зберігають налаштування пристрою (параметри порту RS-485, правила обробки натискань кнопок і т.д.).

Максимальна кількість зчитування регістрів за один раз – не більше 60-ти. Для збереження швидкодії модулів на високому рівні рекомендується за один раз читати не більше 10 регістрів.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x03
Початкова адреса	2 байта	0x0000 … 0xFFFF (0 … 65535)
Кількість регістрів для зчитування	2 байта	0x0001 … 0x003C (1 … 60)
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x03
Число байт у відповіді	1 байт	2 * N
Байти регістрів	N * 2 байт	N * 0xXXXX
CRC	2 байта	

N = Кількість регістрів для зчитування

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x83
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

Адреси регістрів та їх опис вказані в таблиці регістрів параметрів.

Приклад

Запит

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x03
Початкова адреса	2 байта	0x006B
Кількість регістрів для зчитування	2 байта	0x0003
CRC	2 байта	0xXXXX

Відповідь

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x03
Число байт у відповіді	1 байт	0x06
Байти регістра 1	2 байт	0xXXXX
Байти регістра 2	2 байт	0xXXXX
Байти регістра 3	2 байт	0xXXXX
CRC	2 байта	0xXXXX

У прикладі запитуються дані 3-х регістрів (0x0003) пристроя 0x07, починаючи з адреси регістрів 0x006B.

0X04 – ЗЧИТУВАННЯ РЕГІСТРІВ ДАНИХ (READ_INPUT_REGISTERS)

Ця функція використовується для послідовного зчитування від 1 до 60 регістрів даних. Нумерація регістрів починається з нуля. Регістри 16-ти бітні (слово, що складається з 2 байт по 8 біт), беззнакові або знакові (в додатковому коді). 32-х бітні регістри розбиваються на два 16-ти бітних слова в форматі [HIword, LOWword].

Регістри даних зберігають стани дискретних і аналогових входів та виходів і інші дані.

Максимальна кількість зчитування регістрів за один раз – не більше 60-ти. Для збереження швидкодії модулів на високому рівні рекомендується за один раз читати не більше 10 регістрів.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x04
Початкова адреса	2 байта	0x0000 … 0xFFFF (0 … 65535)
Кількість регістрів для зчитування	2 байта	0x0001 … 0x003C (1 … 60)
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x04
Число байт у відповіді	1 байт	2 * N
Байти регістрів	N * 2 байт	N * 0xXXXX
CRC	2 байта	

N = Кількість регістрів для зчитування

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x84
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

Адреси регістрів та їх опис вказані в таблиці регістрів даних.

0X05 – ЗАПИС СТАНУ ОДНОГО РЕЛЕЙНОГО ВИХОДУ (WRITE_COIL)

Ця функція використовується для запису стану (включення/відключення) одного релейного виходу за обраною адресою. Нумерація релейних виходів починається з нуля. Стан передається двома байтами, де 0xFF00 – відповідає включеному стану реле, 0x00FF – вимкненому.

При успішному виконанні команди у відповідь пристрій надсилає копію запиту.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x05
Номер виходу	2 байта	0x0000 … 0x001F (0 … 31)
Значення для запису	2 байта	0xFF00 або 0x00FF
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x05
Номер виходу	2 байта	0x0000 … 0x001F (0 … 31)
Значення для запису	2 байта	0xFF00 або 0x00FF
CRC	2 байта	

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x85
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

0X06 – ЗАПИС ОДНОГО РЕГІСТРА ПАРАМЕТРІВ (WRITE_REGISTER)

Ця функція використовується для запису одного регістра параметрів.

Нумерація регістрів починається з нуля. Регістри 16-ти бітні (слово, що складається з 2 байт по 8 біт), беззнакові або знакові (в додатковому коді). 32-х бітні регістри розбиваються на два 16-ти бітних слова в форматі [HIword, LOWword].

Регістри параметрів зберігають налаштування пристрою (параметри порту RS-485, правила обробки натискань кнопок і т.д.).

