

Spis treści

Wstęp	2
Ogólny opis protokołu Modbus	3
Tryby komunikacji	3
Jednostka danych protokołu Modbus (PDU)	3
Funkcja Kody Modbus	4
Ogólna przestrzeń adresowa protokołu Modbus.....	5
Modbus na magistrali RS485	5
Domyślne ustawienia magistrali RS485	5
Podłączanie pinów przewodów do złącza RJ45 w celu podłączenia do XCONT-HUB	5
Struktura ramki Modbus RTU na magistrali RS485	6
Obliczanie CRC	6
Lista funkcji Modbus	7
Adres przestrzeni Modbus jednostek XCONT-CP	7
Lista rejestrów wejściowych	7
Lista rejestrów przechowywania UCFG	9
Lista rejestrów przechowywania DCFG	10
Bardziej szczegółowy opis znaczenia poszczególnych rejestrów	11
Opis rejestrów wejściowych	11
- act_state 1	11
- act_state 2	11
- act_state 3	12
- AQS_state	12
- AQS_state.....	12
- act_T_ROOM i act_T_WYCIĄG.....	12
Opis rejestrów przechowywania UCFG	13
Panel przedni	13
Opis rejestrów retencji DCFG	14
bity	14
Opis , składnia i przykład zastosowanych funkcji Modbus.....	15
(0x04) Funkcja odczytu rejestrów wejściowych	15
(0x03) Funkcja odczytu rejestrów przechowywania	16
(0x10) Funkcja zapisu wielu rejestrów przechowywania	17

Wstęp

Niniejszy dokument służy do opisu protokołu Modbus używanego w jednostkach centralnych Xhouse (XH) i Xflat (XF). Wersja niniejszego podręcznika jest przeznaczona dla jednostek z wersją oprogramowania układowego 100 i nowszą. Numer wersji oprogramowania układowego jest podany na etykiecie samoprzylepnej na płycie drukowanej oraz w rejestrze wejściowym wersji oprogramowania układowego.

Zacznijmy od kilku przydatnych informacji dotyczących rozwiązywania problemów:
Tylko te rejestry, które są dostępne na dysku, mogą być odczytane z jednostki. W przeciwnym razie jednostka odpowiada błędem z kodem błędu 0x02 - Nielegalny adres danych.

Jednostka potrzebuje pewnej ilości czasu na przetworzenie żądania, dlatego konieczne jest zapewnienie jej wystarczającej ilości czasu na odpowiedź. Czas, jaki musi upłynąć, zanim jednostka odpowie, różni się w zależności od wybranej funkcji modbus i liczby odczytanych/zapisanych rejestrów. Typowy czas odpowiedzi wynosi około 4 ms

W przypadku braku komunikacji między urządzeniem a komputerem należy upewnić się, że wysyłane ramki są poprawne i sprawdzić, czy na magistrali komunikacyjnej występują przerwy trwające co najmniej 4 ms, umożliwiające prawidłowe wykrywanie końców ramek.

Magistrala działa w tzw. trybie Half-duplex. Oznacza to, że nie jest w stanie zaakceptować dalszych żądań, dopóki nie zostanie odebrana odpowiedź na poprzednią ramkę modbus.

Aby sprawdzić lub zweryfikować poprawność obliczeń CRC protokołu Modbus, można skorzystać z kalkulatora online: <https://www.lammertbies.nl/comm/info/crc-calculation.html> Należy przełączyć kalkulator na znaki HEX, a w rezultacie CRC-16 (Modbus) górny i dolny bajt zostaną zamienione w ramce Modbus.

Ogólny opis protokołu Modbus

Protokół Modbus jest protokołem Master-Slave. Na magistrali znajduje się tylko 1 urządzenie master i do 247 urządzeń slave (w naszym przypadku jednostek). Komunikacja jest zawsze inicjowana przez urządzenie master. Urządzenie slave odpowiada tylko na żądania z urządzenia master. Modbus używa reprezentacji danych Big-endian. Oznacza to, że dla elementów powyżej 1 B najwyższy bajt jest wysyłany jako pierwszy, a najniższy bajt później.

Tryby komunikacji

Tryb jednokierunkowy:

Master adresuje jedno konkretne urządzenie podrzędne za pomocą swojego adresu Modbus. Slave przetwarza wiadomość i odpowiada.

