

REGULATOR TEMPERATURY CIEPŁEJ  
WODY UŻYTKOWEJ  
**RG24**



INSTRUKCJA OBSŁUGI INTERFEJSU



---

**Spis treści:**

---

<b>1. Wstęp.....</b>	<b>3</b>
<b>2. Sposób podłączenie interfejsu szeregowego.....</b>	<b>3</b>
<b>3. Implementacja protokołu LUMBUS.....</b>	<b>4</b>
3.1 Opis indeksów regulatora RG24.....	4
3.2 Zdalne sterowanie przekaźnikami.....	10
3.3 Zapisywanie programu dobowego dla temperatur zadanych.....	11
3.4 Zapisywanie programu dobowego działania pompy cyrkulacyjnej.....	12
3.5 Zapisywanie dni dodatkowych.....	12
3.6 Zdalne wpisanie nastaw fabrycznych.....	12
3.7 Zdalny restart regulatora.....	12
<b>4. Specyfikacja protokołu LUMBUS.....</b>	<b>12</b>
4.1 Podstawowe pojęcia.....	12
4.2 Warstwa fizyczna.....	13
4.3 Warstwa transmisji.....	14
4.4 Warstwa aplikacji.....	17
<b>5. Implementacja protokołu MODBUS.....</b>	<b>19</b>
5.1 Opis rejestrów regulatora RG24.....	19
5.2 Zdalne sterowanie przekaźnikami.....	25
5.3 Zapisywanie programu dobowego dla temperatur zadanych.....	26
5.4 Zapisywanie programu dobowego działania pompy cyrkulacyjnej.....	26
5.5 Zapisywanie dni dodatkowych.....	26
5.6 Zdalne wpisanie nastaw fabrycznych.....	26
5.7 Zdalny restart regulatora.....	26
<b>6. Specyfikacja protokołu MODBUS.....</b>	<b>27</b>
6.1 Ramka w trybie ASCII.....	27
6.2 Ramka w trybie RTU.....	28
6.3 Charakterystyka pól ramki.....	28
6.4 Wyznaczenie LRC.....	28
6.5 Wyznaczenie CRC.....	29
6.6 Format znaku przy transmisji szeregowej.....	29
6.7 Przerwanie transakcji.....	29
6.8 Opis funkcji.....	29
6.9 Kody błędów.....	31
6.10 Dodatek A: Obliczanie sum kontrolnych LRC i CRC.....	32

## 1. Wstęp.

Mikroprocesorowy regulator temperatury RG24 ma łącze szeregowe w standardzie RS-485 do komunikacji w systemach komputerowych oraz z innymi urządzeniami. Na łączu szeregowym zostały zaimplementowane 2 asynchroniczne protokoły transmisji: firmowy protokół Lumelu LUMBUS oraz protokół MODBUS w trybach ASCII i RTU. Wybór i konfiguracja protokołu następuje z klawiatury regulatora w sposób opisany w *Instrukcji obsługi regulatora RG24*.

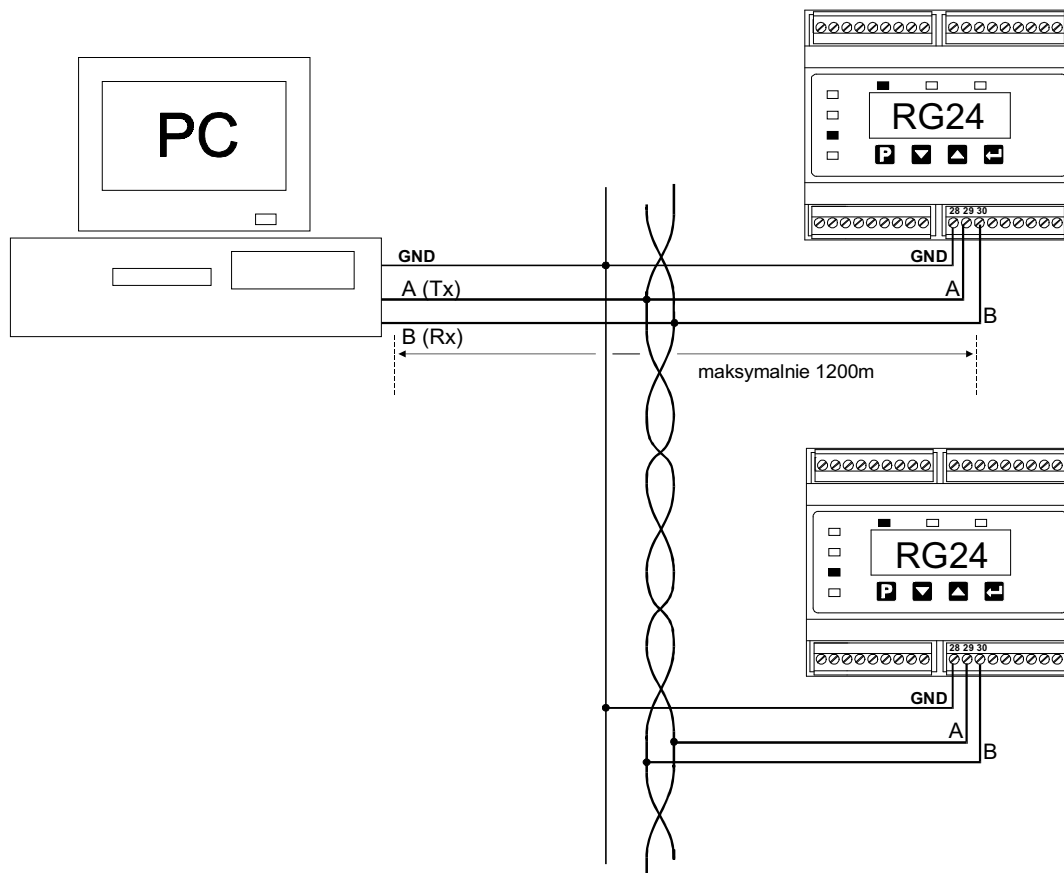
Regulator RG24 ma zmodernizowane i rozbudowane oprogramowanie częściowo bazujące na oprogramowaniu regulatora RG21. W porównaniu do regulatora RG21 dodano kilka nowych funkcji oraz zmieniono działanie innych. Większość nowych parametrów zostało dołączonych na końcu pierwotnej listy indeksów (adresów) regulatora RG21. Kilka parametrów zmieniło zakres wartości. Wszystkie nowe lub zmienione parametry zostały zacieniowane w tablicach indeksów (LUMBUS) i rejestrów (MODBUS). Więcej informacji o oprogramowaniu znajduje się w *Instrukcji obsługi regulatora RG24*.

## 2. Sposób podłączenie interfejsu szeregowego.

Standard RS-485 pozwala na bezpośrednie połączenie do 32 urządzeń na pojedynczym łączu szeregowym o długości do 1200m. Do połączenia większej ilości urządzeń konieczne jest stosowanie dodatkowych układów pośrednicząco - separujących.

Wyprowadzenie linii interfejsu podano w instrukcji obsługi regulatora. Do uzyskania prawidłowej transmisji konieczne jest podłączenie parą skręconych przewodów linii **A** i **B** równoległe z ich odpowiednikami w innych urządzeniach. Ze względu na możliwość występowania różnic potencjału między nadajnikami RS-485 poszczególnych urządzeń, w niektórych przypadkach znacznie przekraczających dopuszczalne wartości dla tych układów, może okazać się konieczne połączenie ze sobą mas interfejsów oznaczonych symbolem **GND**. Połączenia należy wykonać przewodem ekranowanym. Ekran należy podłączyć do zacisku ochronnego instalacji elektrycznej w pojedynczym punkcie.

Sposób łączenia urządzeń pokazano na rysunku rys. 1.



Rys. 1. Sposób łączenia urządzeń.

Do uzyskania połączenia z komputerem PC niezbędna jest karta interfejsu RS 485 lub konwerter RS232 na RS485 z automatycznym sterowaniem kierunkiem transmisji (np. PD5 produkcji Lumel S.A.).

### 3. Implementacja protokołu LUMBUS.

#### 3.1 Opis indeksów regulatora RG24.

Regulator RG24 został zdefiniowany jako urządzenie pasywne o numerze identyfikacyjnym 15 i dodatkowym 3. Wszystkie indeksy regulatora reprezentują zmienne 16 bitowe typu prostego. Wartość wpisywana pod dany indeks interpretowana jest w zależności od jego numeru: jako dwubajtowa liczba całkowita ze znakiem (*int*), jako dwie jednobajtowe liczby bez znaku (MSB, LSB) lub jako dwa bajty zawierające maskę i znaczniki (MSB, LSB). Oznaczenia {R-}, {-W} w kolumnie *Opcje* oznaczają dopuszczalne działania na zmiennych - odpowiednio odczyt i zapis. W przypadku indeksów do zmiennych interpretowanych bitowo, maska umożliwia dostęp do wybranych bitów danej – ustawienie bitu maski na „1” umożliwi dokonanie operacji na odpowiadającym mu bicie danej bez zmiany innych bitów. Wynikiem odczytu maski jest zawsze wartość "0". Oznaczenie MSB, LSB w kolumnie *Format danej* określają odpowiednio: starszy bajt i młodszy bajt danej.

Zestawienie indeksów regulatora RG24

Tablica 1.

Indeks	Symbol	Opis parametru	Zakres	Format danej	Jednostka	Opcje
9		zestaw parametrów określających stan regulatora	patrz → tablica 2.			{R-}
<i>Parametry regulacji PID:</i>						
10	<i>Pb</i>	zakres proporcjonalności	1..999	int	[%]	{RW}
11	<i>t</i>	czas całkowania	0..9999	int	[sek]	{RW}
12	<i>tđ</i>	czas różniczkowania	0..999	int	[sek]	{RW}
13	<i>đUn</i>	minimalna zmiana położenia zaworu	1..100	int	[% x 10]	{RW}
14	<i>zn</i>	strefa nieczułości	0..20	int	[°C]	{RW}
<i>Zegar i kalendarz hhmm dRtR</i>						
15		rok	2001..2099	int		{RW}
16		dzień:miesiąc	1..31, 1..12	int [MSB,LSB]	[dd,mm]	{RW}
17	<i>dt 1-7</i>	dzień tygodnia	0..6	int	pon-niedz	{R-}
18	<i>hhmm</i>	godzina:minuta	0..23,0..59	int [MSB,LSB]	[gg,mm]	{RW}
<i>Dni dodatkowe</i>						
19 : 38	<i>dd</i>	dodatkowe dni robocze	1..31, 1..12 0, 0 – brak	int [MSB,LSB]	[dd,mm]	{RW}
39 : 58	<i>dd</i>	dodatkowe dni świąteczne	1..31, 1..12 0, 0 – brak	int [MSB,LSB]	[dd,mm]	{RW}
59 : 78	<i>dd</i>	dni specjalne	1..31, 1..12 0, 0 – brak	int [MSB,LSB]	[dd,mm]	{RW}
<i>Bloki funkcyjne bLo</i>						
79	<i>bLo1</i> <i>bLo2</i> <i>bLo3</i> <i>bLo4</i>	MSB: maska LSB: <b>bit0</b> = blok1 <b>bit1</b> = blok2 <b>bit2</b> = blok3 <b>bit3</b> = blok4	0..0Fh, 0..0Fh	int [MSB,LSB]	<b>0</b> = wyłączony <b>1</b> = załączony	{RW}
80		zmiana programu dobowego, która będzie obowiązywała do końca dnia kalendarzowego (odpowiada „szybkiej zmianie programu” przyciskiem <input type="checkbox"/> ).	0..3	int	<b>0</b> = bez zmian <b>1</b> = roboczy <b>2</b> = świąteczny <b>3</b> = specjalny	{RW}

c.d. tablicy 1.

