MSW8-LPM

wersja 0.1 (wersja robocza)

Dokumentacja użytkownika

Podstawy

Komunikacja z multiprzyciskiem MSW8-LPM dostępna jest za pomocą transmisji szeregowej EIA-485 (wcześniej RS-485) przy wykorzystaniu protokołu Modbus w trybie RTU. Urządzenie działa jako Slave. Format podstawowej ramki komunikacyjnej przedstawiony jest w tabeli 1. Prędkość komunikacji może być programowo zmieniana i może wynieść 1200bps, 2400bps, 4800bps, 9600bps lub 19200bps.

parametr	ustawienie
bit parzystości	brak
ilość bitów danych	8
ilość bitów stopu	1

Tabela 1: Format podstawowej ramki komunikacyjnej

Modbus

Multiprzycisk MSW8-LPM używa protokołu komunikacyjnego MODBUS w trybie RTU. Funkcje, na jakie może odpowiadać, to:

kod funkcji	opis
1	Odczyt wielu zmiennych typu coil
2	Odczyt wielu zmiennych typu discrete input
3	Odczyt wielu zmiennych typu holding register
4	Odczyt wielu zmiennych typu input register
5	Zapis pojedynczej zmiennej typu coil
6	Zapis pojedynczej zmiennej typu holding register
15	Zapis wielu zmiennych typu coil
16	Zapis wielu zmiennych typu holding register

Tabela 2: Funkcje dostępne na multiczujniku MSW8-LPM

MSW8-LPM wykorzystuje wszystkie bloki danych, jakie oferuje MODBUS, czyli coils, discrete inputs, holding registers i input registers. Organizacja bloków danych przedstawiona jest w tabeli 3.

adres (zakres 1)		adres (zakres 2)		zmienna	wartość domyślna	uwagi
hex	dec	hex	dec			
HOLDING REGISTERS (funkcja 3, 6):						
0x000D	13	0x8000	32768	actSoftStat	65280	
0x000E	14	0x8001	32769	usartSpeed	960	
0x000F	15	0x8002	32770	tempMultiTime	1	(A)
0x0010	16	0x8003	32771	jungWorkModesA	0	
0x0011	17	0x8004	32772	jungWorkModesB	0	
0x0012	18	0x8005	32773	oneTurnOffJung	100	(A)
0x0013	19	0x8006	32774	doubleTurnOffJung	50	(A)
0x0014	20	0x8007	32775	pressTurnOffJung	100	(A)
0x0015	21	0x8008	32776	upTurnOffJung	110	(A)
0x0016	22	0x8009	32777	downTurnOffJung	70	(A)
0x0017	23	0x800A	32778	doubleUpTurnOffJung	150	(A)
0x0018	24	0x800B	32779	doubleDownTurnOffJung	30	(A)
0x0019	25	0x800C	32780	accelRockerJung	50	(A)
0x001A	26	0x800D	32781	fastRockerJung	1	(B), (C)
0x001B	27	0x800E	32782	slowRockerJung	25	(B), (C)
0x001C	28	0x800F	32783	pressTimeJung	50	(B), (C)
0x001D	29	0x8010	32784	doubleTimeJung	25	(B), (C)
0x001E	30	0x8011	32785	diodeValueJung	0	(E)
0x0020-0x009F	32-159	0x8012-0x8091	32786-32913	jungIlumScenes	0	(C)
0x00A0-0x00A3	160-163	0x8092-0x8095	32914-32917	diodesPWM	0	(D)
0x00FF	255	-	-	slaveAddress	1	
COILS (funkcja 1, 5, 15):						
0x0000	0	0x8096 [0]	32918	stateDO1	0	
0x0001	1	0x8096 [1]	32918	stateDO2	0	
0x0002	2	0x8096 [2]	32918	stateDO3	0	
0x0010	16	0x8096 [3]	32918	configDO1	0	