Jednostka danych protokołu Modbus (PDU)

Funkcje Modbus 1 B	Dane N* 1 B
-----------------------	----------------

Protokół Modbus definiuje trzy podstawowe typy jednostek PDU:

- 1) **Żądanie PDU** - Służy do adresowania jednego lub większej liczby urządzeń podrzędnych przez urządzenie nadrzędne.
Pole funkcji Modbus zawiera podany kod funkcji Modbus. Następnie pole danych według adresów funkcji Modbus, liczby zmiennych, wartości zmiennych i innych
- 2) **Odpowiedź PDU** - Służy do wysyłania pozytywnej odpowiedzi do urządzeń podrzędnych na otrzymane żądanie PDU.
Pole funkcji Modbus zawiera tę samą wartość, co w otrzymanym Request PDU. Część danych następnie zgodnie z podanymi wartościami operacyjnymi funkcji Modbus, odczytuje wejścia, cewki ...
- 3) **PDU odpowiedzi na wyjątek** - Służy do wysyłania negatywnej odpowiedzi do urządzeń podrzędnych na otrzymane żądanie PDU.
Pole funkcji Modbus zawiera wartość funkcji Modbus z Request PDU + 0x80 jako wskazanie błędu. Część danych identyfikuje następnie błąd.

Kody błędów w jednostce PDU odpowiedzi na wyjątek

Code	Typ kodu funkcji	Znaczenie
0x01	Illegal Modbus function	Wymagana funkcja Modbus nie jest obsługiwana przez serwer (jednostkę)
0x02	Illegal data address	Określony adres (cewki, rejestru ...) jest poza zakresem obsługiwanym przez serwer
0x03	Illegal data value	Podane dane są nieprawidłowe
0x04	Device failure	Wystąpił trwały błąd podczas przetwarzania żądania.
0x05	Confirmation	Kod do wykorzystania w programowaniu. Serwer zgłasza otrzymanie prawidłowego żądania, ale jego wykonanie potrwa dłużej
0x06	Device busy	Kod do wykorzystania w programowaniu. Serwer jest zajęty wykonywaniem długotrwałego polecenia.
0x08	Memory parity failed	Kod do wykorzystania podczas pracy z plikami. Serwer wykrył błąd parzystości podczas próby odczytu pliku
0x0A	Gateway - transmission path unavailable	Kod do pracy z bramą. Brama nie może zarezerwować wewnętrznej ścieżki transmisji z portu wejściowego do portu wyjściowego. Prawdopodobnie jest przeciążona lub nieprawidłowo ustawiona.
0x0B	Gateway - target device does not match	Kod do pracy z bramą. Urządzenie docelowe nie odpowiada, prawdopodobnie nieobecne.

Kody funkcji Modbus

- 1) Publiczne kody funkcji - Są one jasno zdefiniowane i publicznie udokumentowane. Ich unikalność jest gwarantowana. Zawierają również pewne nieużywane kody do wykorzystania w przyszłości.
- 2) Kody funkcji zdefiniowane przez użytkownika - Pozwalają użytkownikowi zaimplementować funkcję, która nie jest zdefiniowana przez protokół. Unikalność kodu nie jest gwarantowana.

Zakresy kodów funkcji Modbus

Kod funkcji	Typ kodu funkcji
1 ... 64	Kody publiczne funkcji
65 ... 72	Kody funkcji zdefiniowane przez użytkownika
73 ... 100	Kody publiczne funkcji
101 ... 110	Kody funkcji zdefiniowane przez użytkownika
111 ... 127	Kody publiczne funkcji

Ogólna przestrzeń adresowa protokołu Modbus

Przestrzeń adresowa protokołu Modbus opiera się na zestawie tabel o charakterystycznych znaczeniach. Zdefiniowano następujące cztery podstawowe tabele:

Tabela	Opis	Uprawnienia	Przestrzeń adresowa (nie jest wymagana)
Wejścia dyskretne	1-bit	Tylko do odczytu	0x2710 to 0x4E1F
Cewki	1-bit	Odczyt i zapis	0x0000 to 0x270F
Rejestry wejściowe	16-bit	Tylko do odczytu	0x7530 to 0x9C3F
Rejestry konserwujące	16-bit	Odczyt i zapis	0x9C40 to 0xC34F

Modbus na magistrali RS485

Domyślne ustawienia magistrali RS485

Protokół Modbus RTU definiuje domyślne ustawienia łącza szeregowego w następujący sposób:

Szybkość transmisji 19200

1 bit startu

8 bitów danych

1 bit parzystości

1 bit stopu

Podane wyżej parametry są domyślnymi ustawieniami urządzenia

Można również ustawić szybkość transmisji 4800, 9600 i brak opcji parzystości.