Programy dobowe dla temperatur zadanych $SP_{c.u}$ (cz. 1):						
Program dobowy roboczy dla temperatur zadanych: $SP$						
81	$tSP1$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
82	$SP1$	poprawka dobowa	-50..50	int	[°C]	{RW}
83	$tSP2$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
84	$SP2$	poprawka dobowa	-50..50	int	[°C]	{RW}
85	$tSP3$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
86	$SP3$	poprawka dobowa	-50..50	int	[°C]	{RW}
87	$tSP4$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
88	$SP4$	poprawka dobowa	-50..50	int	[°C]	{RW}
89	$tSP5$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
90	$SP5$	poprawka dobowa	-50..50	int	[°C]	{RW}
91	$tSP6$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
92	$SP6$	poprawka dobowa	-50..50	int	[°C]	{RW}
Program dobowy świąteczny dla temperatur zadanych: $SP$						
93	$tSP1$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
94	$SP1$	poprawka dobowa	-50..50	int	[°C]	{RW}
95	$tSP2$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
96	$SP2$	poprawka dobowa	-50..50	int	[°C]	{RW}
97	$tSP3$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
98	$SP3$	poprawka dobowa	-50..50	int	[°C]	{RW}
99	$tSP4$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
100	$SP4$	poprawka dobowa	-50..50	int	[°C]	{RW}
101	$tSP5$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
102	$SP5$	poprawka dobowa	-50..50	int	[°C]	{RW}
103	$tSP6$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
104	$SP6$	poprawka dobowa	-50..50	int	[°C]	{RW}
Program dobowy specjalny dla temperatur zadanych: $SP$						
105	$tSP1$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
106	$SP1$	poprawka dobowa	-50..50	int	[°C]	{RW}
107	$tSP2$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
108	$SP2$	poprawka dobowa	-50..50	int	[°C]	{RW}
109	$tSP3$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
110	$SP3$	poprawka dobowa	-50..50	int	[°C]	{RW}
111	$tSP4$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
112	$SP4$	poprawka dobowa	-50..50	int	[°C]	{RW}
113	$tSP5$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
114	$SP5$	poprawka dobowa	-50..50	int	[°C]	{RW}
115	$tSP6$	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
116	$SP6$	poprawka dobowa	-50..50	int	[°C]	{RW}
Parametry priorytetu C.W.U. (cz. 1.) $Prcu$						
117	$SPPr$	wartość temperatury T1, poniżej której zadziała priorytet c.w.u.	5..95	int	[°C]	{RW}
Parametry pracy pompy cyrkulacyjnej $PPcy$						
118	$SPc1$	zadana wartość temperatury T1 do sterowania pompą cyrkulacyjną	5..95	int	[°C]	{RW}
119	$SPc2$	zadana wartość temperatury T2 wody w przewodzie cyrkulacyjnym	5..95	int	[°C]	{RW}

c.d. tablicy 1.

120	h . 12	histereza dla wartości zadanych <i>SPc 1 i SPc 2</i>	0..200	int	[°C x 10]	{ RW }
121	ł r y b	tryb sterowania pompą cyrkulacyjną	0..3	int	0 = tryb 1 1 = tryb 2 2 = tryb 3 3 = tryb 4	{ RW }
<b>Program roboczy działania pompy cyrkulacyjnej: c 3</b>						
122	c y 1..6	odblokowanie lub wstrzymanie działania pompy cyrkulacyjnej MSB: maska LSB: <b>bit0</b> = cy1 <b>bit1</b> = cy2 <b>bit2</b> = cy3 <b>bit3</b> = cy4 <b>bit4</b> = cy5 <b>bit5</b> = cy6	0..3Fh, 0..3Fh	int [MSB,LSB]	0 = wyłączony 1 = załączony	{ RW }
123	ł c y 1	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
124	ł c y 2	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
125	ł c y 3	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
126	ł c y 4	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
127	ł c y 5	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
128	ł c y 6	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
<b>Program święteczny działania pompy cyrkulacyjnej: c 3</b>						
129	c y 1..6	odblokowanie lub wstrzymanie działania pompy cyrkulacyjnej MSB: maska LSB: <b>bit0</b> = cy1 <b>bit1</b> = cy2 <b>bit2</b> = cy3 <b>bit3</b> = cy4 <b>bit4</b> = cy5 <b>bit5</b> = cy6	0..3Fh, 0..3Fh	int [MSB,LSB]	0 = wyłączony 1 = załączony	{ RW }
130	ł c y 1	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
131	ł c y 2	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
132	ł c y 3	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
133	ł c y 4	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
134	ł c y 5	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
135	ł c y 6	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
<b>Program specjalny działania pompy cyrkulacyjnej: c 3</b>						
136	c y 1..6	odblokowanie lub wstrzymanie działania pompy cyrkulacyjnej MSB: maska LSB: <b>bit0</b> = cy1 <b>bit1</b> = cy2 <b>bit2</b> = cy3 <b>bit3</b> = cy4 <b>bit4</b> = cy5 <b>bit5</b> = cy6	0..3Fh, 0..3Fh	int [MSB,LSB]	0 = wyłączony 1 = załączony	{ RW }

c.d. tablicy 1.

137	ł c y 1	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
-----	---------	--	--------------	---------------	---------	--------

138	<b>tcY2</b>	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
139	<b>tcY3</b>	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
140	<b>tcY4</b>	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
141	<b>tcY5</b>	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
142	<b>tcY6</b>	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Parametry pracy pompy ładującej PPLR</b>						
143	<b>SPL3</b>	zadana wartość temperatury T3 w górnej części zasobnika ciepłej wody	5..95	int	[°C]	{RW}
144	<b>SPL4</b>	zadana wartość temperatury T4 w dolnej części zasobnika ciepłej wody	5..95	int	[°C]	{RW}
145	<b>h.34</b>	histereza dla wartości zadanych <b>SPL3</b> i <b>SPL4</b>	0..200	int	[°C x 10]	{RW}
<b>Program roboczy wymuszenia załączeń pompy obiegowej: LR</b>						
146	<b>ELR1</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
147	<b>ELR2</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
148	<b>ELR3</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
149	<b>ELR4</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
150	<b>ELR5</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
151	<b>ELR6</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Program świąteczny wymuszenia załączeń pompy obiegowej: LR</b>						
152	<b>ELR1</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
153	<b>ELR2</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
154	<b>ELR3</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
155	<b>ELR4</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
156	<b>ELR5</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
157	<b>ELR6</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Program specjalny wymuszenia załączeń pompy obiegowej: LR</b>						
158	<b>ELR1</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
159	<b>ELR2</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
160	<b>ELR3</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
161	<b>ELR4</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
162	<b>ELR5</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
163	<b>ELR6</b>	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Różnice temperatur dla czujników rŁ</b>						
164	<b>rŁ1</b>	różnica temperatury dla T1	-9.9..10.0	int	[°C x 10]	{RW}
165	<b>rŁ2</b>	różnica temperatury dla T2	-9.9..10.0	int	[°C x 10]	{RW}
166	<b>rŁ3</b>	różnica temperatury dla T3	-9.9..10.0	int	[°C x 10]	{RW}
167	<b>rŁ4</b>	różnica temperatury dla T4	-9.9..10.0	int	[°C x 10]	{RW}
<b>Konfiguracja czujników CONF</b>						
168	<b>Ł2</b> <b>Ł3</b> <b>Ł4</b>	MSB: maska LSB: <b>bit0</b> = czujnik T2 <b>bit1</b> = czujnik T3 <b>bit2</b> = czujnik T4	0..07h, 0..07h	int [MSB, LSB]	<b>0</b> = brak czujnika <b>1</b> = jest	{RW}

c.d. tablicy 1.

169	$t_{p0}$	czas pełnego otwarcia zaworu. <i>Zmiana wartości parametru spowoduje pozycjonowanie napędu zaworu</i>	8..360	int	[sek]	{RW}
170	cod	kod bezpieczeństwa	0..9999	int		{RW}
171		numer wersji programu		int		{R-}
172		1-słowo statusu regulatora	<i>patrz → tablica 3.</i>			{R-}
173		2-słowo statusu regulatora	<i>patrz → tablica 4.</i>			{R-}
174	$t_1$	temperatura T1 na wyjściu wymiennika	-500..1500	int	[°C x 10]	{R-}
175	$t_2$	temperatura T2 wody w przewodzie cyrkulacyjnym	-500..1500	int	[°C x 10]	{R-}
176	$t_3$	temperatura T3 wody w górnej części zasobnika	-500..1500	int	[°C x 10]	{R-}
177	$t_4$	temperatura T4 wody w dolnej części zasobnika	-500..1500	int	[°C x 10]	{R-}
178		aktualna wartość zadana temperatury T1 wody na wyjściu wymiennika	50..950	int	[°C x 10]	{R-}
179		położenie zaworu	0..10000	int	[%·100]	{R-}
180		stan regulacji automatycznej	0..1	int	<b>0</b> = STOP <b>1</b> = START	{RW}
<b>Zdalne sterowanie elementami wykonawczymi:</b>						
181	$S_{t-r-r}$	zdalne sterowanie elementami wykonawczymi	0..3, 0..3	int [MSB, LSB] MSB: maska LSB: <b>bit0:</b> tryb sterowania elementami wykonawczymi <b>bit1:</b> tryb sterowania przekaźnikami otwierania i zamykania	<b>0</b> = automat. <b>1</b> = zdalne  <b>0</b> = sterow. przez indeks 182 <b>1</b> = sterow. przez indeks 183	{RW}
182		nowe położenie siłownika zaworu w czasie zdalnego sterowania	0..10000	int	[% x 100]	{RW}
183		stan przekaźników w czasie zdalnego sterowania	0..0Fh, 0..0Fh	int [MSB, LSB] MSB: maska LSB: <b>bit0</b> = P1 <b>bit1</b> = P2 <b>bit2</b> = P3 <b>bit3</b> = P4	<b>0</b> = wyłączony <b>1</b> = załączony	{RW}
184	$F_{R-br}$	przywrócenie nastaw fabrycznych (oprócz ustawień interfejsu)	AA55h Przy odczycie zwracana jest wartość 0	int		{RW}
185		aktualna wartość temperatury T1 wody na wyjściu wymiennika, przy której zadziała priorytet ciepłej wody	50..950	int	[°C x 10]	{R-}
186		aktualna wartość zadana temperatury T1 do sterowania pompą cyrkulacyjną	50..950	int	[°C x 10]	{R-}
187		aktualna wartość zadana temperatury T2 wody w przewodzie cyrkulacyjnym	50..950	int	[°C x 10]	{R-}

c.d. tablicy 1.



188		aktualna wartość zadana temperatury T3 wody w górnej części zasobnika	50..950	int	[°C x 10]	{R-}
189		aktualna wartość zadana temperatury T4 wody w dolnej części zasobnika	50..950	int	[°C x 10]	{R-}
190	<b>ŁPc</b>	czas pełnego zamknięcia zaworu. <i>Zmiana wartości parametru spowoduje pozycjonowanie napędu zaworu</i>	8..360	int	[sek]	{RW}
<b>Parametry priorytetu C.W.U. (cz. 2) Prcu</b>						
191	<b>h,Pr</b>	histereza działania priorytetu c.w.u.	0..200	int	[°C x 10]	{RW}
<b>Programy dobowe dla temperatur zadanych SPcu (cz. 2):</b>						
192	<b>SPcc</b>	zadana wartość temperatury T1 na wyjściu z wymiennika	5..95	int	[°C]	{RW}
193	<b>ŁcuH</b>	największa wartość, jaką mogą przyjąć wszystkie temperatury zadane	30..95	int	[°C]	{RW}
194	<b>ŁcuL</b>	najmniejsza wartość, jaką mogą przyjąć wszystkie temperatury zadane	5..30	int	[°C]	{RW}
<b>Parametry funkcji dezynfekcji instalacji dF c</b>						
195	<b>Er.dF</b>	status dezynfekcji	0..5	int		{RW}
196	<b>SP.dF</b>	wartość zadana temperatury T1 podczas dezynfekcji	60..95	int	[°C]	{RW}
197	<b>dŁ.dF</b>	dopuszczalne obniżenie temperatury T1 bez wstrzymania zliczania efektywnego czasu dezynfekcji	0..20	int	[°C]	{RW}
198	<b>cr.dF</b>	godzina rozpoczęcia dezynfekcji	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
199	<b>c.dF</b>	efektywny czas dezynfekcji	30..120	int	[minuty]	{RW}
200	<b>cŁ.dF</b>	maksymalny czas na przeprowadzenie dezynfekcji	60..240	int	[minuty]	{RW}
201	<b>dn,</b>	wyłączenie lub załączenie dezynfekcji w kolejne dni tygodnia: MSB: maska LSB: <b>bit0</b> = poniedziałek <b>bit1</b> = wtorek <b>bit2</b> = środa <b>bit3</b> = czwartek <b>bit4</b> = piątek <b>bit5</b> = sobota <b>bit6</b> = niedziela	0..7Fh, 0..7Fh	int [MSB,LSB]	<b>0</b> = wyłączona <b>1</b> = załączona	{RW}

Odczyt indeksu 9 powoduje przesłanie 14 słów następujących danych :

Indeks 9 {R-} : zestaw parametrów określających stan regulatora:

Tablica 2.

słowo	opis	indeks
1	identyfikator urządzenia	-
2	1 słowo statusu (tablica 3.)	172
3	2 słowo statusu (tablica 4.)	173
4	temperatura T1 wody na wyjściu wymiennika	174
5	temperatura T2 wody w przewodzie cyrkulacyjnym	175
6	temperatura T3 wody w górnej części zasobnika	176
7	temperatura T4 wody w dolnej części zasobnika	177
8	aktualna wartość zadana temperatury T1 wody na wyjściu wymiennika	178
9	położenie zaworu (sygnał sterujący Un)	179
10	aktualna wartość temperatury T1, przy której zadziała priorytet ciepłej wody	185
11	aktualna wartość zadana temperatury T1 do sterowania pompą cyrkulacyjną	186