При успішному виконанні команди у відповідь пристрій надсилає копію запиту.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x06
Номер регістра	2 байта	0x0000 … 0xFFFF
Значення для запису	2 байта	0x0000 … 0xFFFF
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x06
Номер регістра	2 байта	0x0000 … 0xFFFF
Значення для запису	2 байта	0x0000 … 0xFFFF
CRC	2 байта	

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x86
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

Адреси регістрів та їх опис вказані в таблиці регістрів параметрів.

Приклад

Запит

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x06
Номер регістра	2 байта	0x0101
Значення для запису	2 байта	0x0005
CRC	2 байта	0xXXXX

Відповідь

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x06
Номер регістра	2 байта	0x0101
Значення для запису	2 байта	0x0005
CRC	2 байта	0xXXXX

У прикладі встановлюється швидкість обміну даними 19200 біт/с пристрою 0x07, шляхом запису значення 0x0005 за адресою регістра параметрів 0x0101.

0X0F – ЗАПИС СТАНУ ДЕКІЛЬКОХ РЕЛЕЙНИХ ВИХОДІВ (WRITE_MULTIPLE_COILS)

Ця функція використовується для групового запису стану кількох релейних виходів певного діапазону. Нумерація релейних виходів починається з нуля. Стан одного реле задається одним бітом (1 – відповідає включеному стану, 0 – вимкненому). При установці стану менш ніж 8-ми реле, використовуються молодші біти в 2-байтному регістрі, а старші заповнюються нулями.

При успішному виконанні команди відповідь має формат: функція, початкова адреса запису, число встановлених релейних виходів.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x0F
Початковий номер виходів	2 байта	0x0000 … 0x000F (0 … 15)
Кількість виходів	2 байта	0x0001 … 0x0010 (1 … 16)
Число байт	1 байт	N (1 … 2)
Байти стану виходів	N * 1 байт	N * 0xXX
CRC	2 байта	

N = [Кількість виходів]/8, якщо залишок > 0, то приймається N=N+1

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x0F
Початковий номер виходів	2 байта	0x0000 … 0x001F (0 … 31)
Число встановлених виходів	2 байта	0x0001 … 0x0020 (1 … 32)
CRC	2 байта	

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x8F
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

0X10 – ЗАПИС ДЕКІЛЬКОХ РЕГІСТРІВ ПАРАМЕТРІВ (WRITE_MULTIPLE_REGISTERS)

Ця функція використовується для групового запису кількох регістрів параметрів певного діапазону.

Нумерація регістрів починається з нуля. Регістри 16-ти бітні (слово, що складається з 2 байт по 8 біт), беззнакові або знакові (в додатковому коді). 32-х бітні регістри розбиваються на два 16-ти бітних слова в форматі [HIword, LOWword]. Регістри параметрів зберігають налаштування пристрою (параметри порту RS-485, правила обробки натискань кнопок і т.д.).

При успішному виконанні команди відповідь має формат: функція, початкова адреса запису, число записаних регістрів.

Максимальна кількість записуваних регістрів за один раз - не більше 20-ти. Для збереження швидкодії модулів на високому рівні рекомендується за один раз записувати не більше 5 регістрів.

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x10
Адреса початку запису	2 байта	0x0000 … 0xFFFF (0 … 65535)
Число регістрів	2 байта	0x0001 … 0x0014 (1 … 20)
Число байт	1 байт	2 * N
Значення для запису	N * 2 байт	N * 0xXXXX
CRC	2 байта	

N = Кількість регістрів для запису

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x10
Адреса початку запису	2 байта	0x0000 … 0xFFFF (0 … 65535)
Число записаних регістрів	2 байта	0x0001 … 0x0014 (1 … 20)
CRC	2 байта	

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x90
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

Адреси регістрів та їх опис вказані в таблиці регістрів параметрів.

Приклад

Запит

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x10
Адреса початку запису	2 байта	0x0101
Число регістрів	2 байта	0x0002
Число байт	1 байт	0x0004
Значення для запису 1	2 байта	0x0005
Значення для запису 2	2 байта	0x0002
CRC	2 байта	0xXXXX

Відповідь

Адреса пристрою	1 байт	0x07
Функція (код запиту)	1 байт	0x10
Адреса початку запису	2 байта	0x0101
Число записаних регістрів	2 байта	0x0002
CRC	2 байта	0xXXXX

У прикладі встановлюється швидкість обміну даними 19200 біт/с і формат передачі даних 8N2 пристрою 0x07, шляхом запису двох значень (0x0005 і 0x0002) за адресами регістра параметрів 0x0101 і 0x0102 (початкова адреса 0x0101, число регістрів – 0x0002).