Podłączenie pinów przewodu do złącza RJ45 w celu podłączenia do XCONT-HUB



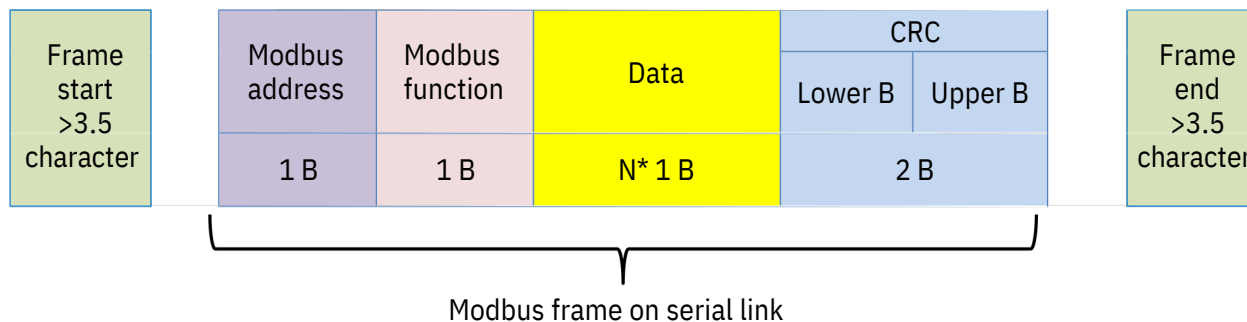
Piny 1, 2, 7 i 8 muszą pozostać niepodłączone

Piny 3, 4 - przewód danych magistrali RS485 A

Piny 5, 6 - przewód danych magistrali RS485 B

Struktura ramki Modbus RTU na magistrali RS485

W trybie Modbus RTU 1 B składa się z dwóch czterobitowych znaków szesnastkowych. Transmisja ramki Modbus zaczyna się i kończy kreską na magistrali dłuższą niż 3,5 znaku. Podczas transmisji ramki odstępy między znakami nie mogą przekraczać 1,5 znaku.



Adresowanie urządzeń podrzędnych:

Adres	Znaczenie
0	Adres rozgłoszeniowy
1 to 247	Indywidualne adresy podrzędne
248 to 255	Zarezerwowane

W ramach Modbus RTU zawierających odpowiedzi przeznaczone dla urządzenia nadrzędnego zachowywany jest adres Modbus odpowiadającego mu urządzenia podrzędnego.

Obliczanie CRC

Obliczenie CRC wykonywane jest dla całej ramki, obejmującej adres urządzenia podrzędnego Modbus, funkcję Modbus i część danych ramki.

1. Inicjalizacja 16-bitowego rejestru CRC na 0xFFFF.
2. Uzupelnij XOR pierwszych 8 bitów ramki niższym bajtem rejestru CRC i zapisz wynik w rejestrze CRC.
3. Przesuń rejestr CRC o 1 b w prawo (w kierunku LSB), wypelnij MSB rejestru CRC wartością 0. Przechwyтуjemy i oceniamy najniższy bit, który został odrzucony podczas przesunięcia.
4. Jeśli ten bit był równy 1, wykonujemy XOR między rejestrem CRC a wartością 0xA001 (generując wielomian = $1+x^2+x^{15}+x^{16}$). Wynik jest ponownie zapisywany w rejestrze CRC.
5. Powtarzaj kroki 3 i 4, aż zostanie wykonanych osiem przesunięć rejestru CRC.
6. Wykonaj operację XOR kolejnych 8 bitów ramki z dolnym bajtem rejestru CRC i powtórz kroki od 3 do 5.
7. Kontynuuj w ten sposób aż do ostatniego bajtu ramki.
8. Wynik obliczenia CRC jest przechowywany w rejestrze CRC.
9. Podczas umieszczania wartości CRC w ramce Modbus, górny i dolny bajt rejestru CRC muszą zostać zamienione (patrz struktura ramki Modbus RTU na łączu szeregowym).

Lista funkcji Modbus			
Oznaczenie	Kod funkcji	Komenda	Public/User
Inp. registers	0x04	Odczyt rejestrów wejściowych	Public
Ret. registers	0x03	Odczyt rejestrów retencji	Public
	0x10	Zapis wielu rejestrów retencji	Public