12	aktualna wartość zadana temperatury T2 wody w przewodzie cyrkulacyjnym	187
13	aktualna wartość zadana temperatury T3 wody w górnej części zasobnika	188
14	aktualna wartość zadana temperatury T4 wody w dolnej części zasobnika	189

Opis znaczników statusu regulatora (indeksy 172 i 173):

Indeks 172 { R- } : 1 słowo statusu regulatora:

Tablica 3.

bit	Opis
15	regulator pracuje w TRYBIE AWARYJNYM (błąd pamięci EEPROM)
14	czas i dane kalendarza są nieaktualne
13	temperatura T1 poza zakresem pomiarowym ( <i>RL - 1</i> )
12	temperatura T2 poza zakresem pomiarowym ( <i>RL - 2</i> )
11	temperatura T3 poza zakresem pomiarowym ( <i>RL - 3</i> )
10	temperatura T4 poza zakresem pomiarowym ( <i>RL - 4</i> )
9	stan wejścia binarnego (1=zwarte, 0=rozwarne)
8	siłownik w trakcie pozycjonowania
7	regulator w trybie regulacji automatycznej ( <i>St r t</i> )
6	regulator w trybie zdalnego sterowania ( <i>St r r</i> )
5	regulator w trybie sterowanie ręcznego ( <i>St r E</i> )
4	regulator w trybie ręcznego programowania parametrów ( <i>Pr og</i> )
3	aktywny PRIORYTET C.W.U.
2	nieprawidłowo zakończona dezynfekcja instalacji
1	aktywna DEZYNFEKCJA INSTALACJI
0	x

Indeks 173 { R- } : 2 słowo statusu regulatora:

Tablica 4.

bit	Opis
15	x
14	x
13	x
12	x
11	x
10	x
9	x
8	x
7	x
6	czas: 0 = zimowy, 1 = letni
5	stan przełącznika P1
4	stan przełącznika P2
3	stan przełącznika P3
2	stan przełącznika P4
1	program
0	dobowy:
	00b = brak realizacji programu dobowego 01b = roboczy, 10b = świąteczny, 11b = specjalny

x = zarezerwowany

### 3.2 Zdalne sterowanie przełącznikami.

Zdalne sterowanie przełącznikami odbywa się przez indeksy 181, 182 oraz 183. Dostęp do poszczególnych bitów zmiennych dostępnych przez indeksy 181 i 183 umożliwiają bity maski umieszczone w starszym bajcie słowa.

Załączenie trybu zdalnego sterowania następuje po ustawieniu na "1" bitu0 zmiennej o indeksie 181. Na wyświetlaczu regulatora pojawi się wówczas komunikat **S t r r**, a sterowanie przełącznikami będzie możliwe tylko przez interfejs lub przy użyciu funkcji sterowania ręcznego. Działanie pozostałych funkcji regulatora pozostaje niezmienione.

Załączenie trybu zdalnego sterowania oraz stany przełączników zostają zapamiętane w pamięci nieulotnej regulatora i obowiązują do czasu wyłączenia zdalnego sterowania. Podczas pracy w trybie zdalnego sterowania możliwe jest uruchomienie funkcji sterowania ręcznego **S t r E** z klawiatury regulatora, która ma wyższy priorytet realizacji. Do chwili zakończenia sterowania ręcznego sterowanie zdalne nie wpływa na stan przełączników.

Możliwe są dwa tryby zdalnego sterowania przełącznikami. Różnica polega na sposobie sterowania przełącznikami otwierania i zamykania: **tryb 1** pozwala na sterowanie tymi przełącznikami przez podanie względnej zmiany położenia zaworu, a **tryb 2** przez bezpośrednie podanie stanu każdego z przełączników. Sterowanie przełącznikami pompy i dodatkowym polega w obu przypadkach na bezpośrednim podaniu ich stanu.



W odróżnieniu od pracy automatycznej, zdalne sterowanie w trybie 2 umożliwia jednoczesne załączenie przełączników P1 i P2, co może spowodować uszkodzenie napędu zaworu.

Sposób wyboru danego trybu oraz jego opis podano w tablicy poniżej:

Tablica trybów zdalnego sterowania przełącznikami.

Tablica 5.

Wartości zmiennych:			Opis
Indeks 181	indeks 182	indeks 183	
bit0 = 0 bit1 nieistotny	zapis niemożliwy	zapis niemożliwy	zdalne sterowanie wyłączone
bit0 = 1 bit1 = 0	nowe położenie zaworu wyrażone w [%·100]	bit0 zapis niemożliwy bit1 zapis niemożliwy bit2 = stan przełącznika P3 bit3 = stan przełącznika P4	<b>tryb 1</b> zdalnego sterowania. 1. przełączniki P1 (otwierania) i P2 (zamykania) sterowane przez zmienną pod indeksem <u>182</u> o wartości równej nowemu położeniu zaworu. Położenie podaje się w: ( [%] otwarcia zaworu ) ·100. 2. przełączniki P3 i P4 sterowane przez indeks <u>183</u> .
bit0 = 1 bit1 = 1	zapis niemożliwy	bit0 = stan przełącznika P1 bit1 = stan przełącznika P2 bit2 = stan przełącznika P3 bit3 = stan przełącznika P4	<b>tryb 2</b> zdalnego sterowania. 1. wszystkie przełączniki sterowane przez indeks <u>183</u> .

Zdalne sterowanie można wyłączyć w następujący sposób:

1. zdalnie, przez wyzerowanie bitu0 zmiennej spod indeksu 181; po wyłączeniu zdalnego sterowania stany przełączników nie są zmieniane do chwili wysterowania przez regulator;
2. ręcznie, przez naciśnięcie i przytrzymanie przycisku **P** podczas załączania zasilania regulatora; wyłączenie zdalnego sterowania w ten sposób powoduje wyzerowanie zmiennych dostępnych przez indeksy 181, 182 i 183;
3. przez przywrócenie nastaw fabrycznych zdalnie (indeks 184) lub ręcznie; również powoduje zerowanie zmiennych spod indeksów 181, 182 i 183;

### 3.3 Zapisywanie programu dobowego dla temperatur zadanych.

Podczas programowania czasów **t S P 1..6** programu dobowego (indeksy 81..116) regulator na bieżąco kontroluje jego uporządkowanie. Polega to na sprawdzaniu, czy czas odcinka występującego po aktualnie zapisywanym jest późniejszy (lub taki sam). Jeżeli tak nie jest, zostaje zmieniony na wartość aktualnie zapisywaną.

### 3.4 Zapisywanie programu dobowego działania pompy cyrkulacyjnej.

Również podczas programowania czasów **ŁC91.6** programu działania pompy cyrkulacyjnej (indeksy 122..142) regulator na bieżąco kontroluje jego uporządkowanie. Polega to na sprawdzaniu, czy czas odcinka występującego po aktualnie zapisywanym jest późniejszy (lub taki sam). Jeżeli tak nie jest, zostaje zmieniony na wartość aktualnie zapisywaną.

### 3.5 Zapisywanie dni dodatkowych.

Dodatkowe dni robocze, świąteczne oraz specjalne umieszczone są pod indeksami odpowiednio: 19..38, 39..58, 59..78. Podczas zapisywania nowej daty dodatkowej sprawdzane jest, czy istnieje już dzień dodatkowy z tą datą. W przypadku powtórzenia się daty dni dodatkowych, regulator nie wykona zapisu daty i zwróci komunikat o błędzie.

### 3.6 Zdalne wpisanie nastaw fabrycznych.

Nastawy fabryczne wszystkich parametrów regulatora oprócz prędkości transmisji, adresu grupowego i adresu regulatora można przywrócić przez zapisanie wartości AA55h (MSB=170, LSB=85) pod indeksem 184. Po zdalnym wpisaniu nastaw fabrycznych nastąpi kilkusekundowy brak komunikacji z regulatorem. W tym czasie zostaną zapisane nastawy fabryczne, przeprowadzone zostanie rozpoznawanie czujników, po czym regulator powróci do normalnej pracy.

### 3.7 Zdalny restart regulatora.

W celach serwisowych możliwe jest wywołanie zdalnego restartu regulatora z inicjalizacją wszystkich rejestrów pamięci (reset sprzętowy). Aby to zrobić, należy przesłać wartość AA55h (MSB=170, LSB=85) pod indeks 250. Odczyt wartości spod tego indeksu nie jest możliwy. Po zapisie podanej wartości regulator zachowa się w taki sam sposób, jak po załączeniu zasilania.

## 4. Specyfikacja protokołu LUMBUS.

Aby uzyskać wymianę informacji przy wykorzystaniu łącza szeregowego należy wybrać typ interfejsu i ustalić sposób interpretacji danych przesyłanych przy jego pomocy. Typ interfejsu definiuje jedynie parametry elektryczne transmisji i sposób łączenia urządzeń. Od interpretacji danych zależą takie cechy jak możliwość obsługi wielu urządzeń, sprawdzanie poprawności transmisji oraz zasady dostępu do urządzenia. Zadaniem protokołu jest określenie jakie typy danych są dozwolone i jak je interpretować.

### Cechy protokołu:

- łączność asynchroniczna;
- możliwa współpraca wielu urządzeń;
- wyróżnienie dwóch typów urządzeń - pasywnych i aktywnych;
- możliwa współpraca wielu urządzeń aktywnych;
- detekcja błędów transmisyjnych;
- konstrukcja warstwowa .

### 4.1 Podstawowe pojęcia

**Bit** - elementarna jednostka informacji przyjmuje wartość 0 albo 1.

**Bajt** - ciąg 8 bitów reprezentujący liczbę całkowitą z zakresu 0-255. Bity numerowane są kolejno od 7 do 0, a wartość bajtu określamy na podstawie wzoru:

$$\text{wartość} = \sum_{n=0}^{7} 2^n \times \text{bit}_n$$

**Słowo** - dwa bajty reprezentujące liczbę całkowitą z zakresu 0-65535. Pierwszy transmitowany bajt nazywamy starszym a drugi młodszym, wartość słowa określamy na podstawie wzoru:

$$\text{wartość} = 256 \times \text{starszy} + \text{młodszy}.$$

**Tetrada** - ciąg 4 bitów reprezentujący liczbę z zakresu 0-15. Najczęściej wartość tetrady przedstawia się w zapisie szesnastkowym 0-F. W odniesieniu do bajtu używa się pojęcia tetrady starszej reprezentującej bity 7-4 oraz tetrady młodszej dla bitów 3-0.

**Znak** - elementarny ciąg bitów wysyłanych i odbieranych przez interfejs jako niepodzielna jednostka informacyjna.

**Alfabet** - grupa znaków o określonej funkcji w protokole łączności.

**Blok** - ciąg bajtów reprezentujący określoną strukturę danych.

**Pakiet** - ciąg znaków używany do wymiany danych pomiędzy urządzeniami. Pakiet zawiera kodowane przy pomocy alfabetów trzy podstawowe bloki : adresowy, danych i kontroli. Dzięki alfabetycznemu kodowaniu bloków uzyskuje się częściową kompresję danych oraz możliwe jest wyróżnienie początku i końca pakietu.

**Ramka** - ciąg bajtów, który po odpowiednim zakodowaniu tworzy pakiet.

**Żądanie** - pakiet rozpoczynający wymianę danych pomiędzy dwoma urządzeniami.

**Odpowiedź** - pakiet wysyłany jako reakcja na odebrane żądanie.

**Urządzenie pasywne (slave)** - urządzenie, które może jedynie odpowiadać na wysyłane do niego żądania.

**Urządzenie aktywne (master)** - urządzenie, które może wysyłać lub odpowiadać na żądania.

**Uprawnienie** - prawo wysyłania żądań.

**Urządzenie uprawnione** - urządzenie aktywne w danej chwili posiadające uprawnienie.

**Pięśćć logiczny** - logiczne uporządkowanie urządzeń aktywnych, które określa zasady przekazywania uprawnienia pomiędzy urządzeniami.

#### 4.2 Warstwa fizyczna

Warstwa fizyczna definiuje cechy fizyczne urządzeń współpracujących przy użyciu tego protokołu oraz określa logiczny podział transmitowanych przez nie znaków.

Cechy fizyczne:

- interfejs szeregowy RS-232C albo RS-485;
- szybkość transmisji 1200, 2400, 4800 i 9600 bitów/s;
- znak składa się z 1 bitu startu, 7 bitów danych, 1 bitu nieparzystości i 1 bitu stopu;
- RS-485 łączy się równolegle a urządzenia nieaktywne wyłączają swoje nadajniki;
- urządzenia nieaktywne w sposób ciągły monitorują magistralę, aby asynchronicznie odbierać żądania wysyłane do nich od urządzenia aktualnie aktywnego;
- urządzenie nieaktywne ma obowiązek rozpocząć transmisję odpowiedzi na żądanie w czasie nie większym niż czas potrzebny na transmisję trzech znaków;
- interwały pomiędzy znakami pakietu nie mogą być większe niż czas potrzebny na transmisję dwóch znaków.