0x0011	17	0x8096 [4]	32918	configDO2	0
0x0012	18	0x8096 [5]	32918	configDO3	0
0x0050-0x006F	80-111	-	-	Multi DS: usuniecie czujnika, który zapisany jest pod indeksem i, gdzie: i=adres-0x50 Odczyt tych coilsów ZAWSZE zwraca wartość 0.	
DISCRETE INPUTS (funkcja 2):					
0x0000	0	0x8096 [6]	32918	stateDI1	0
0x0001	1	0x8096 [7]	32918	stateDI2	0
0x0002	2	0x8096 [8]	32918	statePresence	0
0x0010	16	0x8096 [9]	32918	risingEdgeDI1	0
0x0011	17	0x8096 [10]	32918	risingEdgeDI2	0
0x0020	32	0x8096 [11]	32918	fallingEdgeDI1	0
0x0021	33	0x8096 [12]	32918	fallingEdgeDI2	0
0x0030	48	0x8096 [13]	32918	errorDIO1	0
0x0031	49	0x8096 [14]	32918	errorDIO1	0
0x0032	50	0x8096 [15]	32918	errorPresence	0
0x0050: 0x006F	80-111	0x8097-0x8098	32919-32920	Multi DS: brak odpowiedzi dla termometru który zapisany jest pod indeksem i, gdzie: zakres 1: i=adres-0x50 zakres2: 0x116[0]->indeks 0, 0x116[15]->indeks 15, 0x117[0]->indeks 16, 0x117[15]->indeks 31, itd.	
INPUT REGISTERS (funkcja 4):					
0x0004	4	0x8099	32921	avSensorsReg	0
0x0005	5	0x809A	32922	windSpeed	0
0x000D	13	0x809B	32923	bootVer	x
0x000E	14	0x809C	32924	softVer	x
0x0010	16	0x809D	32925	allDataSize	167
0x0011	17	0x809E	32926	dsCount	0
0x0012	18	0x809F	32927	jungSwitchesState	0
0x0013	19	0x80A0	32928	outDeviceOneClick	0

0x0014	20	0x80A1	32929	outDeviceDoubleClick	0
0x0015	21	0x80A2	32930	outDevicePress	0
0x0016	22	0x80A3	32931	jungRocker[0]	0
0x0017	23	0x80A4	32932	jungRocker[1]	0
0x0018	24	0x80A5	32933	jungRocker[2]	0
0x0019	25	0x80A6	32934	jungRocker[3]	0
0x0100: 0x011F	256: 287	0x80A7: 0x80A7 +dsCount -1	32935: 32935+ dsCount-1	tempDsReg	x
0x0120: 0x017F	288: 383	0x80A7 +dsCount : 0x80A7 +dsCount *4-1	32935 +dsCount : 32935 +dsCount *4-1	dsRomCode	x

Tabela 3: Organizacja danych na MSW8-LPM

Objaśnienie:

(A) wartość jest wielokrotnością 100ms

(B) wartość jest wielokrotnością 20ms

(C) maksymalna wartość, jaką można zapisać to 255

(D) maksymalna wartość, jaką można zapisać to 100

(E) maksymalna wartość, jaką można zapisać to 15

Opis szczegółowy danych

actSoftStat

Zmienna wykorzystywana jest przy aktualizacji oprogramowania.

usartSpeed

Zmienna określa prędkość komunikacji z jaką działa MSW8-LPM, gdzie usartSpeed=prędkość/10. Wartości, jakie może przyjąć ta zmienna to 120, 240, 480, 960, 1920, co oznacza komunikację z prędkością kolejno 1200bps, 2400bps, 4800bps, 9600bps i 19200bps. Przy próbie zapisu innej wartości, jest ona pomijana. Przy zapisie poprawnej wartości innej niż dotychczas, po wysłaniu ewentualnej odpowiedzi, multiczujnik restartuje się i zaczyna komunikację z nową prędkością.

tempMultiTime

Zmienna określa odstęp czasu między kolejnymi aktualizacjami temperatury dla czujników podłączonych pod wspólną linię.

Zmienna przyjmuje wartość większą od 0. Przy próbie zapisu wartości 0, zapisywane jest 1. Wartość tej zmiennej odpowiada wielokrotności 0,1 sekundy.

jungWorkModesA i jungWorkModesB

Zmienne konfigurujące tryb pracy par przycisków. Wartości zapisane w tych rejestrach należy rozpatrywać jako dwa niezależne bajty:

jungWorkModesA[7:0] – konfiguracja pary D0+D1

jungWorkModesA[15:8] – konfiguracja pary D2+D3

jungWorkModesB[7:0] – konfiguracja pary D4+D5

jungWorkModesB[15:8] – konfiguracja pary D6+D7

Dostępne tryby pracy:

0 – niezaprogramowany

1 – tryb niezależny I z diodą reagującą na stan pojedynczego kliknięcia

2 – tryb niezależny I z diodą reagującą na stan podwójnego kliknięcia

3 – tryb niezależny I z diodą reagującą na stan przytrzymania przycisku

4 – tryb niezależny II z diodą reagującą na stan pojedynczego kliknięcia

5 – tryb niezależny II z diodą reagującą na stan podwójnego kliknięcia

6 – tryb niezależny II z diodą reagującą na stan przytrzymania przycisku

7 – tryb roletowy I z diodą reagującą na stan pojedynczego kliknięcia

8 – tryb roletowy I z diodą reagującą na stan podwójnego kliknięcia

9 – tryb roletowy I z diodą reagującą na stan przytrzymania przycisku

10 – tryb roletowy II z diodą reagującą na stan pojedynczego kliknięcia

11 – tryb roletowy II z diodą reagującą na stan podwójnego kliknięcia

12 – tryb roletowy II z diodą reagującą na stan przytrzymania przycisku

13 – tryb rocker z diodą reagującą na stan pojedynczego kliknięcia

14 – tryb rocker z diodą reagującą na stan podwójnego kliknięcia

15 – tryb przełącznik scen I

16 – tryb przełącznik scen II

oneTurnOffJung

W trybie „niezależny II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy pojedynczego kliknięcia.

doubleTurnOffJung

W trybie „niezależny II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy podwójnego kliknięcia.

pressTurnOffJung

W trybie „niezależny II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy przytrzymania przycisku.

upTurnOffJung

W trybie „roletowy I” i „roletowy II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy pojedynczego kliknięcia prawego przycisku dla danej pary.

downTurnOffJung

W trybie „roletowy I” i „roletowy II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy pojedynczego kliknięcia lewego przycisku dla danej pary.

doubleUpTurnOffJung

W trybie „roletowy I” i „roletowy II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy podwójnego kliknięcia prawego przycisku dla danej pary.

doubleDownTurnOffJung

W trybie „roletowy I” i „roletowy II”, wartość określa czas, po jakim następuje samoistne przełączenie stanu przycisku na wartość 0. Dotyczy podwójnego kliknięcia lewego przycisku dla danej pary.

accelRockerJung

W trybie „rocker” wartość ta określa jak długo należy przytrzymać przycisk, aby zwiększyć szybkość z jaką zmienia się wartość w odpowiadającym dla danej pary przycisków rejestrze jungRocker[x].

fastRockerJung

W trybie „rocker” wartość ta określa co jaki czas, na skutek przytrzymania przycisku, następuje zmiana wartości w odpowiadającym dla danego przycisku rejestrze jungRocker[x]. Wartość ta jest wykorzystywana gdy przycisk zostanie przytrzymany przez czas większy niż określony w zmiennej accelRockerJung.

slowRockerJung

W trybie „rocker” wartość ta określa co jaki czas, na skutek przytrzymania przycisku, następuje zmiana wartości w odpowiadającym dla danego przycisku rejestrze jungRocker[x]. Wartość ta jest wykorzystywana gdy przycisk zostanie przytrzymany przez czas krótszy niż określony w zmiennej accelRockerJung.

pressTimeJung

Wartość określa przez jaki czas musi być przytrzymany przycisk, aby wykryć stan przytrzymania wciśniętego przycisku.

doubleTimeJung

Wartość określa przedział czasu w którym mają nastąpić pojedyncze wciśnięcia przycisku, tak, aby łącznie odczytać te wciśnięcia jako podwójne kliknięcie.

diodeValueJung

Podczas gdy dana para przycisku jest niezaprogramowana, zmienna ta pozwala kontrolować diodę znajdującą się między przyciskami. Najmłodszy bit tej wartości odpowiada za zaświecenie diody znajdującej się przy pierwszej parze przycisków, a czwarty z kolei bit odpowiada za diodę znajdującą się przy czwartej parze przycisków.

Przykład: W przypadku gdy wszystkie pary przycisków są niezaprogramowane, wartość 10 (binarnie: 0b0000000000000001010) spowoduje zapalenie się diody przy drugiej i czwartej parze przycisków. W przypadku gdy dwie pierwsze pary przycisków są zaprogramowane a dwie kolejne, nie są zaprogramowane, zaświeci się dioda przy czwartej parze przycisków.

jungIlumScenes

Obszar ten składa się z ośmiu bloków danych, gdzie każdy z nich zawiera 16 rejestrów. Każdy

blok danych przypisany jest do kolejnego przycisku. Dane te wykorzystywane są w trybie „przełącznika scen I” oraz „przełącznika scen II”.