Adres przestrzeni Modbus jednostek XCONT-CP

Lista rejestrów wejściowych					
Rejestr wejściowy	Adres rejestru	Opis		Format	R/W
FW version	0x7530	Wersja oprogramowania układowego w urządzeniu	-	uint16	R
act_state 1	0x7531	Rejestr stanu 1: bit 0 - 3 - stan interfejsu użytkownika bit 4 - 7 - poprzedni stan interfejsu użytkownika bit 8 - 11 - stan wentylatora bit 12 - 15 - poprzedni stan wentylator	-	uint16	R
act_state 2	0x7532	Rejestr stanu 2: bit 0 - 3 - stan podgrzewania wstępnego bit 4 - 7 - poprzedni stan podgrzewania wstępnego bit 8 - 11 - ostatni stan podgrzewania wstępnego z aktywnymi wentylatorami bit 12 - 15 - etap modyfikacji wentylatora	-	uint16	R
act_state 3	0x7533	Rejestr stanu 3: bit 0 - 2 - stan podgrzewania wstępnego bit 3 - 5 - poprzedni stan podgrzewania wstępnego bit 6 - stan filtra bit 7 - poprzedni stan filtra bit 8 - flaga zmiany stanu interfejsu użytkownika bit 9 - flaga zmiany stanu wentylatora bit 10 - flaga zmiany statusu wstępnego podgrzewania bit 11 - flaga zmiany statusu ponownego podgrzewania bit 12 - flaga zmiany statusu filtra bit 13 - flaga maks. przepiętnienia zgodnie z wymaganiami wentylacji bit 14 - flaga zmiany kierunku ostatniej ochrony przeciwzamrozeniowej bit 15 - nieużywane - uint16 R AQS_state 0x7534	-	uint16	R
AQS_state	0x7534	Rejestr statusu czujnika jakości powietrza: bit 0 - błąd czujnika CO2 bit 1 - błąd czujnika RH bit 2 - aktywne żądanie wentylacji z AQS bit 3 - aktywny poziom CO2 bit 4 - aktywny poziom RH bit 5 - nieużywane bit 6-15 - wymagana prędkość wentylatora zgodnie z wartościami AQS	- - - - - -	uint16	R
AQS_state.fan_flow	0x7535	Wymagana wartość przepływu zgodnie z wartościami AQS	0.01 V 0.1 m3/h	uint16	R

Lista rejestrów wejściowych

Rejestry wejściowe	Adres rejestru	Opis		Format	R/W
set_FAN_FLOW	0x7536	Required fan flow (according to user and AQS settings)	0.1 m ³ /h	uint16	R
act_PID.prop	0x7537	Proportional part of PID controller	1 mV	int16	R
act_PID.integ	0x7538	Integration part of PID controller	1 mV	uint16	R
act_PID.der	0x7539	Derivative part of the PID controller	1 mV	int16	R
act_PID.result	0x753A	PID output and status bits: bit 0 - 9 - PID controller output value bit 10 - PID calculation enable flag bit 11 - PID calculation reset flag bits 12 - 15 - unused	0.01 V - - -	uint16	R
set_FAN_SPEED	0x753B	Required fan voltage (according to user and AQS settings)	0.01 V	uint16	R
set_T_ROOM	0x753C	Required temperature	0.1°C	int16	R
act_CO2	0x753D	Current CO2 sensor value	1 ppm	uint16	R
act_RH	0x753E	RH sensor current value	0.1% RH	uint16	R
act_Flow	0x753F	Current Flow sensor value	0.1 m ³ /h	uint16	R
act_T_ROOM	0x7540	Room temperature sensor: bit 0 - 13 - temperature sensor value bit 14 - 15 - sensor status	0.1°C -	int16	R
act_T_EXHAUST	0x7541	Frost protection temperature sensor: bit 0 - 13 - temperature sensor value bit 14 - 15 - sensor status	0.1°C -	int16	R
FAN1.act_AO	0x7542	Actual fan 1 voltage	-	uint16	R
FAN1.act_tacho	0x7543	Fan 1 feedback: bit 0 - tacho input status bit 1 - successful motor start flag bit 2 - 15 - tacho input error timer	0.01 V 1-OK, 0-NG 1-OK, 0-NG 0.01 s	uint16	R
FAN2.act_AO	0x7544	Actual fan 2 voltage	0.01 V	uint16	R
FAN2.act_tacho	0x7545	Fan 2 feedback: bit 0 - tacho input status bit 1 - successful motor start flag bit 2 - 15 - tacho input error timer Preheat pre-cooling timer:	1-OK, 0-NG 1-OK, 0-NG 0.01 s	uint16	R
timer.act_cool_preheat	0x7546	bit 0 - timer enable bit 1 - 15 - timer value Reheat after-cooling timer:	- 1 s	uint16	R
timer.act_cool_postheat	0x7547	bit 0 - timer enable bit 1 - 15 - timer value Summer mode duration timer:	- 1 s	uint16	R
timer.act_summer_mode	0x7548	bit 0 - timer enable bit 1 - 15 - timer value Boost mode duration timer:	- 1 s	uint16	R
timer.act_boost_mode	0x7549	bit 0 - timer enable bit 1 - 15 - timer value Boost signalling and preheat relay timers:	- 1 s	uint16	R
timers_1	0x754A	bit 0 - enable signalling timer	-	uint16	R