Znaki transmitowane przez łącze składają się z 10 bitów. Bity startu i stopu są generowane automatycznie przez układy UART i służą do synchronizacji przy transmisji pojedynczych znaków. Po zostale osiem bitów podzielone zostało w następujący sposób:

- 3..0 tetrada danych (wartość 0..15);
- 6..4 alfabet (wartość 0.. 7);
- 7 bit nieparzystości.

Podzielenie znaku na osiem przedziałów liczbowych pozwala na zdefiniowanie ośmiu alfabetów o następujących funkcjach w protokole:

alfabet	zakres	funkcja	znaczenie znaku
0	0x00..0x0f	początek pakietu wymiany danych	najstarsza tetrada adresu
1	0x10..0x1f	początek pakietu zarządzania	starsza tetrada adresu
2	0x20..0x2f	dla pakietu odpowiedzi znak 0x20 pozwala wydłużyć czas odpowiedzi o czas kompletowania danych	starsza tetrada sumy kontrolnej
3	0x30..0x3f	koniec pakietu	młodsza tetrada sumy kontrolnej
4	0x40..0x4f	pojedyncza tetrada danych	tetrada danych
5	0x50..0x5f	dwie tetrazy danych, pierwsza o wartości	druga tetrada danych

		0	
6	0x60..0x6f	trzy tetrydy danych, dwie pierwsze o wartości 0	trzecia tetrada danych
7	0x70..0x7f	cztery tetrydy danych, trzy pierwsze o wartości 0	czwarta tetrada danych

Transmisja znaku odbywa się w następującej kolejności: bit startu, bit 0, bit 1, ..., bit 7, bit stopu. Bit nieparzystości jest równy 1 gdy liczba bitów równych 1 jest parzysta, nieparzystość liczymy dla bitów 6-0.

### 4.3 Warstwa transmisji

Warstwa transmisji definiuje zasady nawiązywania dialogu i wymiany bloków danych pomiędzy urządzeniami. Wyróżniono dwa typy urządzeń: pasywne (slave) oraz aktywne (master). Pasywne mogą jedynie odpowiadać na wysyłane żądania, prawo wysyłania żądania ma tylko jedno z urządzeń aktywnych. Jeżeli w systemie jest więcej urządzeń aktywnych, to tworzą one pierścień logiczny, w którym w danej chwili wyróżnione jest tylko jedno urządzenie aktywne uprawnione do wysyłania żądań. Podczas pracy w pierścieniu urządzenie uprawnione ma obowiązek przekazywania uprawnienia kolejnemu urządzeniu aktywnemu co sprawia, że urządzenia aktywne mogą cyklicznie generować żądania. W warstwie transmisji wydzielono dwie niezależne usługi, zarządzanie pierścieniem i wymiana danych. Nawiązanie dialogu odbywa się zawsze pomiędzy dwoma urządzeniami, urządzeniem uprawnionym i dowolnym aktywnym lub pasywnym. Dialog rozpoczyna urządzenie uprawnione wysyłając zależnie od realizowanej usługi pakiet żądania wymiany danych albo pakiet zarządzania, potwierdzeniem prawidłowego połączenia jest pakiet odpowiedzi wysyłany przez drugie urządzenie albo przejęcie kontroli przez drugie urządzenie.

#### 4.3.1 Struktura pakietu wymiany danych

Pakiet wymiany danych służy do realizacji dialogu pomiędzy urządzeniem uprawnionym i drugim wybranym przez nie urządzeniem. W pakiecie wyróżniono trzy bloki: adresowy, danych i sumy kontrolnej, każdy z nich pełni inną rolę w transmisji oraz posiada własne reguły kodowania.

Pierwszy znak pakietu jest kodowany w alfabecie 0 i zawiera najstarszą tetradę adresu, pozostałe tetrydy adresu i blok danych są kodowane przy pomocy alfabetów 4, 5, 6 i 7 według następującej reguły:

- ciąg tetrad zamieniany jest na ciąg znaków według zasady:
  - ◆ tetrada różna od zera lub ostatnia kodowana jest w alfabecie 4;
  - ◆ dwie tetrydy, pierwsza zerowa i druga różna od zera lub ostatnia są kodowane jako znak w alfabecie 5;
  - ◆ trzy tetrydy, dwie pierwsze zerowe i trzecia różna od zera lub ostatnia są kodowane jako znak w alfabecie 6;
  - ◆ cztery tetrydy, trzy pierwsze zerowe i czwarta różna od zera lub ostatnia są kodowane jako znak w alfabecie 7;
- do znaków dodawany jest bit nieparzystości jako bit 7 znaku;

Dwa ostatnie znaki pakietu reprezentują bajt sumy kontrolnej, starsza tetrada sumy jest kodowana w alfabecie 2, a młodsza w alfabecie 3. Młodsza tetrada sumy kontrolnej jest ostatnim znakiem pakietu. Jeżeli pakiet jest pakietem odpowiedzi to pomiędzy pierwszym i ostatnim znakiem pakietu może wystąpić dowolna ilość znaków 0x20 (spacji). Znak ten służy do utrzymania transmisji na magistrali podczas gdy urządzenie kompletuje dane odpowiedzi.

#### 4.3.2 Blok adresu

Blok adresu jednoznacznie wskazuje na urządzenie lub grupę urządzeń, do których kierowany jest pakiet, a w przypadku odpowiedzi wskazuje urządzenie od którego pochodzi pakiet. Wyróżniono trzy typy adresu:

typ adresu	format binarny		zakres	długość	opis
krótki	00dd	dddd	0..63	bajt	adres urządzenia reprezentowany przez bajt o wartości liczbowej równej: bit 7=0, bit 6=0, bity 5..0=wartość adresu
długi	10dd	dddd dddd dddd	64.. ..16383	słowo	adres urządzenia reprezentowany przez słowo o wartości liczbowej równej: bit 15=1, bit 14=0, bity 13..0=wartość adresu
grupowy	01dd	dddd	0..63	bajt	adres grupy lub grup urządzeń reprezentowany przez bajt o wartości liczbowej równej: bit 7=0, bit 6=1, bit 5..0=1/0 grupa 5..0=tak/nie wartość 64 adresuje wszystkie urządzenia

Grupowe żądania mogą dotyczyć jedynie obiektów wyszczególnionych w aplikacji, taki sposób adresowania wyklucza generowanie przez urządzenia odpowiedzi. Wartość adresu grupowego jest traktowana jako ciąg sześciu bitów, każdy bit oznacza jedną z sześciu grup. Grupa jest adresowana gdy odpowiadający jej bit ma wartość jeden, dozwolone są dowolne kombinacje grup. Urządzenie może należeć do jednej lub kilku grup i reaguje na każde żądanie, które adresuje przynajmniej jedną z grup do której należy urządzenie.

Blok adresu jest pierwszym polem ramki dlatego pierwsza tetrada adresu kodowana jest w alfabecie 0, następne są kodowane przy pomocy alfabetów 4..7. Adresy krótki i grupowy wykorzystują tylko alfabet 4, kodowanie adresu długiego odbywa się wg reguł opisanych w punkcie 1.3.1. Ostatni znak pola adresu musi wystąpić w ramce.

*Kodowanie adresu krótkiego (przykład dla adresu 29 0x1d):*

utworzenie bajtu o wartości adresu	0x1d;
podzielenie bajtu na starszą i młodszą tetradę	0x01,0x0d;
dodanie do starszej tetrady alfabetu 0	0x01;
dodanie do młodszej tetrady alfabetu 4	0x4d;
dodanie bitu nieparzystości	0x01,0xcd.

*Kodowanie adresu długiego (przykład dla adresu 4612 0x1204):*

utworzenie pary bajtów o wartości adresu	0x12,	0x04;
ustawienie bitu 15=1 i bitu 14=0	0x92,	0x04;
podzielenie bajtów na tetrady	0x09,0x02,0x00,0x04;	
dodanie do najstarszej tetrady alfabetu 0	0x09;	
dodanie do pozostałych tetrad alfabetu 4,5 lub 6	0x42,	0x54;
dodanie bitu nieparzystości	0x89,0xc2,	0x54.

*Kodowanie adresu grupowego*

(przykład wywołania urządzeń należących do grupy 1 lub grupy 4):

utworzenie bajtu o wartości reprezentującej wybrane grupy:

ustawienie bitu 1 i bitu 4 (wywołanie grupy 1 i 4)	0x12
bit 6 adresu grupy ustawiony na jeden	0x52;
podzielenie bajtu na starszą i młodszą tetradę	0x05,0x02;
dodanie do starszej tetrady alfabetu 0	0x05;
dodanie do młodszej tetrady alfabetu 4	0x42;
dodanie bitu nieparzystości	0x85,0xc2.

### 4.3.3 Blok danych

Blok danych składa się z jednego lub więcej bajtów danych zakodowanych w alfabetych 4-7. W warstwie transmisji określono zasady tworzenia i kodowania danych bez wnikania w treść i rodzaj informacji przekazywanej przez ten blok. Podstawowym polem bloku danych jest pierwszy bajt bloku danych nazywany FC (field control), określa on sposób interpretacji całego bloku danych. Pole FC zdefiniowano następująco:

nr bitu FC	7	6	5	4	3	2	1	0
żądanie	x	f4	f3	f2	f1	f0	p	0
odpowieź	x	e3	e2	e1	e0	t1	t0	1

Znaczenie bitów:

- **x** - bit zarezerwowany wartość 0;
- **p** - priorytet informacji 1-wysoki 0-niski;
- **f4..0** - kod funkcji
  - 0 - odczyt parametru,
  - 1 - zapis parametru,
  - 2 - odczyt warunkowy (parametry śledzone),
  - 3 - identyfikacja urządzenia,
  - 4-31 - zarezerwowane;
- **t1..0** - typ urządzenia odpowiadającego
  - 0 - urządzenie pasywne,
  - 1 - aktywne nie gotowe do wejścia do pierścienia,
  - 2 - aktywne gotowe do wejścia do pierścienia,
  - 3 - aktywne pracujące w pierścieniu;
- **e3..0** - kod wykonania operacji określonej w ramce żądania
  - 0 - urządzenie wykonało żądanie poprawnie,
  - 1 - realizacja żądania niemożliwa,
  - 2 - przy odczycie śledzonym pominięto obiekty,
  - 4 - przy zapisie przekroczony zakres;

Następne bajty bloku danych tworzą pole danych. W polu danych mogą występować takie elementy jak indeks, wartość obiektu wskazywanego przez indeks, identyfikator urządzenia nadającego żądanie, typ urządzenia albo pole jest puste.

Obiekty, których dotyczy wymiana danych są reprezentowane liczbowo przez indeks. Występuje on za polem FC przy żądaniach zapisu, odczytu lub odczytu warunkowego i jednoznacznie określa obiekt, którego dotyczy realizowana funkcja. Indeks występuje także w bloku odpowiedzi na żądanie warunkowego odczytu i poprzedza każdą transmitowaną wartość. Wartość liczbowa indeksu jest kodowana jako słowo i zawiera się w przedziale liczbowym 0..65535.

W bloku żądania zapisu oraz w blokach odpowiedzi na żądanie odczytu występuje pole wartości. Sposób interpretacji, długość i format tego pola definiuje warstwa aplikacji.

W funkcji odczytu warunkowego transmitowany jest bajt identyfikatora urządzenia nadającego żądanie. Odpowiedź na tą funkcję zależy od tego czy dana odczytywana warunkowo uległa modyfikacji od czasu poprzedniego odczytu przez urządzenie o takim identyfikatorze. W systemie z wieloma urządzeniami aktywnymi należy unikać powtarzania się identyfikatorów.

W funkcji identyfikacji odczytywane jest słowo typu urządzenia. Każdy typ urządzenia lub wykonanie specjalne musi mieć przydzielony indywidualny numer z zakresu 0..65535, jeśli ma inny zestaw obiektów lub inna jest ich interpretacja. Numer identyfikacyjny służy do automatycznego rozpoznawania urządzeń w sieciach zawierających różne urządzenia posiadające identyczny protokół.

### 4.3.4 Blok sumy kontrolnej

Blok sumy kontrolnej FCS służy do detekcji błędów transmisyjnych. Jest to jednobajtowa wartość sumy bez przeniesienia wszystkich znaków pakietu z bloków adresowego i danych:

$$\text{FCS} = \text{znak1} \text{ xor } \text{znak2} \text{ xor } \dots \text{ znakN}$$

Bajt sumy kontrolnej po podzieleniu na tetrazy jest kodowany w alfabetych 2 i 3.



### 4.3.5 Usługi warstwy transmisji

Warstwa transmisji realizuje dwie podstawowe usługi: wymianę danych i zarządzanie pierścieniem logicznym, tworzonym w sieci zawierającej więcej niż jedno urządzenie aktywne.

