**Опис цифрового протоколу обміну інформацією із контролером трубки Піто**

1. **Параметри інтерфейсу:**

Інтерфейс – USART.

Швидкість обміну – 19200 бод.

Режим обміну – двонаправлений. Обмін здійснюється пакетами повідомлень незалежно.

Розрядність – 8 біт, 1 стоп-біт.

Контроль парності – відсутній.

Формат повідомлень – комбінований.

Обмін здійснюється пакетами повідомлень.

Контролер трубки Піто незалежно передає пакети фіксованої довжини 24 байт кожні 25 мсек в режимі Streamin\_mode, в якому контролер знаходиться одразу після старту. Після переходу в command\_mode контролер турбки Піто приймає пакети команд фіксованої довжини (12 байт) і відправляє пакети з відповіддю фіксованої дожини (12 байт).

1. **Структура пакета передачі (24 байт)**

[header] [pressure] [temp\_1] [pressure\_diff] [temp\_2] [altitude] [air\_speed] [temp\_3] [status] [errors] [reserved\_1] [reserved\_2] [crc] [footer]

де,

|  |  |  |  |
| --- | --- | --- | --- |
| Частина фрейму | Розмір в байтах | Тип | Опис |
| header | 2 | uint8\_t[2] | Початок пакету передачі: 0xB3 і 0x39 |
| pressure | 4 | float | Значення з датчика тиску, кПа |
| temp\_1 | 1 | uint8\_t | Температура датчика тиску, градуси Цельсія  ( + 50С для компенсації відʼємних значень) |
| pressure\_diff | 4 | float | Значення з датчика перепаду тиску, кПа |
| temp\_2 | 1 | uint8\_t | Температура датчика перепаду тиску, градуси Цельсія  ( + 50С для компенсації відʼємних значень) |
| altitude | 2 | uint16\_t | Розраховане значення висоти, м |
| air\_speed | 2 | uint16\_t | Розраховане значення швидкості, м/с |
| temp\_3 | 1 | uint8\_t | Температура в трубці, поряд зі спіраллю підігріву, градуси Цельсія  ( + 50С для компенсації відʼємних значень) |
| status | 1 | uint8\_t | Регістр статусу(біт 0 – статус ввімкненості нагрівача, біти 1-7 - резервні) |
| errors | 1 | uint8\_t | Регістр помилок (біт 0 – отримана некоректна команда, біт 1 – помилка CRC в отриманому фреймі, біт 2 – нема даних про температуру біля нагрівача, біт 3 – немає даних перепаду тиску, біт 4 – немає даних тиску) |
| reserved\_1 | 1 | uint8\_t | Резервний байт |
| reserved\_2 | 1 | uint8\_t | Резервний байт |
| crc | 1 | uint8\_8 | Розраховане значення CRC для фрейму |
| footer | 2 | uint8\_t | Кінець пакету передачі: 0xCA і 0xFE |

1. **Структура пакету команд (12 байт):**

[header] [req\_type] [parameter] [payload] [reserved] [crc] [footer]