0x0020	32	
0x0021	33	
0x0022	34	
0x0023	35	
0x0024	36	
0x0025	37	
0x0026	38	
0x0027	39	
0x0028	40	
0x0029	41	
0x002A	42	
0x002B	43	
0x002C	44	
0x002D	45	
0x002E	46	
0x002F	47	

Ilustracja 1: Blok danych dla przycisku pierwszego

Adresy poszczególnych rejestrów w bloku danych przedstawiony jest na ilustracji 1. Przedstawione adresy dotyczą pierwszego przycisku. Dla kolejnych przycisków następuje przesunięcie adresów w odpowiednią wielokrotność 16.

diodesPWM

Zestaw czterech rejestrów które określają procentowy poziom wypełnienia dla każdego z czterech sygnałów PWM. Kolejne wartości odpowiadają sygnałowi podawanemu na DO1, DO2, nóżce SDA i nóżce SCL. Sygnał synchronizujący te kanały odczytywany jest z nóżki INT.

slaveAddress

Zmienna ta określa adres własny multiczujnika MSW8-LPM, wykorzystywany w protokole komunikacyjnym Modbus, który jest aktualnym adresem multiczujnika gdy na dipswitchu ustawiona jest wartość 255. Dostęp do tego rejestru możliwy jest za pomocą funkcji 3 lub 6, tylko gdy na dipswitchu jest wartość 0 lub 255. Do rejestru można zapisać wartości z przedziału [1, 255]. Jeśli na dipswitchu ustawiona była wartość 255 oraz zmienna slaveAddress uległa zmianie, następuje restart multiczujnika.

stateDO1, stateDO2

Zmienne określają poziom napięcia które jest wymuszane na DO1 lub DO2, w przypadku kiedy są one ustawione jako wyjście.

Wartość 0 oznacza niskie napięcie, wartość 1-wysokie.

configDO1, configDO2

Konfiguracja linii DO/DI.

Wartość 0 oznacza że linia pracuje jako wejście, wartość 1 oznacza pracę jako wejście i wyjście.

stateDI1, stateDI2

Zmienne określają poziom napięcia na DI1 lub DI2.

Wejścia są zwierne do masy, w związku z czym wartość 0 oznacza że na linii jest wysokie napięcie, wartość 1 oznacza niskie napięcie na linii.

statePresence

nieobsługiwane

risingEdgeDI1, risingEdgeDI

Pola te określają, czy wykryto narastające zbocze od czasu poprzedniego odczytu. Po odczytaniu tej wartości, pola te są zerowane.

Wartość 0 oznacza brak zbocza narastającego od poprzedniego odczytu, wartość 1 oznacza wykrycie zbocza narastającego.

fallingEdgeDI1, fallingEdgeDI2

Pola te określają, czy wykryto opadające zbocze od czasu poprzedniego odczytu. Po odczytaniu tej wartości, pola te są zerowane.

Wartość 0 oznacza brak zbocza opadającego od poprzedniego odczytu, wartość 1 oznacza wykrycie zbocza opadającego.

errorDIO1, errorDIO2

W przypadku konfiguracji linii jako wyjście (DO), testowane jest, czy żądany poziom napięcia jest zgodny z faktycznym. Jeśli nie jest, zgłaszany jest błąd poprzez zapis wartości 1 na polu errorDIO1 lub errorDIO2.

avSensorsReg

Zmienna jest zestawem flag, określających dostępność czujników. Bit o wartości 1 oznacza że czujnik jest dostępny, 0-czujnik niedostępny.

	0	...	PWM	0	0	0	**WND**	**MUL**	0	0	**0x0004 (0x8099)**
bit	15	...	7	6	5	4	3	2	1	0	

MUL - temperatura (na wspólnej linii)

WND - prędkość wiatru

PWM - kanały PWM

windSpeed

Zmienna przedstawia prędkość, z jaką wykrywane są zbocza opadające na linii DI1 i prezentuje je jako ilość spadków napięcia na 10s. Pomiar dokonywany jest między dwoma kolejnymi spadkami. Przy braku spadku w czasie 10s, następuje wyzerowanie wartości.

bootVer

Wersja programu bootloader.

softVer

Wersja programu głównego.

allDataSize

Ilość możliwych do odczytania rejestrów przy adresacji z zakresu 2.