Do wymiany danych służą cztery podstawowe funkcje: identyfikacja urządzenia, zapis wartości do obiektu, odczyt i odczyt warunkowy.

Funkcje te realizowane są w cyklu żądanie - odpowiedź przy pomocy następujących bloków danych:

Funkcja transmisji	Żądanie		Odpowiedź	
	FC	Pole danych	FC	Pole danych
Odczyt wartości obiektu	00000000	Indeks ...	00000tt1	Wartość
Zapis wartości obiektu	00000100	Indeks Wartość	00000tt1	...
Odczyt warunkowy	00001000	Indeks Identyfikator	00000tt1	Indeks Wartość ...
Identyfikacja	00001100		00000tt1	Typ urządzenia (słowo)

... - ciąg bajtów określony w warstwie aplikacji (domyślnie ciąg pusty)

Zarządzanie pierścieniem logicznym odbywa się przy pomocy funkcji przekazujących uprawnienia. Funkcje te są implementowane tylko w urządzeniach aktywnych i określają zasady pracy tych urządzeń w sieciach typu multi-master. W zależności od topologii sieci (typu interfejsu), różnią się realizacje funkcji zarządzających uprawnieniem.

### 4.4 Warstwa aplikacji

#### 4.4.1 Standardowe typy danych

Aby ujednoczyć formaty przesyłanych wartości, w warstwie zdefiniowano podstawowe typy danych z których mogą być tworzone typy pochodne potrzebne do stworzenia aplikacji konkretnego urządzenia.

Typ	Długość	Zakres liczbowy	Uwagi																																													
float	4 bajty	±1.175494 E-38 .. ±3.402823 E+38	liczba rzeczywista wg standardu IEEE 754 kolejne bajty zawierają następujące informacje: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>nr bitu</th> <th>7</th> <th>6</th> <th>5</th> <th>4</th> <th>3</th> <th>2</th> <th>1</th> <th>0</th> </tr> </thead> <tbody> <tr> <td><b>bajt 1</b></td> <td>m7</td> <td>m6</td> <td>m5</td> <td>m4</td> <td>m3</td> <td>m2</td> <td>m1</td> <td>m0</td> </tr> <tr> <td><b>bajt 2</b></td> <td>m15</td> <td>m14</td> <td>m13</td> <td>m12</td> <td>m11</td> <td>m10</td> <td>m9</td> <td>m8</td> </tr> <tr> <td><b>bajt 3</b></td> <td>x0</td> <td>m22</td> <td>m21</td> <td>m20</td> <td>m19</td> <td>m18</td> <td>m17</td> <td>m16</td> </tr> <tr> <td><b>bajt 4</b></td> <td>s</td> <td>x7</td> <td>x6</td> <td>x5</td> <td>x4</td> <td>x3</td> <td>x2</td> <td>x1</td> </tr> </tbody> </table> s- bit znaku 0-liczba dodatnia 1-liczba ujemna x7-0 liczba typu char określająca dwójkowy wykładnik reprezentowanej liczby m22-0 mantysa liczby, dokładność liczby jest 24-bitowa ponieważ wykładnik liczby musi być tak dobrany, aby bit 23 mantysy był równy jeden (liczba wyrównana do 24-bitów m23=1) wartość= (-1) <sup>s</sup> 2 <sup>x</sup> m	nr bitu	7	6	5	4	3	2	1	0	<b>bajt 1</b>	m7	m6	m5	m4	m3	m2	m1	m0	<b>bajt 2</b>	m15	m14	m13	m12	m11	m10	m9	m8	<b>bajt 3</b>	x0	m22	m21	m20	m19	m18	m17	m16	<b>bajt 4</b>	s	x7	x6	x5	x4	x3	x2	x1
nr bitu	7	6	5	4	3	2	1	0																																								
<b>bajt 1</b>	m7	m6	m5	m4	m3	m2	m1	m0																																								
<b>bajt 2</b>	m15	m14	m13	m12	m11	m10	m9	m8																																								
<b>bajt 3</b>	x0	m22	m21	m20	m19	m18	m17	m16																																								
<b>bajt 4</b>	s	x7	x6	x5	x4	x3	x2	x1																																								
bit	1 bajt	prawda 1 fałsz 0	wartość bajtowa bitu kodowana dla fałszu jako zero, dla prawdy jako różna od zera																																													

bajt	1 bajt	0..255 (0..0xff)	liczba naturalna
------	--------	------------------	------------------

uchar			
char	1 bajt	-128..0..127	liczba całkowita w kodzie uzupełnienia do dwóch -wartość $\llcorner \Rightarrow$ (wartość xor 255)+1 np. $-3 \llcorner \Rightarrow (3 \text{ xor } 255)+1 \llcorner \Rightarrow 253$
word (słowo) uint	2 bajty	0..65535 (0..0xffff)	liczba naturalna, starszy bajt pierwszy
int	2 bajty	-32768..0..32767	liczba całkowita w kodzie uzupełnienia do dwóch, starszy bajt pierwszy
ulong	4 bajty	$0..2^{32}-1$ (0..0xffffffff)	liczba naturalna, najstarszy bajt pierwszy
long	4 bajty	$-2^{31}..2^{31}-1$	liczba całkowita w kodzie uzupełnienia do dwóch, najstarszy bajt pierwszy
string	dowolna	znaki ASCII	dowolna liczba bajtów reprezentujących znaki ASCII zakończona bajtem o wartości 0
tablica	dowolna	zależny od typu	ciąg wartości tego samego typu, typ określa sposób kodowania i ilość bajtów reprezentującą pojedynczy element tablicy, ilość elementów reprezentowana jest przez wymiar tablicy
rekord	dowolna	zależny od defini- cji	ciąg wartości dowolnego typu określonego podczas definicji rekordu

#### 4.4.2 Indeksy

Indeks jest liczbową reprezentacją zmiennej udostępnionej w aplikacji urządzenia. Formatem indeksu jest typ word, co pozwala na zdefiniowanie 65536 zmiennych w urządzeniu. Typy zmiennych dzielimy na dwie grupy:

- **typy proste:** bit, byte, char, word, int, dword, long, float, string;
- **typy złożone:** tablice, rekordy, tablice rekordów.

Dla zmiennych typu prostego przydzielamy jedną liczbę z zakresu 0..65535. Dla różnych zmiennych liczby te nie mogą się powtarzać.

Dla zmiennych typu złożonego przydzielany jest przedział liczbowy z zakresu 0..65535. Pierwsza liczba z zakresu nazywana indeksem bazowym reprezentuje całą strukturę złożoną, następne wartości pozwalają na dostęp do pojedynczych elementów struktury.

Odwołanie się do indeksu, który nie reprezentuje żadnej zmiennej powoduje pojawienie się komunikatu eeee=1, realizacja żądania niemożliwa.

Warstwa aplikacji może definiować własne reguły przydziału indeksów i kodowania danych.



Programy dobowe dla temperatur zadanych **SP<sub>CU</sub>** (cz. 1):Program dobowy roboczy dla temperatur zadanych: **SP**

4081	<b>SP1</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4082	<b>SP1</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4083	<b>SP2</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4084	<b>SP2</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4085	<b>SP3</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4086	<b>SP3</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4087	<b>SP4</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4088	<b>SP4</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4089	<b>SP5</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4090	<b>SP5</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4091	<b>SP6</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4092	<b>SP6</b>	poprawka dobowa	-50..50	int	[°C]	{RW}

Program dobowy świąteczny dla temperatur zadanych: **SP**

4093	<b>SP1</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4094	<b>SP1</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4095	<b>SP2</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4096	<b>SP2</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4097	<b>SP3</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4098	<b>SP3</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4099	<b>SP4</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4100	<b>SP4</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4101	<b>SP5</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4102	<b>SP5</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4103	<b>SP6</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4104	<b>SP6</b>	poprawka dobowa	-50..50	int	[°C]	{RW}

Program dobowy specjalny dla temperatur zadanych: **SP**

4105	<b>SP1</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4106	<b>SP1</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4107	<b>SP2</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4108	<b>SP2</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4109	<b>SP3</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4110	<b>SP3</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4111	<b>SP4</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4112	<b>SP4</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4113	<b>SP5</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4114	<b>SP5</b>	poprawka dobowa	-50..50	int	[°C]	{RW}
4115	<b>SP6</b>	czas rozpoczęcia poprawki	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4116	<b>SP6</b>	poprawka dobowa	-50..50	int	[°C]	{RW}

Parametry priorytetu C.W.U. (cz. 1.) **PP<sub>CU</sub>**

4117	<b>SPPr</b>	wartość temperatury T1, poniżej której zadziała priorytet c.w.u.	5..95	int	[°C]	{RW}
------	-------------	--	-------	-----	------	------

Parametry pracy pompy cyrkulacyjnej **PP<sub>CU</sub>**

4118	<b>SPc1</b>	zadana wartość temperatury T1 do sterowania pompą cyrkulacyjną	5..95	int	[°C]	{RW}
4119	<b>SPc2</b>	zadana wartość temperatury T2 wody w przewodzie cyrkulacyjnym	5..95	int	[°C]	{RW}

c.d. tablicy 6.

4120	h . 12	histereza dla wartości zadanych $SPc1$ i $SPc2$	0..200	int	[°C x 10]	{ RW }
4121	tryb	tryb sterowania pompą cyrkulacyjną	0..3	int	0 = tryb 1 1 = tryb 2 2 = tryb 3 3 = tryb 4	{ RW }
<b>Program roboczy działania pompy cyrkulacyjnej: c 3</b>						
4122	c 3 1..6	odblokowanie lub wstrzymanie działania pompy cyrkulacyjnej MSB: maska LSB: <b>bit0</b> = cy1 <b>bit1</b> = cy2 <b>bit2</b> = cy3 <b>bit3</b> = cy4 <b>bit4</b> = cy5 <b>bit5</b> = cy6	0..3Fh, 0..3Fh	int [MSB,LSB]	0 = wyłączony 1 = załączony	{ RW }
4123	tc 3 1	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4124	tc 3 2	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4125	tc 3 3	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4126	tc 3 4	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4127	tc 3 5	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4128	tc 3 6	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
<b>Program święteczny działania pompy cyrkulacyjnej: c 3</b>						
4129	c 3 1..6	odblokowanie lub wstrzymanie działania pompy cyrkulacyjnej MSB: maska LSB: <b>bit0</b> = cy1 <b>bit1</b> = cy2 <b>bit2</b> = cy3 <b>bit3</b> = cy4 <b>bit4</b> = cy5 <b>bit5</b> = cy6	0..3Fh, 0..3Fh	int [MSB,LSB]	0 = wyłączony 1 = załączony	{ RW }
4130	tc 3 1	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4131	tc 3 2	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4132	tc 3 3	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4132	tc 3 4	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4133	tc 3 5	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
4135	tc 3 6	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
<b>Program specjalny działania pompy cyrkulacyjnej: c 3</b>						
4136	c 3 1..6	odblokowanie lub wstrzymanie działania pompy cyrkulacyjnej MSB: maska LSB: <b>bit0</b> = cy1 <b>bit1</b> = cy2 <b>bit2</b> = cy3 <b>bit3</b> = cy4 <b>bit4</b> = cy5 <b>bit5</b> = cy6	0..3Fh, 0..3Fh	int [MSB,LSB]	0 = wyłączony 1 = załączony	{ RW }

c.d. tablicy 6.

4137	tc 3 1	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{ RW }
------	--------	--	--------------	---------------	---------	--------

4138	ŁC42	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4139	ŁC43	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4140	ŁC44	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4141	ŁC45	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4142	ŁC46	godzina odblokowania / wstrzymania pracy pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Parametry pracy pompy ładującej PPLR</b>						
4143	SPŁ3	zadana wartość temperatury T3 w górnej części zasobnika ciepłej wody	5..95	int	[°C]	{RW}
4144	SPŁ4	zadana wartość temperatury T4 w dolnej części zasobnika ciepłej wody	5..95	int	[°C]	{RW}
4145	h.34	histereza dla wartości zadanych SPŁ3 i SPŁ4	0..200	int	[°C x 10]	{RW}
<b>Program roboczy wymuszenia załączeń pompy obiegowej: LR</b>						
4146	ŁLR1	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4147	ŁLR2	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4148	ŁLR3	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4149	ŁLR4	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4150	ŁLR5	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4151	ŁLR6	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Program świąteczny wymuszenia załączeń pompy obiegowej: LR</b>						
4152	ŁLR1	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4153	ŁLR2	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4154	ŁLR3	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4155	ŁLR4	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4156	ŁLR5	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4157	ŁLR6	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Program specjalny wymuszenia załączeń pompy obiegowej: LR</b>						
4158	ŁLR1	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4159	ŁLR2	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4160	ŁLR3	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4161	ŁLR4	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4162	ŁLR5	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4163	ŁLR6	godzina załączenia pompy	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
<b>Różnice temperatur dla czujników rŁ</b>						
4164	rŁ1	różnica temperatury dla T1	-9.9..10.0	int	[°C x 10]	{RW}
4165	rŁ2	różnica temperatury dla T2	-9.9..10.0	int	[°C x 10]	{RW}
4166	rŁ3	różnica temperatury dla T3	-9.9..10.0	int	[°C x 10]	{RW}
4167	rŁ4	różnica temperatury dla T4	-9.9..10.0	int	[°C x 10]	{RW}
<b>Konfiguracja czujników ŁonF</b>						
4168	Ł2 Ł3 Ł4	MSB: maska LSB: bit0 = czujnik T2 bit1 = czujnik T3 bit2 = czujnik T4	0..07h, 0..07h	int [MSB, LSB]	0 = brak czujnika 1 = jest	{RW}

c.d. tablicy 6.