де,

|  |  |  |  |
| --- | --- | --- | --- |
| Частина фрейму | Розмір в байтах | Тип | Опис |
| header | 2 | uint8\_t[2] | Початок пакету передачі: 0xA5 і 0x5A |
| req\_type | 1 | uin8\_t | Тип запиту:  0x00 – команда,  0x01 – читання,  0x02 – запис. |
| param | 1 | uint8\_t | Параметр команди чи запиту на читання/запис.  Для команд:  0x0A – перейти в режим streaming\_mode  0x0B – перейти в режим command\_mode  0x0C – зберегти конфігурацію у FLASH  0x0F – перезавантажити контролер  Для запитів читання/запису:  0x01 – затримка під час запуску для стабілізації параметрів  0x02 – кількість вимірювання для розрахунку значення калібровки тиску і перепаду тиску  0x03 – включення/вимкнення калібровки тиску  0x04 – включення/вимкнення калібровки перепаду тиску  0x05 – верхня межа температури нагрівача  0х06 – кількість вимірювань тиску і перепаду тиску для усереднення  0x07 – швидкість передачі даних UART |
| payload | 4 | uint8\_t | Параметр для запитів запису, для команд і запитів читання значення payload ігнорується.  Для запитів запису:  2 байти затримки під час запуску для стабілізації параметрів.  2 байт для кількість вимірювання для розрахунку значення калібровки тиску і перепаду тиску.  1 байт включення/вимкнення калібровки тиску (0x00 – вимкнути, 0x01 - включити).  1 байт включення/вимкнення калібровки перепаду тиску (0x00 – вимкнути, 0x01 – включити).  1 байт для верхньої межі температури нагрівача.  1 байт для кількості вимірювань тиску і перепаду тиску для усереднення.  4 байти для швидкості передачі даних UART.  Байти, незадіяні в конфігуруванні ігноруються. |
| reserved | 1 | uint8\_t | Резервний байт |
| crc | 1 | uint8\_8 | Розраховане значення CRC для фрейму |
| footer | 2 | uint8\_t | Кінець пакету передачі: 0xCA і 0xFE |

1. **Структура пакету відповіді на команди (12 байт):**

[header] [req\_type] [parameter] [payload] [reserved] [crc] [footer]

де,

|  |  |  |  |
| --- | --- | --- | --- |
| Частина фрейму | Розмір в байтах | Тип | Опис |
| header | 2 | uint8\_t[2] | Початок пакету передачі: 0xB3 і 0x39 |
| req\_type | 1 | uin8\_t | Тип відповіді:  0x00 – команда,  0x01 – читання,  0x02 – запис. |
| param | 1 | uint8\_t | Параметр команди чи запиту на читання/запис.  Для команд:  0x0A – перехід в режим streaming\_mode  0x0B – перехід в режим command\_mode  0x0C – збереження конфігурації у FLASH  0x0F – перезавантаження контроллера  Для запитів читання/запису:  0x01 – затримка під час запуску для стабілізації параметрів  0x02 – кількість вимірювання для розрахунку значення калібровки тиску і перепаду тиску  0x03 – включення/вимкнення калібровки тиску  0x04 – включення/вимкнення калібровки перепаду тиску  0x05 – верхня межа температури нагрівача  0х06 – кількість вимірювань тиску і перепаду тиску для усереднення  0x07 – швидкість передачі даних UART |
| payload | 4 | uint8\_t | Для команд і запитів запису: 0-байт вказує на статус виконання команди (запису): 0x00 – вдале виконання/запис, 0xFF – невдале виконання/запис.  Для запитів читання:  2 байти затримки під час запуску для стабілізації параметрів.  2 байт для кількості вимірювань для розрахунку значення калібровки тиску і перепаду тиску.  1 байт включення/вимкнення калібровки тиску (0x00 – вимкнена, 0x01 - включена).  1 байт включення/вимкнення калібровки перепаду тиску (0x00 – вимкнена, 0x01 – включена).  1 байт для верхньої межі температури нагрівача.  1 байт для кількості вимірювань тиску і перепаду тиску для усереднення.  4 байти для швидкості передачі даних UART.  Незадіяні в байти заповнюються 0x00. |
| reserved | 1 | uint8\_t | Резервний байт |
| crc | 1 | uint8\_8 | Розраховане значення CRC для фрейму |
| footer | 2 | uint8\_t | Кінець пакету передачі: 0xCA і 0xFE |

1. **Формування CRC**

Формування CRC включає суму всіх байтів окрім header, footer та CRC. Від цієї суми беремо молодший байт. Приклад коду:

uint16\_t eCRC = 0;

for(uint8\_t e=3; e<22; e++) eCRC += eParsBuf[e];

eParsBuf[28] = eCRC&0xFF;