Lista rejestrów wejściowych

Rejestry wejściowe	Adres rejestru	Opis		Format	R/W
		bit 1 - 7 - boost signalling timer value bit 8 - preheat relay timer enable bit 9 - 15 - preheat relay timer value	1 s - 1 s		
timers_2	0x754B	Reheat and AQS protection time relay timers bit 0 - enable the reheat relay timer bit 1 - 7 - value of the reheat relay timer bit 8 - enable the AQS protection time timer bit 9 - 15 - value of the AQS protection time timer	- 1 s - 1 s	uint16	R
act_FILTER_ELAPSED_TIME	0x754C	Filter active time timer	1 h	uint16	R
act_ui_timer	0x754D	Timing of user interface state transitions: bit 0 - timer enable bit 1 - 15 - timer value	- 0.01s	uint16	R
act_preheat_timer High	0x754E	Upper register of the preheat state timer: bit 0 - timer enable bit 1 - 15 - lower 15 bits of the timer value	- 1 s	uint16	R
act_preheat_timer Low	0x754F	Lower register of the preheat timer: bits 0 - 15 - upper 16 bits of the timer value	1 s	uint16	R
act_startup_timer	0x7550	Protection interval timing after device restart: bit 0 - timer enable bit 1 - 15 - timer value	- 1 s	uint16	R
Relay	0x7551	Relay status bits: bit 0 - preheat relay status bit 1 - reheat relay status bits 2 - 15 - unused	- - -	uint16	R

Lista rejestrów retencji UCFG

Ret. rejestr	Adres rejestru	Opis interfejsu		Wartość domyślna	R/W
Front panel	0x9C40	Parameters set on the front panel: bit 0 - flag power on bit 1 - flag AQS auto/manual bit 2 - flag active summer mode bit 3 - flag summer mode auto off bit 4 - flag active boost mode bit 5 - flag touch button lock bit 6 - 9 - fan level bit 10 -15 - temperature level	1-ON/ 0-OFF 1-AU/0-MAN 1-ANO/0-NO 1-ANO/0-NO 1-ANO/0-NO 1-LOCK/0-NO - -	2	R/W R/W R/W R R/W R/W R/W R/W
set_CO2	0x9C41	CO2 value at which the fans switch on (1 ppm)	600-1000	800	R/W
set_RH	0x9C42	RH value at which the fans switch on (0.1% RH)	500-750	650	R/W

Lista rejestrów retencji DCFG

Ret. rejestr	Adres rejestru	Opis	Zakres	Wartość domyślna	R/W
bits	0x9C50	Bit setting items: bit 0 - 2 - not used bit 3 - flag automatic shutdown after reset bit 4 - flag permanently active fans at min. speed bit 5 - 6 - modbus baud rate setting bit 7 - modbus parity setting bit 8 -15 modbus address setting	- 1-YES/0-NO 1-YES/0-NO -	3	R/W R/W R/W
		Room temperature value correction (0.1°C)	1-NONE/0-EVE 1 - 247	0 1	R/W R/W
corr_T_ROOM	0x9C51	Summer mode duration (1 s)	-100 - 100	0	R/W
set_SUMMER_MODE_DURATION	0x9C52		3600 - 32400	28800	R/W
set_BOOST_MODE_FAN_SPEED	0x9C53	Fan voltage in the Boost mode (0.01 V)	XF: 500 - 800 XH: 600 - 900	XF: 800 XH: 900	R/W
set_BOOST_MODE_FAN_FLOW	0x9C54	Fan flow in the Boost mode (0.1m3/h)	XF: xx - xx XH: xx - xx	XF: xx XH: xx	R/W
set_BOOST_MODE_DURATION	0x9C55	Boost mode duration (1 s)	30 - 3600	60	R/W
set_FAN_OFFSET_STATIC	0x9C56	Fan modification settings (%)	0 - 35	0	R/W
set_FILTER_LIFETIME	0x9C57	Filter lifetime (1 h):	2200 - 8800	4400	R/W

Bardziej szczegółowy opis znaczenia poszczególnych rejestrów

Opis rejestrów wejściowych

- act_state 1

bity stanu interfejsu użytkownika i poprzedniego interfejsu użytkownika mogą przyjmować następujące wartości:

- 0 = UI_STATE_OFF – urządzenie wyłączone
- 1 = UI_STATE_OFF_COOLING – urządzenie wyłączone, trwa wstępne lub ponowne nagrzewanie chłodzenia
- 2 = UI_STATE_FAN1_ERROR – błąd wentylatora 1
- 3 = UI_STATE_FAN2_ERROR – błąd wentylatora 2
- 4 = UI_STATE_T_ROOM_ERROR – błąd czujnika pomieszczenia
- 5 = UI_STATE_T_EXHAUST_ERROR – błąd czujnika mrozu
- 6 = UI_STATE_ACTIVE_LOCK – aktywna blokada przycisków dotykowych
- 7 = UI_STATE_ACT_DEACT_LOCK – stan dezaktywacji blokady przycisków dotykowych
- 8 = UI_STATE_SHOW_SETTINGS – stan aktywnego sterownika, wyświetla bieżące ustawienia
- 9 = UI_STATE_SET_FAN – tryb ustawiania zakresu wentylatora
- 10 = UI_STATE_SET_HEAT - tryb ustawiania zakresu temperatur
- 11 = UI_STATE_RUN normalny tryb włączania zasilania, jeśli nie zostanie naciśnięty żaden przycisk
- 12 = UI_STATE_SERVICE_MENU - tryb menu serwisowego
- 13 = UI_STATE_CUSTOMER_MENU - tryb menu użytkownika

Bity stanu wentylatora i poprzednie bity stanu wentylatora mogą przyjmować następujące wartości:

- 0 = FAN_STATE_OFF – urządzenie jest wyłączone
- 1 = FAN_STATE_OFF_COOL – urządzenie jest wyłączone, wentylatory podgrzewają wstępnie lub podgrzewają ponownie
- 2 = FAN_STATE_FAN1_ERROR – błąd wentylatora 1
- 3 = FAN_STATE_FAN2_ERROR – błąd wentylatora 2
- 4 = FAN_STATE_ACTIVE – wentylatory aktywne w trybie normalnym
- 5 = FAN_STATE_ACTIVE_ANTIFREEZE – wentylatory aktywne w trybie przeciwmroźniowym
- 6 = FAN_STATE_INACTIVE – nieaktywne wentylatory w trybie normalnym
- 7 = FAN_STATE_INACTIVE_ANTIFREEZE – nieaktywne wentylatory w trybie przeciwmroźniowym
- 8 = FAN_STATE_SUMMER_MODE – wentylatory w trybie letnim
- 9 = FAN_STATE_BOOST_MODE – wentylatory w trybie Boost

- act_state 2

bity stanu wstępnego nagrzewania, poprzedniego stanu wstępnego nagrzewania i ostatniego stanu wstępnego nagrzewania z aktywnymi wentylatorami mogą przyjmować następujące wartości:

- 0 = PREHEAT_STATE_OFF - urządzenie jest wyłączone
- 1 = PREHEAT_STATE_OFF_COOL, urządzenie jest wyłączone, wstępne nagrzewanie, chłodzenie
- 2 = PREHEAT_STATE_T_ROOM_ERROR - błąd czujnika temperatury w pomieszczeniu
- 3 = PREHEAT_STATE_T_EXHAUST_ERROR - błąd czujnika mrozu
- 4 = PREHEAT_STATE_ACTIVE_LOW - pierwszy poziom ochrony przed mrozem z aktywnym wstępnym nagrzewaniem
- 5 = PREHEAT_STATE_ACTIVE_MED1 - drugi poziom ochrony przed mrozem z modyfikacją wentylatora
- 6 = PREHEAT_STATE_ACTIVE_MED2 - trzeci poziom ochrony przed mrozem z wyłączonym WENTYLATOREM 1
- 7 = PREHEAT_STATE_ACTIVE_HIGH - najwyższy poziom ochrony z wyłączonymi WENTYLATORAMI 1 i 2
- 8 = PREHEAT_STATE_INACTIVE - nieaktywna ochrona przeciwmrozowa, aktywne wentylatory
- 9 = PREHEAT_STATE_INACTIVE_FAN_OFF - nieaktywna ochrona przeciwmrozowa, nieaktywne wentylatory

10 = PREHEAT_STATE_COOL - wstępne podgrzewanie, chłodzenie, wymaganie nieaktywnego wentylatora
11 = PREHEAT_STATE_COOL_TO_MED2 - wstępne podgrzewanie, chłodzenie, gdy FAN 1 jest wyłączony, przejście do MED2
12 = PREHEAT_STATE_COOL_TO_INACTIVE - chłodzenie między stanami LOW i INACTIVE Bity stopni w modyfikacji wentylatora symbolizują różnicę napięć między FAN 1 i FAN 2 w jednostkach 0,5 V.