dsCount

Ilość czujników temperatury podłączonych do wspólnej linii.

jungSwitchesState

Aktualne odzwierciedlenie stanu poszczególnych przycisków, najmłodszy bit – pierwszy przycisk itd.

outDeviceOneClick

Stan urządzenia. Stan ten zależy od wybranej funkcjonalności poszczególnych przycisków.

outDeviceDoubleClick

Stan urządzenia (podwójne kliknięcie). Stan ten zależy od wybranej funkcjonalności poszczególnych przycisków.

outDevicePress

Stan urządzenia (przytrzymanie przycisku). Stan ten zależy od wybranej funkcjonalności poszczególnych przycisków.

jungRocker[0]

W trybie „rocker” wartość ta przypisana jest dla pierwszej pary przycisków. Przechowuje wartość którą można modyfikować przez tą parę przycisków.

jungRocker[1]

W trybie „rocker” wartość ta przypisana jest dla drugiej pary przycisków. Przechowuje wartość którą można modyfikować przez tą parę przycisków.

jungRocker[2]

W trybie „rocker” wartość ta przypisana jest dla trzeciej pary przycisków. Przechowuje wartość którą można modyfikować przez tą parę przycisków.

jungRocker[3]

W trybie „rocker” wartość ta przypisana jest dla czwartej pary przycisków. Przechowuje wartość którą można modyfikować przez tą parę przycisków.

tempDsReg

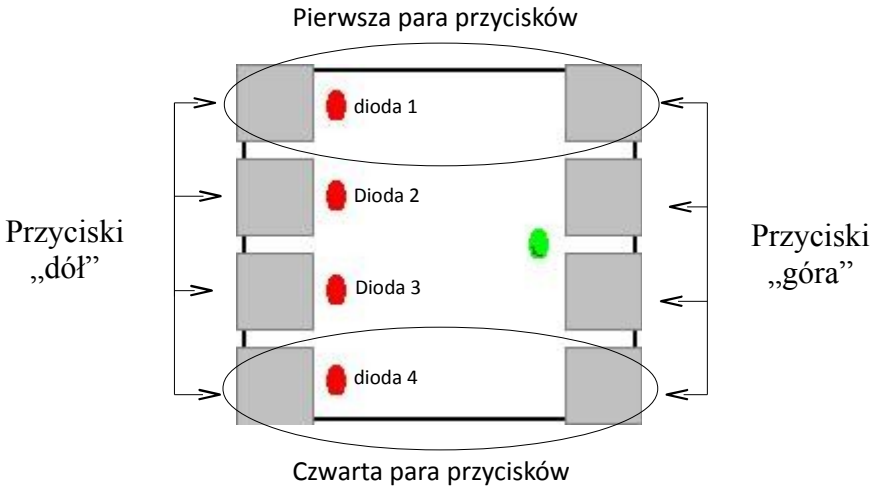
Tablica zawierająca pomiary temperatur z termometrów podłączonych do wspólnej linii.

dDsRomCode

Tablica zawierająca numery seryjne termometrów podłączonych do wspólnej linii. Każdy numer seryjny składa się z 3 rejestrów (6 bajtów). Identyfikacja termometru, którego dotyczy dana wartość odbywa się na podstawie nr indeksu w tablicy.

Przykład: Dostępnych jest 5 czujników na wspólnej linii. Termometr, którego numer seryjny zapisany jest jako pierwszy (pierwsze 3 rejestry) odpowiada pierwszej temperaturze z tablicy tempDsReg. Termometr, którego numer seryjny zapisany jest jako drugi (kolejne 3 rejestry) odpowiada drugiej temperaturze z tablicy tempDsReg, itd.

Opis szczegółowy przycisku i trybów pracy



0. Tryb niezaprogramowanyv

W trybie tym nie działa żadna logika przycisków. Niezaprogramowane przyciski oddziałują tylko na zmienną jungSwitchesState. Poprzez zapis zmiennej diodeValueJung można kontrolować diodę dla niezaprogramowanej pary przycisków.

1-3. Tryby niezależne I

Tryby niezależne I oddziałują na rejestry outDeviceOneClick, outDeviceDoubleClick oraz outDevicePress. Kliknięcie/podwójne kliknięcie/przytrzymanie przycisku ustawia wartość 1 na odpowiednim bicie. Ponowne kliknięcie/podwójne kliknięcie/przytrzymanie przycisku zeruje ten bit.