4169	<b>ŁPo</b>	czas pełnego otwarcia zaworu. <i>Zmiana wartości parametru spowoduje pozycjonowania napędu zaworu</i>	8..360	int	[sek]	{RW}
4170	<b>cod</b>	kod bezpieczeństwa	0..9999	int		{RW}
4171		numer wersji programu		int		{R-}
4172		1-słowo statusu regulatora	<i>patrz → tablica 7.</i>			{R-}
4173		2-słowo statusu regulatora	<i>patrz → tablica 8.</i>			{R-}
4174	<b>Ł1</b>	temperatura T1 na wyjściu wymiennika	-500..1500	int	[°C x 10]	{R-}
4175	<b>Ł2</b>	temperatura T2 wody w przewodzie cyrkulacyjnym	-500..1500	int	[°C x 10]	{R-}
4176	<b>Ł3</b>	temperatura T3 wody w górnej części zasobnika	-500..1500	int	[°C x 10]	{R-}
4177	<b>Ł4</b>	temperatura T4 wody w dolnej części zasobnika	-500..1500	int	[°C x 10]	{R-}
4178		aktualna wartość zadana temperatury T1 wody na wyjściu wymiennika	50..950	int	[°C x 10]	{R-}
4179		położenie zaworu	0..10000	int	[%·100]	{R-}
4180		stan regulacji automatycznej	0..1	int	<b>0</b> = STOP <b>1</b> = START	{RW}
<b>Zdalne sterowanie elementami wykonawczymi:</b>						
4181	<b>5Łrr</b>	zdalne sterowanie elementami wykonawczymi	0..3, 0..3	int [MSB, LSB] MSB: maska LSB: <b>bit0:</b> tryb sterowania elementami wykonawczymi <b>bit1:</b> tryb sterowania przekaźnikami otwierania i zamykania	<b>0</b> = automat. <b>1</b> = zdalne  <b>0</b> = sterow. przez rejestr 4182 <b>1</b> = sterow. przez rejestr 4183	{RW}
4182		nowe położenie siłownika zaworu w czasie zdalnego sterowania	0..10000	int	[% x 100]	{RW}
4183		stan przekaźników w czasie zdalnego sterowania	0..0Fh, 0..0Fh	int [MSB, LSB] MSB: maska LSB: <b>bit0</b> = P1 <b>bit1</b> = P2 <b>bit2</b> = P3 <b>bit3</b> = P4	<b>0</b> = wyłączony <b>1</b> = załączony	{RW}
4184	<b>FRbr</b>	przywrócenie nastaw fabrycznych (oprócz ustawień interfejsu)	AA55h Przy odczycie zwracana jest wartość 0	int		{RW}
4185		aktualna wartość temperatury T1 wody na wyjściu wymiennika, przy której zadziała priorytet ciepłej wody	50..950	int	[°C x 10]	{R-}
4186		aktualna wartość zadana temperatury T1 do sterowania pompą cyrkulacyjną	50..950	int	[°C x 10]	{R-}
4187		aktualna wartość zadana temperatury T2 wody w przewodzie cyrkulacyjnym	50..950	int	[°C x 10]	{R-}

c.d. tablicy 6.

4188		aktualna wartość zadana temperatury T3 wody w górnej części zasobnika	50..950	int	[°C x 10]	{R-}
4189		aktualna wartość zadana temperatury T4 wody w dolnej części zasobnika	50..950	int	[°C x 10]	{R-}
4190	$\xi P_c$	czas pełnego zamknięcia zaworu. <i>Zmiana wartości parametru spowoduje pozycjonowanie napędu zaworu</i>	8..360	int	[sek]	{RW}
<b>Parametry priorytetu C.W.U. (cz. 2.) <math>Prcu</math></b>						
4191	$h, P_r$	histereza działania priorytetu c.w.u.	0..200	int	[°C x 10]	{RW}
<b>Programy dobowe dla temperatur zadanych <math>SPcu</math> (cz. 2):</b>						
4192	$SPcc$	zadana wartość temperatury T1 na wyjściu z wymiennika	5..95	int	[°C]	{RW}
4193	$\xi cuH$	największa wartość, jaką mogą przyjąć wszystkie temperatury zadane	30..95	int	[°C]	{RW}
4194	$\xi cuL$	najmniejsza wartość, jaką mogą przyjąć wszystkie temperatury zadane	5..30	int	[°C]	{RW}
<b>Parametry funkcji dezynfekcji instalacji <math>dF_c</math></b>						
4195	$\xi r.dF$	status dezynfekcji	0..5	int		{RW}
4196	$SP.dF$	wartość zadana temperatury T1 podczas dezynfekcji	60..95	int	[°C]	{RW}
4197	$dt.dF$	dopuszczalne obniżenie temperatury T1 bez wstrzymania zliczania efektywnego czasu dezynfekcji	0..20	int	[°C]	{RW}
4198	$cr.dF$	godzina rozpoczęcia dezynfekcji	0..23, 0..59	int [MSB,LSB]	[gg,mm]	{RW}
4199	$c.dF$	efektywny czas dezynfekcji	30..120	int	[minuty]	{RW}
4200	$ct.dF$	maksymalny czas na przeprowadzenie dezynfekcji	60..240	int	[minuty]	{RW}
4201	$dn, i$	wyłączenie lub załączenie dezynfekcji w kolejne dni tygodnia: MSB: maska LSB: <b>bit0</b> = poniedziałek <b>bit1</b> = wtorek <b>bit2</b> = środa <b>bit3</b> = czwartek <b>bit4</b> = piątek <b>bit5</b> = sobota <b>bit6</b> = niedziela	0..7Fh, 0..7Fh	int [MSB,LSB]	<b>0</b> = wyłączona <b>1</b> = załączona	{RW}

Opis znaczników statusu regulatora (rejstry 4172 i 4173):

Rejestr 4172 {R-}: 1 słowo statusu regulatora:

Tablica 7.

bit	Opis
15	regulator pracuje w TRYBIE AWARYJNYM (błąd pamięci EEPROM)
14	czas i dane kalendarza są nieaktualne
13	temperatura T1 poza zakresem pomiarowym ( $RL - 1$ )
12	temperatura T2 poza zakresem pomiarowym ( $RL - 2$ )
11	temperatura T3 poza zakresem pomiarowym ( $RL - 3$ )
10	temperatura T4 poza zakresem pomiarowym ( $RL - 4$ )
9	stan wejścia binarnego (1=zwarte, 0=rozwarne)
8	silownik w trakcie pozycjonowania
7	regulator w trybie regulacji automatycznej ( $St - t$ )



6	regulator w trybie zdalnego sterowania ( <b>S</b> t <b>r</b> r)
5	regulator w trybie sterowanie ręcznego ( <b>S</b> t <b>r</b> <b>E</b> )
4	regulator w trybie ręcznego programowania parametrów ( <b>P</b> r <b>o</b> g)
3	aktywny PRIORYTET C.W.U.
2	nieprawidłowo zakończona dezynfekcja instalacji
1	aktywna DEZYNFEKCJA INSTALACJI
0	x

Rejestr 173 { R- } : 2 słowo statusu regulatora:

Tablica 8.

bit	Opis
15	x
14	x
13	x
12	x
11	x
10	x
9	x
8	x
7	x
6	czas: 0 = zimowy, 1 = letni
5	stan przełącznika P1
4	stan przełącznika P2
3	stan przełącznika P3
2	stan przełącznika P4
1	program
0	dobowy:
	00b = brak realizacji programu dobowego 01b = roboczy, 10b = świąteczny, 11b = specjalny

x = zarezerwowany

## 5.2 Zdalne sterowanie przełącznikami.

Zdalne sterowanie przełącznikami odbywa się przez rejestry 4181, 4182 oraz 4183. Dostęp do poszczególnych bitów zmiennych dostępnych w rejestrach 4181 i 4183 umożliwiają bity maski umieszczane w starszym bajcie słowa.

Załączenie trybu zdalnego sterowania następuje po ustawieniu na "1" bitu0 zmiennej w rejestrze 4181. Na wyświetlaczu regulatora pojawi się wówczas komunikat **S**t**r**r, a sterowanie przełącznikami będzie możliwe tylko przez interfejs lub przy użyciu funkcji sterowania ręcznego. Działanie pozostałych funkcji regulatora pozostaje niezmienione.

Załączenie trybu zdalnego sterowania oraz stany przełączników zostają zapamiętane w pamięci nieulotnej regulatora i obowiązują do czasu wyłączenia zdalnego sterowania. Podczas pracy w trybie zdalnego sterowania możliwe jest uruchomienie funkcji sterowania ręcznego **S**t**r****E** z klawiatury regulatora, która ma wyższy priorytet realizacji. Do chwili zakończenia sterowania ręcznego sterowanie zdalne nie wpływa na stan przełączników.

Możliwe są dwa tryby zdalnego sterowania przełącznikami. Różnica polega na sposobie sterowania przełącznikami otwierania i zamykania: **tryb 1** pozwala na sterowanie tymi przełącznikami przez podanie względnej zmiany położenia zaworu, a **tryb 2** przez bezpośrednie podanie stanu każdego z przełączników. Sterowanie przełącznikami pompy i dodatkowym polega w obu przypadkach na bezpośrednim podaniu ich stanu.



W odróżnieniu od pracy automatycznej, zdalne sterowanie w trybie 2 umożliwia jednoczesne załączenie przełączników P1 i P2, co może spowodować uszkodzenie napędu zaworu.

Sposób wyboru danego trybu oraz jego opis podano w tablicy poniżej:

Tablica trybów zdalnego sterowania przekaźnikami.

Tablica 9.

Wartości zmiennych:			Opis
Rejestr 4181	rejestr 4182	rejestr 4183	
bit0 = 0 bit1 nieistotny	zapis niemożliwy	zapis niemożliwy	zdalne sterowanie wyłączone
bit0 = 1 bit1 = 0	nowe położenie zaworu wyrażone w [%·100]	bit0 zapis niemożliwy bit1 zapis niemożliwy bit2 = stan przekaźnika P3 bit3 = stan przekaźnika P4	<b>tryb 1</b> zdalnego sterowania. 1. przekaźniki P1 (otwierania) i P2 (zamykania) sterowane przez zmienną w rejestrze 4182 o wartości równej nowemu położeniu zaworu. Położenie podaje się w: ( [%] otwarcia zaworu ) ·100. 2. przekaźniki P3 i P4 sterowane przez rejestr 4183.
bit0 = 1 bit1 = 1	zapis niemożliwy	bit0 = stan przekaźnika P1 bit1 = stan przekaźnika P2 bit2 = stan przekaźnika P3 bit3 = stan przekaźnika P4	<b>tryb 2</b> zdalnego sterowania. 1. wszystkie przekaźniki sterowane przez rejestr 4183.

Zdalne sterowanie można wyłączyć w następujący sposób:

1. zdalnie, przez wyzerowanie bitu0 zmiennej w rejestrze 4181; po wyłączeniu zdalnego sterowania stany przekaźników nie są zmieniane do chwili wysterowania przez regulator;
2. ręcznie, przez naciśnięcie i przytrzymanie przycisku **P** podczas załączania zasilania regulatora; wyłączenie zdalnego sterowania w ten sposób powoduje wyzerowanie zmiennych dostępnych w rejestrach 4181, 4182 i 4183;
3. przez przywrócenie nastaw fabrycznych zdalnie (rejestr 4184) lub ręcznie; również powoduje zerowanie zmiennych w rejestrach 4181, 4182 i 4183;

### 5.3 Zapisywanie programu dobowego dla temperatur zadanych.

Podczas programowania czasów  $t_{SP} 1..6$  programu dobowego (rejestry 4081..4116) regulator na bieżąco kontroluje jego uporządkowanie. Polega to na sprawdzaniu, czy czas odcinka występującego po aktualnie zapisywanym jest późniejszy (lub taki sam). Jeżeli tak nie jest, zostaje zmieniony na wartość aktualnie zapisywaną.