0X08 – ДІАГНОСТИКА (DIAGNOSTIC)

Функція діагностики містить набір команд для перевірки комунікаційної системи. Для вибору відповідної команди в запит після коду функції необхідно додати два байта, містять код команди.

Лічильники підраховують дані з моменту включення пристрою і скидаються при відключенні (перезавантаженні).

Дана реалізація протоколу Modbus підтримує наступні команди діагностики:

	Адреса	**Функція**	**Команда**	**Дані**	**CRC**
0x0000 – Повернення запиту					
Запит	0x01 … 0xF7	0x08	**0x0000**	0xXXXX	
Відповідь	0x01 … 0xF7	0x08	0x0000	0xXXXX	
0x000A – Скидання лічильників і регістра діагностики					
Запит	0x01 … 0xF7	0x08	**0x000A**	0x0000	
Відповідь	0x01 … 0xF7	0x08	0x000A	0x0000	
0x000C – Зчитування кількості запитів з помилками CRC					
Запит	0x01 … 0xF7	0x08	**0x000C**	0x0000	
Відповідь	0x01 … 0xF7	0x08	0x000C	0xXXXX	
0x000D – Зчитування кількості запитів з помилками					
Запит	0x01 … 0xF7	0x08	**0x000D**	0x0000	
Відповідь	0x01 … 0xF7	0x08	0x000D	0xXXXX	
0x000E – Зчитування кількості адресованих пристрою запитів					
Запит	0x01 … 0xF7	0x08	**0x000E**	0x0000	
Відповідь	0x01 … 0xF7	0x08	0x000E	0xXXXX	
0x000F – Зчитування кількості запитів без відповіді					
Запит	0x01 … 0xF7	0x08	**0x000F**	0x0000	
Відповідь	0x01 … 0xF7	0x08	0x000F	0xXXXX	

0X11 – ЗЧИТУВАННЯ ВІДОМОСТЕЙ ПРО ПРИСТРІЙ (REPORT_SLAVE_ID)

Функція повертає інформацію про пристрій у вигляді:

Запит

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x11
CRC	2 байта	

Відповідь

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x11
Число байт у відповіді	1 байт	37
Ідентифікатор пристрою	1 байт	0xAA
Індикатор пуску	1 байт	0x00 (OFF) або 0xFF (ON)
Ім’я та властивості модуля	35 байт	Текст (ASCII)
CRC	2 байта	

Помилка

Адреса пристрою	1 байт	0x01 … 0xF7
Функція (код запиту)	1 байт	0x91
Код помилки	1 байт	0x01, 0x02, 0x03 або 0x04
CRC	2 байта	

КОДИ ПОМИЛОК

0x1 – Помилка функції (ILLEGAL FUNCTION)
Код функції, зазначений в запиті, не є допустимим. Це може бути, наприклад, якщо модуль, що використовується, не підтримує цю функцію, або неправильно налаштований, або в момент опитування знаходиться в стані, що не дозволяє йому обробити такий запит.

0x2 – Помилка адреси даних (ILLEGAL DATA ADDRESS)
Адреса даних не є допустимою. Наприклад, якщо кількість запитаних байт перевищує розмір регістра або запитується неіснуюча адреса.

0x3 – Помилка значення даних (ILLEGAL DATA VALUE)
Значення, що міститься в полі даних запиту, неприпустиме. Показує помилку в структурі складного запиту, наприклад, якщо довжина запиту не відповідає стандарту. Цей код не може показувати, що величина, що посилається для запису в регістр, виходить за межі динамічного діапазону або не має фізичного сенсу, оскільки протокол Modbus не може знати про це.

0x4 – Помилка обробки даних пристроєм
При спробі виконати запит в підпорядкованому пристрої сталася фатальна помилка.