Ze względu na wybrany podtryb, różne jest zachowanie diody właściwej dla danej pary. Może być ona kontrolowana przez rejestr outDeviceOneClick, outDeviceDoubleClick lub outDevicePress. Dioda świeci się gdy suma logiczna stanów przycisków (właściwych dla tej diody) jest różna od zera.

4-6. Tryby niezależne II

Tryby oparte są na niezależnym I. Dodatkowo uwzględniane są wartości oneTurnOffJung, doubleTurnOffJung oraz pressTurnOffJung w których zawarte są czasy, po jakich nastąpi samoczynne wyzerowanie stanu przycisku. Dla każdego rodzaju przyciśnięcia (pojedyncze.podwójne/przytrzymanie) można niezależnie określić czas wyłączenia.

Zachowanie diody pośredniczącej dla przycisków jest takie same jak w trybie niezależnym I.

7-9. Tryb roletowy I

Kliknięcie w prawy przycisk z danej pary (tzw. „góra”) powoduje wyzerowanie stanu przycisku lewego (tzw. „dół”) i ustawienie własnego stanu na 1. Po czasie określonym w rejestrze upTurnOffJung następuje automatyczne wyzerowanie stanu przycisku.

Kliknięcie w lewy przycisk z danej pary (tzw. „dół”) powoduje wyzerowanie stanu przycisku prawego (tzw. „góra”) i ustawienie własnego stanu na 1. Po czasie określonym w rejestrze downTurnOffJung następuje automatyczne wyzerowanie stanu przycisku.

Przy pojedynczym kliknięciu zmiany obserwowane są w rejestrze outDeviceOneClick. Analogicznie działanie jest przy podwójnym kliknięciu. Wówczas efekt działania widoczny jest w rejestrze outDeviceDoubleClick a odstępy czasowe zerujące stan zawarte są w doubleUpTurnOffJung i doubleDownTurnOffJung.

Efekt działania przytrzymania – do poprawki.

Wybór podtrybu określa sposób zachowania diody pośredniczącej, która reaguje na zmianę w outDeviceOneClick, outDeviceDoubleClick lub outDevicePress. Kiedy lewy lub prawy przycisk jest włączony, odpowiednia dla nich dioda mruga do momentu wyzerowania stanu przycisków.

10-12. Tryb roletowy II

Sposób obsługi i zachowania diody jest taki sam jak w trybie roletowym I. Różnica zachodzi w prezentacji danych w rejestrze outDeviceOneClick i outDeviceDoubleClick. Po włączeniu przycisku prawego (tzw. „góra”) następuje ustawienie młodszego bitu (młodszego spośród dwóch bitów właściwych dla danej pary przycisków), co oznacza kierunek działania („góra”) oraz ustawienie starszego bitu który oznacza działanie („w toku”). Po czasie upTurnOffJung następuje wyzerowanie starszego bitu, co oznacza „brak działania”. Po włączeniu przycisku lewego (tzw. „dół”) następuje wyzerowanie młodszego bitu (młodszego spośród dwóch bitów właściwych dla danej pary przycisków), co oznacza kierunek działania („dół”) oraz ustawienie starszego bitu który oznacza działanie („w toku”). Po czasie downTurnOffJung następuje wyzerowanie starszego bitu, co oznacza „brak działania”.

Dioda pośrednicząca dla danej pary przycisków zachowuje się według wybranego podtrybu.

13-14. Tryb rocker

Z trybem tym związane są rejestry danych jungRocker[0-3] oraz rejestry konfiguracyjne accelRockerJung, fastRockerJung oraz slowRockerJung. Poprzez przytrzymanie któregoś z przycisków zmienia się wartość właściwy dla danego pola jungRocker[x]. Pojedyncze kliknięcie powoduje aktywację/dezaktywacje (wypełnienie zerami) właściwego rejestru jungRocker[x].

15-16. Tryb przełącznik scen świetlnych (typ I i II)

Przytrzymanie przycisku powoduje zapis aktualnych danych, które znajdują się w bloku odpowiadającym dla danego przycisku Zapis sygnalizowany jest przez krótkie mignięcie diody. Pojedyncze kliknięcia powodują włączanie/wyłączanie danej sceny świetlnej (tryb przełącznik scen świetlnych I). W trybie przełącznik scen świetlnych II aktywacja sceny następuje w wyniku kliknięcia w przycisk, a deaktywacja sceny następuje w wyniku podwójnego kliknięcia.