### 5.4 Zapisywanie programu dobowego działania pompy cyrkulacyjnej.

Również podczas programowania czasów  $t_{CY} 1..6$  programu działania pompy cyrkulacyjnej (rejestry 4122..4142) regulator na bieżąco kontroluje jego uporządkowanie. Polega to na sprawdzaniu, czy czas odcinka występującego po aktualnie zapisywanym jest późniejszy (lub taki sam). Jeżeli tak nie jest, zostaje zmieniony na wartość aktualnie zapisywaną.

### 5.5 Zapisywanie dni dodatkowych.

Dodatkowe dni robocze, świąteczne oraz specjalne umieszczone są w rejestrach odpowiednio: 4019..4038, 4039..4058, 4059..4078. Podczas zapisywania nowej daty dodatkowej sprawdzane jest, czy istnieje już dzień dodatkowy z tą datą. W przypadku powtórzenia się daty dni dodatkowych, regulator nie wykona zapisu daty i zwróci komunikat o błędzie.

### 5.6 Zdalne wpisanie nastaw fabrycznych.

Nastawy fabryczne wszystkich parametrów regulatora oprócz prędkości transmisji, adresu grupowego i adresu regulatora można przywrócić przez zapisanie wartości AA55h (MSB=170, LSB=85) do rejestru 4184. Po zdalnym wpisaniu nastaw fabrycznych nastąpi kilkusekundowy brak komunikacji z regulatorem. W tym czasie zostaną zapisane nastawy fabryczne, przeprowadzone zostanie rozpoznawanie czujników, po czym regulator powróci do normalnej pracy.

### 5.7 Zdalny restart regulatora.

W celach serwisowych możliwe jest wywołanie zdalnego restartu regulatora z inicjalizacją wszystkich rejestrów pamięci (reset sprzętowy). Aby to zrobić, należy przesłać wartość AA55h

(MSB=170, LSB=85) do rejestru 4250. Odczyt wartości tego rejestru nie jest możliwy. Po zapisie podanej wartości regulator zachowa się w taki sam sposób, jak po załączeniu zasilania.

## 6. Specyfikacja protokołu MODBUS.

Interfejs MODBUS jest standardem przyjętym przez producentów sterowników przemysłowych dla asynchronicznej, znakowej wymiany informacji pomiędzy urządzeniami systemów pomiarowo kontrolnych. Posiada on takie cechy jak:

- prosta reguła dostępu do łącza oparta na zasadzie "master-slave",
- zabezpieczenie przesyłanych komunikatów przed błędami,
- potwierdzenie wykonywania rozkazów zdalnych i sygnalizacja błędów,
- skuteczne mechanizmy zabezpieczające przed zawieszeniem systemu,
- wykorzystanie asynchronicznej transmisji znakowej.

Kontrolery urządzeń pracujących w systemie MODBUS komunikują się ze sobą przy wykorzystaniu protokołu typu master-slave, w którym tylko jedno urządzenie może inicjalizować transakcje (jednostka nadrzędna-master), a pozostałe (jednostki podrzędne-slave) odpowiadają jedynie na zdalne zapytania jednostki nadrzędnej. Transakcja składa się z polecenia wysyłanego z jednostki master do slave oraz z odpowiedzi przesyłanej w odwrotnym kierunku. Odpowiedź zawiera dane żądane przez master lub potwierdzenie realizacji jego polecenia. Master może przysyłać informację do pojedynczych odbiorców lub informacje rozgłoszeniowe (broadcast), przeznaczone dla wszystkich urządzeń podrzędnych w systemie (na polecenia rozgłoszeniowe master nie otrzymuje odpowiedzi).

Format przesyłanych informacji jest następujący:

- **master** ⇒ **slave** : adres odbiorcy, kod reprezentujący żądane polecenie, dane, słowo kontrolne zabezpieczające przesyłaną wiadomość,
- **slave** ⇒ **master** : adres nadawcy, potwierdzenie realizacji rozkazu, dane żądane przez master, słowo kontrolne zabezpieczające odpowiedź przed błędami.

Jeżeli urządzenie slave wykryje błąd przy odbiorze wiadomości, lub nie może wykonać polecenia, przygotowuje specjalny komunikat o wystąpieniu błędu i przesyła go jako odpowiedź do mastera.

Urządzenia pracujące w protokole MODBUS mogą być ustawione na komunikację przy użyciu jednego z dwóch trybów transmisji: ASCII lub RTU. Użytkownik wybiera żądany tryb wraz z parametrami portu szeregowego (prędkość transmisji, jednostka informacyjna), podczas konfiguracji każdego urządzenia.

W systemie MODBUS przesyłane wiadomości są zorganizowane w ramki o określonym początku i końcu. Pozwala to urządzeniu odbierającemu na odrzucenie ramek niekompletnych i sygnalizację związanych z tym błędów.

Ze względu na możliwość pracy w jednym z dwóch różnych trybów transmisji (ASCII lub RTU), definiuje się dwie ramki.

### 6.1 Ramka w trybie ASCII

W trybie ASCII każdy bajt wiadomości przesyłany jest w postaci dwóch znaków ASCII. Podstawową zaletą tego trybu jest to, iż pozwala on na długie odstępy między znakami (do 1s) bez powodowania błędów.

Format ramki przedstawiono poniżej:

Znacznik początku	Adres	Funkcja	Dane	Kontrola LRC	Znacznik końca
1 znak dwukropka ":"	2 znaki	2 znaki	n znaków	2 znaki	2 znaki: CR LF

Znacznikiem początku jest znak dwukropka (":"- ASCII 3Ah), natomiast znacznikiem końca dwa znaki CR i LF. Część informacyjną ramki zabezpiecza się kodem LRC (Longitudinal Redundancy Check).

## 6.2 Ramka w trybie RTU

W trybie RTU wiadomości rozpoczynają i kończą się odstępem trwającym minimum 3.5 x (czas trwania pojedynczego znaku), w którym panuje cisza na łączu. Najprostszą implementacją wymienionego interwału czasowego jest wielokrotne odmierzanie czasu trwania znaku przy zadanej szybkości bodowej przyjętej na łączu.

Format ramki przedstawiono poniżej:

Znacznik początku	Adres	Funkcja	Dane	Kontrola CRC	Znacznik końca
T1-T2-T3-T4	8 bitów	8 bitów	n x 8 bitów	16 bitów	T1-T2-T3-T4

Znaczniki początku i końca zaznaczono symbolicznie jako odstęp równy czterem długościom znaku (jednostki informacyjnej). Słowo kontrolne jest 16 bitowe i powstaje jako rezultat obliczenia CRC (Cyclical Redundancy Check) na zawartości ramki.

## 6.3 Charakterystyka pól ramki

### Pole adresowe

Pole adresowe w ramce zawiera dwa znaki (w trybie ASCII) lub osiem bitów (w trybie RTU). Zakres adresów jednostek slave wynosi 0..247. Master adresuje jednostki slave umieszczając jej adres na polu adresowym ramki. Kiedy jednostka slave wysłała odpowiedź, umieszcza swój własny adres na polu adresowym ramki, co pozwala masterowi sprawdzić, z którą jednostką realizowana jest transakcja.

Adres 0 jest wykorzystywany jako adres rozgłoszeniowy, rozpoznawany przez wszystkie jednostki slave podłączone do magistrali.

### Pole funkcji

Pole funkcji zawiera dwa znaki w trybie ASCII lub 8-bitów w trybie RTU. Zakres kodów funkcji od 1..255.

Przy transmisji polecenia z jednostki master do slave, pole funkcji zawiera kod rozkazu, określający działanie, które ma podjąć jednostka slave na żądanie mastera.

Kiedy jednostka slave odpowiada masterowi, pole funkcji wykorzystuje do potwierdzenia wykonania polecenia lub sygnalizacji błędu, jeżeli z jakichś przyczyn nie może wykonać polecenia. Potwierdzenie pozytywne realizowane jest poprzez umieszczenie na polu funkcji kodu wykonanego rozkazu. W przypadku stwierdzenia błędu, jednostka slave umieszcza na polu funkcji szczególną odpowiedź, którą stanowi kod funkcji z ustawionym na 1 najstarszym bitem. Kod błędu umieszczany jest na polu danych ramki odpowiedzi

### Pole danych

Pole danych tworzy zestaw dwucyfrowych liczb heksadecymalnych, o zakresie 00-FF. Liczby te przy transmisji w trybie ASCII reprezentowane są dwoma znakami, a przy transmisji w trybie RTU jednym. Pole danych ramki polecenia zawiera dodatkowe informacje potrzebne jednostce slave do wykonania rozkazu określonego kodem funkcji. Mogą to być adresy rejestrów, liczba bajtów w polu danych, dane itp.

### Pole kontrolne

W protokole MODBUS słowo kontrolne zabezpieczające część informacyjną zależy od zastosowanego trybu transmisji.

W trybie ASCII pole kontrolne składa się z dwóch znaków ASCII, które są rezultatem obliczenia *Longitudinal Redundancy Check* (LRC) na zawartości części informacyjnej ramki (bez znaczników początku i końca). Znaki LRC są dołączane do wiadomości jako ostatnie pole ramki, bezpośrednio przed znacznikiem końca (CR,LF).

W trybie RTU słowo kontrolne jest 16-bitowe i powstaje jako rezultat obliczenia *Cyclical Redundancy Check* (CRC) na zawartości ramki. Pole kontrolne zajmuje dwa bajty dołączane na końcu ramki. Jako pierwszy przesyłany jest mniej znaczący bajt, jako ostatni starszy bajt, który jest jednocześnie znakiem kończącym ramkę.

## 6.4 Wyznaczenie LRC

Obliczanie LRC polega na sumowaniu kolejnych 8-bitowych bajtów wiadomości, odrzuceniu przeniesień i na koniec wyznaczeniu uzupełnienia dwójkowego wyniku. Sumowanie obejmuje całą

wiadomość za wyjątkiem znaczników początku i końca ramki. Wartość 8-bitowa sumy LRC jest umieszczana na końcu ramki w postaci dwóch znaków ASCII, najpierw znak zawierający starszą tetradę, a za nim znak zawierający młodszą tetradę LRC.

### 6.5 Wyznaczenie CRC

Obliczanie CRC realizowane jest według następującego algorytmu:

1. Załadowanie FFFFh do 16-bitowego rejestru CRC.
2. Pobranie bajtu z bloku danych i wykonanie operacji EXOR z młodszym bajtem rejestru CRC. Umieszczenie rezultatu w rejestrze CRC.
3. Przesunięcie zawartości rejestru CRC w prawo o jeden bit połączone z wpisaniem 0 na najbardziej znaczący bit (MSB=0).
4. Sprawdzenie stanu najmłodszego bitu (LSB) wysuniętego z rejestru CRC w poprzednim kroku. Jeżeli jego stan równa się 0, to następuje powrót do kroku 3 (kolejne przesunięcie), jeżeli 1, to wykonywana jest operacja EXOR rejestru CRC ze stałą A001h.
5. Powtórzenie kroków 3 i 4 osiem razy, co odpowiada przetworzeniu całego bajtu.
6. Powtórzenie sekwencji 2,3,4,5 dla kolejnego bajtu wiadomości. Kontynuacja tego procesu aż do przetworzenia wszystkich bajtów wiadomości.
7. Zawartość CRC po wykonaniu wymienionych operacji jest poszukiwaną wartością CRC.
8. Wartość CRC jest umieszczana na końcu ramki najpierw mniej znaczący bajt, a za nim bardziej znaczący bajt.

### 6.6 Format znaku przy transmisji szeregowej

W protokole MODBUS znaki są przesyłane od najmłodszego do najstarszego bitu.

Organizacja jednostki informacyjnej w trybie ASCII:

- 1 bit startu,
- 7 bitów pola danych,
- 1 bit kontroli parzystości (nieparzystości) lub brak bitu kontroli parzystości,
- 1 bit stopu przy kontroli parzystości lub 2 bity stopu przy braku kontroli parzystości

Organizacja jednostki informacyjnej w trybie RTU:

- 1 bit startu,
- 8 bitów pola danych,
- 1 bit kontroli parzystości (nieparzystości) lub brak bitu kontroli parzystości,
- 1 bit stopu przy kontroli parzystości lub 2 bity stopu przy braku kontroli parzystości.

### 6.7 Przerwanie transakcji

W jednostce master użytkownik ustawia ważny parametr jakim jest "maksymalny czas odpowiedzi na ramkę zapytania", po którego przekroczeniu transakcja jest przerywana. Czas ten dobiera się tak, aby każda jednostka slave pracująca w systemie (nawet ta najwolniejsza) zdążyła normalnie odpowiedzieć na ramkę zapytania. Przekroczenie tego czasu świadczy zatem o błędzie i tak jest traktowane przez jednostkę master.

Jeżeli jednostka slave wykryje błąd transmisji, nie wykonuje polecenia oraz nie wysyła żadnej odpowiedzi. Spowoduje to przekroczenie czasu oczekiwania na ramkę odpowiedzi i przerwanie transakcji.

