# ЭТАП 1 До поиска периодические запросы (системное сканирование)



telegram\_type | dest\_add | source\_add | function\_code |FCS\_data (CRC)| END

**telegramm\_type:**

#define SD1 0x10 // Телеграмма без поля данных

#define SD2 0x68 // Телеграмма переменной длины

#define SD3 0xA2 // Телеграмма с данными фиксированной длины

#define SD4 0xDC // Токен

#define SC 0xE5 // Квитирование

#define ED 0x16 // Конец

**function\_code:**

***/\* FC Request \*/***

#define FDL\_STATUS 0x09 // SPS: Запрос статуса

#define SRD\_HIGH 0x0D // SPS: Установить выходы, прочитать входы

#define FCV\_ 0x10

#define FCB\_ 0x20

#define REQUEST\_ 0x40

***/\* FC Response \*/***

#define FDL\_STATUS\_OK 0x00 // SLA: OK

#define DATA\_LOW 0x08 // SLA: (Data low) Отправить ввод данных

#define DATA\_HIGH 0x0A // SLA: (Data high) Ожидание диагностики

**ИТОГО:**

Тип телеграммы SD1(Телеграмма без поля данных)

Функциональный код: REQUEST\_+ FDL\_STATUS

Ответ на такой запрос:



Такое же сообщение с замененными адресами

# ЭТАП Сообщение после того, как устройство обнаружено (диагностика)



telegram\_type | PDU\_size | PDU\_size |telegram\_type| dest\_add | source\_add |function\_code| DSAP | SSAP | FCS\_data (CRC)| END

DSAP Destination Service Access Point

SSAP Source Service Access Point

#define SAP\_SET\_SLAVE\_ADR 0x37 // Мастер устанавливает адрес ведомого, ведомый отвечает SC

#define SAP\_RD\_INP 0x38// Мастер запрашивает входные данные, ведомый отправляет входные данные

#define SAP\_RD\_OUTP 0x39// Мастер запрашивает выходные данные, ведомый отправляет выходные данные

#define SAP\_GLOBAL\_CONTROL 0x3A// Master Control, Slave Не отвечает

#define SAP\_GET\_CFG 0x3B// Мастер запрашивает конфигурацию, ведомый отправляет конфигурацию

#define SAP\_SLAVE\_DIAGNOSIS 0x3C// Мастер запрашивает диагностику, ведомый отправляет данные диагностики

#define SAP\_SET\_PRM 0x3D// Мастер отправляет параметры, ведомый отправляет SC

#define SAP\_CHK\_CFG 0x3E// Мастер отправляет конфигурацию, ведомый отправляет SC

**/\* Status Byte 1 \*/**

#define STATION\_NOT\_EXISTENT\_ 0x01

#define STATION\_NOT\_READY\_ 0x02

#define CFG\_FAULT\_ 0x04

#define EXT\_DIAG\_ 0x08 // Доступна расширенная диагностика

#define NOT\_SUPPORTED\_ 0x10

#define INV\_SLAVE\_RESPONSE\_ 0x20

#define PRM\_FAULT\_ 0x40

#define MASTER\_LOCK 0x80

**/\* Status Byte 2 \*/**

#define PRM\_REQ\_ 0x01

#define STAT\_DIAG\_ 0x02

#define STATUS\_2\_DEFAULT 0x04

#define WD\_ON\_ 0x08

#define FREEZE\_MODE\_ 0x10

#define SYNC\_MODE\_ 0x20

//#define free 0x40

#define DEACTIVATED\_ 0x80

**/\* Status Byte 3 \*/**

#define DIAG\_SIZE\_OK 0x01

#define DIAG\_SIZE\_ERROR 0x80

**ИТОГО:**

Тип телеграммы SD2(Телеграмма переменной длины)

Функциональный код: REQUEST\_ + FCB\_ + SRD\_HIGH

DSAP Мастер запрашивает диагностику, ведомый отправляет данные диагностики

Ответ:



function\_code status1 | status2 | status3 |master\_add|ID\_HIGH| ID LOW

function\_code: (Data low) Отправить ввод данных

status1 – STATION\_NOT\_READY Станция не готова

status2 STATUS\_2\_DEFAULT + PRM\_REQ\_

status3: DIAG\_SIZE\_OK (меняется в GSD)

|master\_add : MASTER\_ADD\_DEFAULT

# ЭТАП 3. ПАРАМЕТРИЗАЦИЯ

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Fun |DSAP | SSAP | ID\_HIGH| ID LOW GN ExtUserParams (ИЗ GSD Ext\_User\_Prm\_Data\_Const)

**ИТОГО:**

function\_code REQUEST\_ + FCV\_ + SRD\_HIGH

DSAP Мастер отправляет параметры, ведомый отправляет SC (0xE5 – квитирование)

GN – количество групп. Судя по всему для каждой группы отдельная параметризация

ExtUserParams – не понятно для чего используются.

Ответ на этом этапе простое квитирование

# ЭТАП 4. КОНФИГУРИРОВАНИЕ



direction

На данном этапе происходит настройка модулей. Указывается сколько модулей и каков каждый из них. Тип Ввод/вывод и размер

DSAP - Мастер отправляет конфигурацию, ведомый отправляет SC(0xE5 – квитирование)

Direction – направление

В нашем случае пустой модуль CFG\_SP\_VOID

Здесь должно быть несколько байт, свой для каждого модуля биты в которых означают:

Для стандартного модуля

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 7 бит | 6 бит | 5 бит | 4 бит | 3 бит | 2 бит | 1 бит | 0 бит |
|  | dword | output | input | Размер модуля (words) | | | |

Для спец модулей (если размер модуля иной чем 16 слов (16/32 байт)) на каждый модуль приходится два байта конфигурирования

Для спец модуля 1 байт

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 7 бит | 6 бит | 5 бит | 4 бит | 3 бит | 2 бит | 1 бит | 0 бит |
| output | input | 0 | 0 | Размер модуля, заданный производителем (не 16/32 байт) | | | |

Для спец модуля 2 байт

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 7 бит | 6 бит | 5 бит | 4 бит | 3 бит | 2 бит | 1 бит | 0 бит |
|  | dword | Количество слов в модуле 0 = 1 байт, 1 = 2 байта и т. д. | | | | | |

Ответ на этом этапе простое квитирование