- act_state 3

bity stanu ponownego nagrzewania i poprzedniego stanu ponownego nagrzewania mogą przyjmować następujące wartości:

0 = POSTHEAT_STATE_OFF — urządzenie jest wyłączone 1 = POSTHEAT_STATE_OFF_COOL — urządzenie jest wyłączone, ponowne nagrzewanie po schłodzeniu
2 = POSTHEAT_STATE_T_ROOM_ERROR — błąd czujnika pomieszczenia
3 = POSTHEAT_STATE_T_EXHAUST_ERROR — błąd czujnika mrozu
4 = POSTHEAT_STATE_ACTIVE — aktywne ponowne nagrzewanie
5 = POSTHEAT_STATE_INACTIVE — nieaktywne ponowne nagrzewanie
6 = POSTHEAT_STATE_COOL — ponowne nagrzewanie po schłodzeniu

- AQS_state

bity 0 i 1 - wskazują stan błędu AQS w przypadku obecności czujnika

bit 2 - wskazuje, że jeden z AQS jest aktywny (aktywuje wentylatory, jeśli włączony jest tryb automatyczny)

bit 3 - aktywny poziom CO₂ - wartość czujnika przekroczyła poziom wymagany do aktywacji wentylatora

bit 4 - aktywny poziom RH - wartość czujnika przekroczyła poziom wymagany do aktywacji wentylatora

bit 5 - nieużywany

bit 6-15 - obliczone zapotrzebowanie na napięcie wentylatorów zgodnie z AQS

- AQS_state. aqs_fan_flow

bit 0-15 - obliczone wymagania dotyczące przepływu wentylatora zgodnie z AQS

- act_T_ROOM and act_T_EXHAUST

bit 0 - 13 - przedstawia wartość danej temperatury

bit 14 - 15 - stan czujnika - 0 = ok, 1 = czujnik odłączony, 2 = czujnik zwarty

Opis rejestrów retencji UCFG

Panel przedni

bit 0 - flaga zasilania włączona - wskazuje, czy jednostka jest włączona czy wyłączona (1 = WŁ., 0 = WYŁ.).
Jednostkę można włączyć lub wyłączyć zdalnie poprzez zapis.

bit 1 - flaga AQS auto/manual - wskazuje aktualnie wybrany tryb wentylatora (1 = automatyczny, zgodnie z AQS, 0 = ręczny). Możesz zmienić tryb, rejestrując się tutaj.

bit 2 - flaga aktywnego trybu letniego - wskazuje aktywny tryb letni (1 = tryb letni aktywny, 0 = tryb letni nieaktywny). Tryb letni można aktywować/dezaktywować poprzez zapis (jeśli spełnione są warunki aktywacji)

bit 3 - flaga trybu letniego auto off - sygnalizuje automatyczne zakończenie trybu letniego. Nieużywany do zapisu. Podczas zapisu pozostaw ustawioną bieżącą wartość

bit 4 - flaga aktywnego trybu wzmocnienia - wskazuje aktywny tryb wzmocnienia (1 = aktywny tryb wzmocnienia, 0 = nieaktywny tryb wzmocnienia). Zapis może aktywować/dezaktywować (jeśli spełnione są wymagania)

bit 5 - flaga blokady przycisku dotykowego - wskazuje aktywną blokadę „dziecka” (1 = blokada przycisku aktywna, 0 = blokada przycisku nieaktywna). Zapis może się zmienić

bit 6 - 9 - poziom poziomu wentylatora - wskazuje aktualnie wybrany poziom wentylatora. Zapis może się zmienić. Nie ustawiaj wartości 8 dla trybu Boost. Bit 4 aktywuje tryb Boost

Value of bits 6-9	XFLAT150 fans voltage	XHOUSE300 fans voltage
0	0 V	0 V
1	2 V	2 V
2	2.8 V	3 V
3	3.7 V	4 V
4	4.5 V	5 V
5	5.3 V	6 V
6	6.2 V	7 V
7	7 V	8 V

bit 10 -15 - Krok poziomu temperatury - Dla jednostek typu E wskazuje aktualnie wybrany poziom temperatury. Dane można zmienić.

Opis bitów rejestrów retencji DCFG

bity

bit 0 - 2 - nieużywany - Podczas zapisu zamień na 0

bit 3 - flaga automatycznego wyłączenia po resecie - wskazuje, czy jednostka wyłączy się automatycznie lub przywróci poprzedni stan w przypadku nieoczekiwanego resetu. (1 = automatyczne wyłączenie, 0 = poprzedni stan). Poprzez zapis można zmienić

bit 4 - flaga stale aktywnych wentylatorów na min. prędkość - wskazuje tryb, w którym wentylatorów nie można wyłączyć. Wentylatory zawsze pracują z minimalną prędkością (1 = wentylatory zawsze z minimalną prędkością, 0 = wentylatory wyłączone). Zapis może się zmienić.

bit 5 - 6 - ustawienie szybkości transmisji modbus - ustawienie szybkości transmisji modbus (0 = wartość wyłączona, 1 = 4800, 2 = 9600, 3 = 19200). Zapis może się zmienić.

bit 7 - ustawienie parzystości modbus - wskazuje ustawienie parzystości modbus (0 = parzystość parzysta, 1 = brak parzystości). Zapis może się zmienić.

bity 8-15 ustawienia adresu modbus - określają aktualny adres modbus urządzenia.