W regulatorze RG24 "maksymalny czas odpowiedzi na ramkę zapytania" wynosi 500 ms.

### 6.8 Opis funkcji.

W regulatorze RG24 zaimplementowane zostały następujące funkcje protokołu:

kod	znaczenie
03	odczyt n-rejestrów
06	zapis pojedynczego rejestru
16	zapis n-rejestrów
17	identyfikacja urządzenia slave

### 6.8.1 Odczyt n-rejestrów (kod 03)

#### Żądanie:

Funkcja umożliwia odczyt wartości zawartych w rejestrach w zaadresowanym urządzeniu slave. Rejestry są 16-bitowymi jednostkami, które mogą zawierać wartości numeryczne związane ze zmiennymi procesowymi itp. Ramka żądania określa 16-bitowy adres początkowy rejestru oraz liczbę rejestrów do odczytania. Maksymalna liczba rejestrów do odczytania jednym rozkazem w regulatorze RG24 wynosi 20.

Funkcja nie jest dostępna w trybie rozgłoszeniowym.

Przykład: Odczyt 3 rejestrów zaczynając od rejestru o adresie 4010 (0FAAh):

adres	funkcja	adres rejestru Hi	adres rejestru Lo	liczba rejestrów Hi	liczba rejestrów Lo	suma LRC
01	03	0Fh	AAh	00	03	40h

#### Odpowiedź:

Dane rejestrów są pakowane począwszy od najmniejszego adresu: najpierw starszy bajt, potem młodszy bajt rejestru.

Przykład odpowiedzi:

adres	funkcja	liczba bajtów	wart. w rej. 4010 Hi	wart. w rej. 4010 Lo	wart. w rej. 4011 Hi	wart. w rej. 4011 Lo	wart. w rej. 4012 Hi	wart. w rej. 4012 Lo	suma LRC
01	03	06	00	14h	00	00	00	00	E2h

### 6.8.2 Zapis wartości do rejestru (kod 06)

#### Żądanie

Funkcja umożliwia modyfikację zawartości rejestru. Jest dostępna w trybie rozgłoszeniowym.

Przykład: Zapis wartości 30 (001Eh) do rejestru o adresie 4010 (0FAAh):

adres	funkcja	adres rejestru Hi	adres rejestru Lo	wartość Hi	wartość Lo	suma LRC
01	06	0Fh	AAh	00	1Eh	22h

#### Odpowiedź:

Prawidłową odpowiedzią na żądanie zapisu wartości do rejestru jest retransmisja komunikatu po wykonaniu operacji.

Odpowiedź:

adres	funkcja	adres rejestru Hi	adres rejestru Lo	wartość Hi	wartość Lo	suma LRC
01	06	0Fh	AAh	00	1Eh	22h

### 6.8.3 Zapis do n-rejestrów (kod 16)

#### Żądanie:

Funkcja dostępna w trybie rozgłoszeniowym. Umożliwia modyfikację zawartości rejestrów. Maksymalna liczba rejestrów do zapisania jednym rozkazem w regulatorze RG24 wynosi 20.

Przykład. Zapis dwóch rejestrów począwszy od rejestru o adresie 4010:

adres	funkcja	adres rej. Hi	adres rej. Lo	liczba rej. Hi	liczba rej. Lo	liczba bajtów	dane Hi	dane Lo	dane Hi	dane Lo	suma LRC
01	10h	0Fh	AAh	00	02	04	00	14h	00	3Ch	E0h

#### Odpowiedź:

Prawidłowa odpowiedź zawiera adres jednostki slave, kod funkcji, adres początkowy oraz liczbę zapisanych rejestrów.

Odpowiedź:

adres	funkcja	adres rej. Hi	adres rej. Lo	liczba rej. Hi	liczba rej. Lo	suma LRC
01	10h	0Fh	AAh	00	02	34h

#### 6.8.4 Raport identyfikujący urządzenie (kod 17)

##### Żądanie:

Funkcja pozwala użytkownikowi uzyskać informacje o typie urządzenia, statusie i zależności od tego konfiguracji.

Przykład:

adres	funkcja	suma LRC
01	11h	EEh

##### Odpowiedź:

Regulator RG24 w ramce odpowiedzi zwraca swój identyfikator (5Bh).

Odpowiedź dla regulatora RG24:

adres	funkcja	liczba bajtów	identyfikator	suma LRC
01	11h	01	5Bh	92h

#### 6.9 Kody błędów.

Gdy urządzenie master wysła żądanie do urządzenia slave, to za wyjątkiem komunikatów w trybie rozgłoszeniowym, oczekuje prawidłowej odpowiedzi. Po wysłaniu żądania jednostki master może wystąpić jedno z czterech możliwych zdarzeń: Jeżeli jednostka slave odbiera żądanie bez błędu transmisji oraz może je wykonać prawidłowo, wówczas zwraca prawidłową odpowiedź. Jeżeli jednostka slave nie odbiera żądania, żadna odpowiedź nie jest zwracana. W programie urządzenia master zostaną spełnione warunki timeout dla żądania. Jeżeli jednostka slave odbiera żądanie, ale z błędami transmisji (błąd parzystości, sumy kontrolnej LRC lub CRC), żadna odpowiedź nie jest zwracana. W programie urządzenia master zostaną spełnione warunki timeout dla żądania. Jeżeli jednostka slave odbiera żądanie bez błędu transmisji, ale nie może go wykonać prawidłowo (np. jeżeli żądaniem jest odczyt nie istniejącego wyjścia bitowego lub rejestru), wówczas zwraca odpowiedź zawierającą kod błędu, informując urządzenie master o przyczynie błędu.

Komunikat z odpowiedzią o błędzie zawiera dwa pola odróżniające go od prawidłowej odpowiedzi:

**Pole kodu funkcji:** W prawidłowej odpowiedzi, jednostka slave retransmituje kod funkcji z komunikatu żądania na polu kodu funkcji odpowiedzi. Wszystkie kody funkcji mają najbardziej znaczący bit (MSB) równy 0 (wartości kodów są poniżej 80h). W błędnej odpowiedzi urządzenie slave ustawia bit MSB kodu funkcji na 1. To powoduje, że wartość kodu funkcji w błędnej odpowiedzi jest dokładnie o 80h większa niż byłaby w prawidłowej odpowiedzi.

Na podstawie kodu funkcji z ustawionym bitem MSB program urządzenia master może rozpoznać błędną odpowiedź i może sprawdzić na polu danych kod błędu.

**Pole danych:** W prawidłowej odpowiedzi, urządzenie slave może zwrócić dane na polu danych (pewne informacje żądane przez jednostkę master). W odpowiedzi, urządzenie slave zwraca kod błędu na polu danych. Określa on warunki urządzenia slave, które spowodowały błąd.

Poniżej przedstawiono przykład żądania urządzenia master: odczytaj rejestr 4250 (109Ah) i odpowiedź urządzenia slave: niedozwolony adres danych, gdyż maksymalny adres rejestru w regulatorze RG24 wynosi 4201.

Przykład żądania:

adres	funkcja	adres rejestru Hi	adres rejestru Lo	liczba rejestrów Hi	liczba rejestrów Lo	suma LRC
01	03	10h	9Ah	00	01	51h

Przykładowa odpowiedź:

adres	funkcja	kod błędu	suma LRC
01	83h	02	7Ah

W poniższej tabeli przedstawione są możliwe kody błędów i ich znaczenie

kod	znaczenie	przyczyna wystąpienia błędu
01	niedozwolona funkcja	- funkcja spoza listy funkcji; - próba zapisu rejestru tylko do odczytu;
02	niedozwolony adres danych	- próba odczytu lub zapisu rejestru o adresie spoza dopuszczalnych wartości;
03	niedozwolona wartość danej	- próba zapisania danej spoza dopuszczalnego zakresu; - próba zapisania już istniejącej daty dnia dodatkowego; - próba odczytu lub zapisu pojedynczym rozkazem liczby rejestrów większej niż 20;
07	negatywne potwierdzenie	- podczas zapisu danych wystąpiła niezgodność deklarowanej liczby zapisywanych bajtów lub rejestrów z faktycznie przesłaną (niekompletna ramka danych przy prawidłowej sumie kontrolnej);

### 6.10 Dodatek A: Obliczanie sum kontrolnych LRC i CRC.

W dodatku tym przedstawiono przykłady funkcji w języku C, obliczające sumę kontrolną LRC dla trybu ASCII oraz CRC dla trybu RTU.

Funkcja do obliczenia LRC ma dwa argumenty:

```
unsigned char *outMsg;    Wskaźnik do bufora komunikacyjnego, zawierające-
                           go dane binarne, z których należy obliczyć LRC

unsigned short usDataLen; Liczba bajtów w buforze komunikacyjnym
```

Funkcja zwraca LRC typu *unsigned char*.

```
static unsigned char LRC(outMsg, usDataLen)
```

```
unsigned char *outMsg;    // bufor do obliczenia LRC
unsigned short usDataLen; // liczba bajtów w buforze
{
    unsigned char uchLRC = 0; // inicjalizacja LRC
    while (usDataLen--)
        uchLRC += *outMsg++; // dodaje bajt bufora bez przeniesienia
    return ((unsigned char) (-(char) uchLRC));
    // zwraca sumę w kodzie uzupełnienia do dwóch
}
```

Poniżej przedstawiono przykład funkcji w języku C obliczającej sumę CRC. Wszystkie możliwe wartości sumy CRC są umieszczone w dwóch tablicach. Pierwsza tablica zawiera starszy bajt wszystkich z 256 możliwych wartości 16-bitowego pola CRC, natomiast druga tablica młodszy bajt.

Wyznaczenie sumy CRC poprzez indeksowanie tablic jest o wiele szybsze niż obliczenie nowej wartości CRC dla każdego znaku z bufora komunikacyjnego.

**Uwaga:** Poniższa funkcja przestawia bajty sumy CRC starszy/młodszy, tak że wartość CRC zwracana przez funkcję może być bezpośrednio umieszczona w buforze komunikacyjnym.

Funkcja do obliczenia CRC ma dwa argumenty:

```
unsigned char *puchMsg;    Wskaźnik do bufora komunikacyjnego, zawierające-
                           go dane binarne, z których należy obliczyć CRC
```



*unsigned short usDataLen;* Liczba bajtów w buforze komunikacyjnym

Funkcja zwraca CRC typu *unsigned short*.

***unsigned short CRC16(puchMsg, usDataLen)***

```
unsigned char *puchMsg;           // bufor do obliczenia CRC
unsigned short usDataLen;       // liczba bajtów w buforze
{
    unsigned char uchCRChi = 0xFF;   // inicjalizacja starszego bajtu CRC
    unsigned char uchCRClo = 0xFF;   // inicjalizacja młodszego bajtu CRC
    while (usDataLen--)
    { uIndex = uchCRChi ^ *puchMsg++; // obliczenie CRC
      uchCRChi = uchCRClo ^ crc_hi[uIndex];
      uchCRClo = crc_lo[uIndex];
    }
    return(uchCRChi<<8 | uchCRClo);
}

```

//tablica starszego bajtu CRC

```
const unsigned char crc_hi[ ]={
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01,
0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80,
0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01,
0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80,
0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01,
0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
0x40
};

```

//tablica młodszego bajtu CRC

```
const unsigned char crc_lo[]={
0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7, 0x05, 0xC5, 0xC4,
0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09,
0x08, 0xC8, 0xD8, 0x18, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD,
0x1D, 0x1C, 0xDC, 0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32, 0x36, 0xF6, 0xF7,
0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A,
0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38, 0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE,
0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1, 0x63, 0xA3, 0xA2,
0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4, 0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F,
0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB,
0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0, 0x50, 0x90, 0x91,
0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C,

```

---

```
0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88,  
0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,  
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83, 0x41, 0x81, 0x80,  
0x40  
};
```



# LUMEL

## **LUMEL S.A.**

ul. Słubicka 4, 65-127 Zielona Góra, Poland  
tel.: +48 68 45 75 100, fax +48 68 45 75 508  
[www.lumel.com.pl](http://www.lumel.com.pl)

---

## **Informacja techniczna:**

tel.: (68) 45 75 140, 45 75 141, 45 75 142, 45 75 145, 45 75 146  
e-mail: [sprzedaz@lumel.com.pl](mailto:sprzedaz@lumel.com.pl)

## **Realizacja zamówień:**

tel.: (68) 45 75 150, 45 75 151, 45 75 152, 45 75 153, 45 75 154, 45 75 155  
fax.: (68) 32 55 650

## **Pracownia systemów automatyki:**

tel.: (68) 45 75 145, 45 75 145

## **Wzorcowanie:**

tel.: (68) 45 75 163  
e-mail: [laboratorium@lumel.com.pl](mailto:laboratorium@lumel.com.pl)

---

RG24-07/2  
60-006-00-00090