**Описание диагностического протокола CLI профиля ERA-Glonass**

Данные передаются через UART в текстовом виде. Каждое сообщение завершается символом перевода строки (0x10).

Обмен осуществляется по схеме запрос-ответ. Запросы и ответы состоят из последовательности блоков, разделенных пробелами Если пробел требуется передать в составе значения параметра, он экранируется обратным слэшем \.

1й блок в запросе и ответе – всегда сервис (что нужно сделать).

Последующие блоки в запросе определяются сервисом.

2й блок в ответе может быть OK либо NOK, в зависимости от того, удалось ли корректно обработать запрос.

Если запрос не обработан корректно, третий блок ответа содержит код/описание ошибки

Если запрос обработан корректно, 3й и последующий блоки определяются сервисом

Запрос всегда должен сопровождаться ответом. Если ответ не следует в течение 5с после окончания передачи запроса, считается, что связь потеряна.

Если 5с прошло, а обработка запроса еще не завершена, блок должен вернуть негативный ответ с ошибкой PENDING и продолжить обработку запроса

Пример

*WRITE Parameter Value* – запрос

*WRITE NOK PENDING* – промежуточный ответ о продолжении обработки запроса

*WRITE OK Parameter Value* – окончательный ответ о завершении обработки запроса

**Возможны следующие сервисы**

**RESET**

Используется для перезагрузки устройства.

Запрос

2й блок – тип перезагрузки

**HARD** – полная перезагрука (аналог сброса питания)

**SOFT**

3й блок (опционально) – необходимость ответа. В случае отсутствия блока ответ требуется

**NO\_RESP** – ответ не требуется, перезагрузка сразу по получении сообщения

**RESP** – ответ требуется, перезагрузка после отправки ответа

Пример

*RESET HARD* – запрос жесткой перезагрузки с ответом

*RESET OK* – позитивный ответ

*RESET NOK Cond\_Not\_Correct* – негативный ответ

*RESET HARD NO\_RESP* – запрос жесткой перезагрузки, не требующий ответа

**READ**

Используется для чтения значения параметра.

Запрос

2й блок – имя параметра, значение которого необходимо прочитать

Ответ

3й блок – имя запрашиваемого параметра

4й блок – значение запрашиваемого параметра

Пример

*READ GNSS\_Fix* – запрос

*READ OK GNSS\_Fix NO\_FIX* – позитивный ответ

*READ NOK Gnss\_Timeout* – негативный ответ

**WRITE**

Используется для записи значения параметра

Запрос

2й блок – имя параметра, значение которого необходимо записать

3й блок – записываемое значение

Ответ

3й блок – имя запрашиваемого параметра

4й блок – записанное значение

Пример

*WRITE ECall\_On ENABLED* – запрос

*WRITE OK ECall\_On ENABLED* – позитивный ответ

*WRITE NOK Incorrect\_Parameter* – негативный ответ

**CONTROL**

Используется для управления состоянием параметров

Запрос

2й блок - имя параметра, которым хотим управлять

3й блок – тип управления

**SET** – принудительная установка значения. В этом случае 4й блок – значение, которое необходимо установить

**RETURN** – возврат управления TCU и установка значение в соответствии с работой штатной программы

Ответ

3й блок – имя запрашиваемого параметра

4й блок – тип управления. Если SET, то 5й блок – установленное значение

Пример

*CONTROL ECALL\_MODE\_PIN\_Ouput\_Status SET ACTIVE* – запрос

*CONTROL OK ECALL\_MODE\_PIN\_Ouput\_Status SET ACTIVE* – позитивный ответ

*CONTROL NOK Not\_Allowed* – негативный ответ

*CONTROL ECALL\_MODE\_PIN\_Ouput\_Status RETURN* – запрос

*CONTROL OK ECALL\_MODE\_PIN\_Ouput\_Status RETURN* – позитивный ответ

**ROUTINE**

Используется для запуска процедур

Запрос

2й блок – подфункция

**START** – запуск процедуры

**RESULT** – запрос результатов процедуры

3й блок – имя процедуры

4й блок (опционально) – параметры запуска процедуры (используется с подсервисом START)

Ответ

3й блок – запрошенная подфункция

4й блок – имя процедуры

5й блок – результаты выполнения процедуры, если была использована подфункция RESULT

Пример

*ROUTINE START Clear\_GNSS\_Data* – запрос

*ROUTINE OK START Clear\_GNSS\_Data* – позитивный ответ

*ROUTINE NOK Conditions\_Not\_Correct* – негативный ответ

*ROUTINE RESULT Clear\_GNSS\_Data* – запрос

*ROUTINE OK RESULT Clear\_GNSS\_Data Complete* – позитивный ответ

*ROUTINE NOK Not\_Complete* – негативный ответ