W przypadku bitów od 5 do 15, w przypadku zapisu, jednostka natychmiast zaczyna zachowywać się zgodnie z nowymi parametrami. Dlatego, jeśli którykolwiek z parametrów zostanie zmieniony, jednostka zazwyczaj przestanie komunikować się, dopóki master sieci nie zmieni parametrów.

Opis, składnia i przykład funkcji Modbus używanych (0x04) Funkcja odczytu rejestrów wejściowych

Ta funkcja służy do odczytu zawartości ciągłego bloku rejestrów wejściowych. Żądanie określa adres pierwszego rejestru i liczbę rejestrów. W odpowiedzi każdy rejestr odpowiada parze bajtów.

1) Żądanie PDU

Modbus function	First register address	Number of registers
0x04	see List of int. reg.	1 to max. 13
1 B	2 B	2 B

Przykład odczytu rejestrów wejściowych act_CO2 i act_RH:

0x04	act_CO2	Number of registers =
	0x75 0x37	2
		0x00 0x02

2) Odpowiedź PDU

Modbus function	Number of bytes	Register statuses
0x04	2*N	
1 B	1 B	2*N B

N = Liczba rejestrów (sprawdź Żądanie PDU)

Przykład odpowiedzi na odczyt rejestrów wejściowych act_CO2 i act_RH:

0x04	Number of bytes	CO2 = 980 ppm	RH = 33.5 %
	0x04	0x03 0xB4	0x01 0x4F

3) PDU odpowiedzi na wyjątek

Modbus function 0x80	Error code
0x84	1, 2, 3 or 4
1 B	1 B

(0x03) Funkcja odczytu rejestrów retencji

Ta funkcja służy do odczytu zawartości ciągłego bloku rejestrów retencji. Żądanie określa adres pierwszego rejestru i liczbę rejestrów. W odpowiedzi każdy rejestr odpowiada parze bajtów.

1) Żądanie PDU

Modbus function	First register address	Number of registers
0x03	See List of ret. reg.	1 to max. 13
1 B	2 B	2 B

Przykład odczytu rejestrów retencji set_CO2 i set_RH:

0x03	Set_CO2	Number of registers = 2
	0x9C 0x41	0x00 0x02

2) Odpowiedź PDU

Modbus function	Number of bytes	Register statuses
0x03	2*N	
1 B	1 B	2*N B

N = Liczba rejestrów (sprawdź Żądanie PDU)

Przykładowa odpowiedź na odczyt rejestrów retencji set_CO2 i set_RH:

0x03	Number of bytes	CO2=750	RH=550
	0x06	0x02 0xEE	0x02 0x26

3) PDU odpowiedzi na wyjątek

Modbus function 0x80	Error code
0x83	1, 2, 3 or 4
1 B	1 B

(0x10) Funkcja zapisu wielu rejestrów retencji

Ta funkcja służy do zapisu ciągłego bloku rejestrów retencji. Żądanie określa adres pierwszego rejestru, który ma zostać zapisany, liczbę rejestrów i wartości, które mają zostać zapisane. Normalna odpowiedź zawiera adres początkowy i liczbę zapisanych rejestrów.

1) Żądanie PDU

Modbus function 0x10 1 B	First register address See List of ret. reg. 2 B	Number of registers 1 to max. 11 2 B	Number of bytes 2*N 1 B	Register statuses 2*N B
-----------------------------	--	--	-------------------------------	----------------------------

N = Liczba rejestrów

Przykład zapisu rejestrów retencji set_FILTER_LIFE_TIME

0x10	Set_FIL_LT		Number of registers = 1		Number of bytes = 2		8800 h	
	0x9C	0x55	0x00	0x01	0x04		0x22	0x60

2) Odpowiedź PDU

Modbus function 0x10 1 B	First register address See Request PDU 2 B	Number of registers See Request PDU 2 B
--------------------------------	--	---

N = Liczba rejestrów

Przykład odpowiedzi na zapis rejestrów retencji set_FILTER_LIFE_TIME:

0x10	Set_FIL_LT		Number of registers = 1	
	0x9C	0x55	0x00	0x01

3) PDU odpowiedzi na wyjątek

Modbus function 0x80 0x90 1 B	Error code 1, 2, 3 or 4 1 B
--------------------------------------	-----------------------------------